

構造物強度データベースシステム
“STAR”

Version 7.1

ソフトウェア詳細説明書

1998年12月

核燃料サイクル開発機構
大洗工学センター

本資料の全部または一部を複写・複製・転載する場合は、下記にお問い合わせ
してください。

〒319-1194 茨城県那珂郡東海村村松4番地49
核燃料サイクル開発機構
技術展開部 技術協力課

Inquiries about copyright and reproduction should be addressed to:
Technical Cooperation Section,
Technology Management Division,
Japan Nuclear Cycle Development Institute
4-49 Muramatsu, Tokai-mura, Naka-gun, Ibaraki 319-1194,
Japan

© 核燃料サイクル開発機構 (Japan Nuclear Cycle Development Institute)
1998

構造物強度データベース・システム
“STAR”
Version 7.1

ソフトウェア詳細説明書

細貝 広視*、川崎 信史**、笠原 直人***

要 旨

STAR システムは、大洗工学センターで実施される熱過渡強度試験に関する試験条件・試験結果・解析結果等をデータベース化し、これらのデータに対して表計算処理・ユーザープログラムによる損傷値計算処理・種々の検索処理を行うことができるクライアントサーバー型のデータベースシステムである。本システムのクライアントプログラムは Visual Basic 5.0 (Enterprise Edition、Service Pack 3) で作成され、データベース管理には Microsoft SQL Server 6.5 を使用しており、Windows 95 あるいは Windows NT 4.0 のもとで動作する。

バージョン 6 では、データベースエンジンとして Microsoft Access で使用されている Jet 3.0(32ビット版)を使用してデータベース(MDB ファイル)を構築したが、バージョン 7 ではデータベース管理システムとして SQL Server を用いることにより本格的なクライアント・サーバ方式への拡張を行った。

データベース・テーブル構造も SQL Server への移行に伴いいくつかの改良を行った。例えば、解析テーブルを時点のテーブルと新規の時点テーブルに分割し、解析テーブルと時点テーブルとのリンクによりこれらに関連付けることによりデータ構造を柔軟なものにした。

また、表計算、損傷値計算、補間処理等の付加機能に関しては STAR 本体とは分割し、それぞれのプログラムを OLE サーバーまたは OCX(OLE カスタムコントロール)化することにより、自律部品として実装した。

これにより上記の付加機能は STAR 以外のプログラムからでも使用できるようになった。

更に構造解析に使用した解析入力データファイル名を示すフィールドを管理データテーブルに追加することにより、STAR に蓄積されている解析結果の元になる解析入力データファイルを呼び出す機能を実現した。

* 常陽産業株式会社

** 大洗工学センター システム技術開発部 構造・材料技術開発Gr (現 日本原子力発電株式会社)

*** 大洗工学センター システム技術開発部 構造・材料技術開発Gr

目 次

1.	概要	1
1.1	動作環境	1
1.2	システム構成	5
1.2.1	クライアント構成	5
1.2.2	サーバー構成	6
1.3	ネットワークへの対応	6
2.	STARデータベースの構造	8
2.1	テーブル定義	8
2.2	テーブル結合	22
2.3	管理データテーブル	23
3.	STARのソースモジュール	26
3.1	各モジュールの概要	27
3.2	STAR 7.1の変更点	38
3.3	STARソースプログラムのインストール	41
4.	AutoSketchによる形状図表示	43
4.1	概要	43
4.2	AutoSketchの起動	44
4.3	画層	44
4.4	入出力仕様	45
5.	外部プログラム	47
5.1	表計算プログラム	47
5.1.1	モジュール構成	47
5.1.2	プロシージャ	47
5.1.3	表計算式	47
5.1.4	TCALCSVの実行例	50
5.2	補間処理プログラム	50
5.2.1	モジュール構成	50
5.2.2	プロシージャ	51
5.2.3	処理	51
5.2.4	INTPSVの実行例	52
5.3	損傷プログラム	52
5.3.1	プロシージャ	52
5.3.2	DAMAGE96の例	54
6.	オンラインヘルプ	56
6.1	ヘルプファイルの概要	56
6.2	ヘルプの番号	56

6.3	リッチテキストファイル	61
6.4	プロジェクトファイル	62
6.5	ヘルプファイルのコンパイル.....	63
7.	ディジタイザ入力システム	64
7.1	出力ファイルの流れ	64
7.2	BASEBLK. DWGの仕様.....	65
	謝辞	66
	参考文献	67
付録		
A.	STAR本体ソースリスト STAR.EXE	69
B.	表計算プログラムソースリスト TCALCSV.EXE	269
C.	補間処理プログラムソースリスト INTPSV.EXE	287

List of Tables

Table 1.1 STAR System Files 5

Table 1.2 Table of STAR Sysytem 6

Table 1.3 STAR Commnads 7

Table 2.1 Table Contents 8

Table 2.2 Structure of Management Data Table 8

Table 2.3 Structure of Analysis Case Table 9

Table 2.4 Structure of Evaluation Method Table..... 9

Table 2.5 Structure of Materials Data Table..... 10

Table 2.6 Structure of Crack & Crack Supplement Data Table..... 10

Table 2.7 Structure of Test Condition Table 11

Table 2.8 Structure of Temperature Data Table 12

Table 2.9 Structure of Shape Data Table 12

Table 2.10 Structure of Analysis & Analysis Supplement Data Table 13

Table 2.11 Structure of Point in Time & Point in Time Supplement Data Tabale 13

Table 2.12 Structure of Calculation & Calculation Supplement Data Table..... 14

Table 2.13 Structure of TTSDS Method Table 15

Table 2.14 Structure of BDS Method Table 16

Table 2.15 Structure of BDSMD Method Table..... 16

Table 2.16 Structure of DDS Method Table 17

Table 2.17 Structure of DDSWELD Method Table 17

Table 2.18 Structure of EQEF Method Table 18

Table 2.19 Structure of EQEFWELD Method Table..... 19

Table 2.20 Structure of EQEF15 Method Table 19

Table 2.21 Structure of EQEF167 Method Table..... 20

Table 2.22 Structure of INELAST Method Table 21

Table 2.23 Structure of Management Data Table..... 23

Table 3.1 List of Source Module File..... 26

Table 3.2 Summary of Source Program File "B_SELECT.FRM" 27

Table 3.3 Summary of Source Program File "I_CND_RG.FRM" 27

Table 3.4 Summary of Source Program File "I_CRK_IP.FRM" 28

Table 3.5 Summary of Source Program File "I_VW_DBG.FRM" 28

Table 3.6 Summary of Source Program File "I_VW_DXF.FRM" 28

Table 3.7	Summary of Source Program File	"I_VW_GRD.FRM"	29
Table 3.8	Summary of Source Program File	"N_ANL_CS.FRM"	29
Table 3.9	Summary of Source Program File	"N_ANL_VW.FRM"	29
Table 3.10	Summary of Source Program File	"N_DGM_CT.FRM"	29
Table 3.11	Summary of Source Program File	"N_DMG_SL.FRM"	30
Table 3.12	Summary of Source Program File	"N_DMG_VS.FRM"	30
Table 3.13	Summary of Source Program File	"N_FNS_VW.FRM"	30
Table 3.14	Summary of Source Program File	"N_Q_SEL.FRM"	31
Table 3.15	Summary of Source Program File	"N_TBL_CT.FRM"	31
Table 3.16	Summary of Source Program File	"S_BUBUN2.FRM"	31
Table 3.17	Summary of Source Program File	"S_CASE2.FRM"	31
Table 3.18	Summary of Source Program File	"S_VIEW.FRM"	31
Table 3.19	Summary of Source Program File	"T_DAT_SL.FRM"	32
Table 3.20	Summary of Source Program File	"T_INT_CT.FRM"	32
Table 3.21	Summary of Source Program File	"T_INT_SL.FRM"	33
Table 3.22	Summary of Source Program File	"T_SELECT.FRM"	33
Table 3.23	Summary of Source Program File	"U_CAS_AD.FRM"	33
Table 3.24	Summary of Source Program File	"U_CASE.FRM"	33
Table 3.25	Summary of Source Program File	"U_HYK_AD.FRM"	34
Table 3.26	Summary of Source Program File	"U_HYOKA.FRM"	34
Table 3.27	Summary of Source Program File	"U_KANRI2.FRM"	34
Table 3.28	Summary of Source Program File	"U_KNR_A2.FRM"	35
Table 3.29	Summary of Source Program File	"U_MAT.FRM"	35
Table 3.30	Summary of Source Program File	"U_SETUP.FRM"	35
Table 3.31	Summary of Source Program File	"U_SQL.FRM"	35
Table 3.32	Summary of Source Program File	"U_VIEW.FRM"	36
Table 3.33	Summary of Source Program File	"X_NORMAL.FRM"	36
Table 3.34	Summary of Source Program File	"X_RESGRD.FRM"	36
Table 3.35	Summary of Source Program File	"X_RESOLE.FRM"	36
Table 3.36	Summary of Source Program File	"X_SELDB1.FRM"	37
Table 3.37	Summary of Source Program File	"X_SELDB3.FRM"	37
Table 3.38	Summary of Source Program File	"X_USER2.FRM"	37
Table 3.39	Entity Type Table		38
Table 3.40	Table of Field Item		39
Table 3.41	Table of Calculation Item		40
Table 3.42	Correspond Table of Management Data – Backup Data		40
Table 3.43	Entry Item Table		41
Table 3.44	List of Entry Process		41
Table 4.1	Correspond Table of STAR Data		43
Table 4.2	Correspond Table of Shape Data		44
Table 4.3	Correspond Table of Block Data		46

Table 5.1	Summary of Program Module "TCALCV.EXE"	47
Table 5.2	Summary of Procedure Module "TCALCSV.CLS"	47
Table 5.3	Summary of Array Table	48
Table 5.4	Explanation of Excel Method	48
Table 5.5	Explanation of Module Program "INTPSV.EXE"	50
Table 5.6	Summary of Procedure Module	51
Table 5.7	Summary of Excel Method Parameter	51
Table 5.8	summary of Procedure Module "DAMAGE96.OCX"	52
Table 5.9	Evaluation Method Table	53
Table 5.10	Explanation of Module Parameter "SetTestCondition"	53
Table 5.11	Explanation of Module Parameter "SetTestCondition"	53
Table 5.12	Explanation of Module Parameter "SetRangeAnalysis"	53
Table 5.13	Explanation of Module Parameter "SetRangeAnalysis"	53
Table 5.14	Explanation of Module Parameter "GetDamageResult"	54
Table 6.1	Summary of Help File "STAR.HLP"	56
Table 6.2	Explanation of Help Number "WhatsThisHelpID"	56
Table 6.3	Explanation of Help Number	60

List of Figures

Fig1.1	Concept of the STAR System	3
Fig1.2	Outline of the STAR Program	4
Fig 2.1	Relation of STAR Table	22
Fig 6.1	Edit Menu "STAR.RTF"	62
Fig 7.1	Output File Flow	64
Fig 7.2	Specification of Shape File "BASEBLK.DWG"	65

1. 概要

STAR システムは、大洗工学センターで実施される熱過渡強度試験に関する試験条件・試験結果・解析結果等をデータベース化し、これらのデータに対して表計算処理・ユーザープログラムによる損傷値計算処理・種々の検索処理を行うことができるデータベースシステムである (Fig1.1)。本システムは Windows 95 あるいは Windows NT 4.0 のもとで動作し、プログラム本体は Visual Basic 5.0 (Enterprise Edition、Service Pack 3) で作成され、データベースには Microsoft SQL Server 6.5 を使用している。SQL Server は Windows NT Server 上で実行されるクライアント・サーバー方式の本格的データベースシステムである。

STAR システム・プログラムの概要を Fig1.2 に示す。

バージョン 6 では、データベースエンジンとして Microsoft Access で使用されている Jet 3.0(32ビット版)を使用してデータベース(MDB ファイル)を構築したが、バージョン 7 ではデータベースとして SQL Server を用いることにより本格的なクライアント・サーバ方式への拡張を行った。

データベース構造も SQL Server への移行に伴いいくつかの改良を行った。例えば、解析テーブルの各時点データ部を新規の時点テーブルに分割し、解析テーブルと時点テーブルとの各行をリンクするように修正した。

表計算、損傷値計算、補間処理の機能を STAR 本体の機能とは分割し、それぞれの機能を OLE サーバーまたは OCX(OLE カスタムコントロール)することにより、個々の機能の部品化を行った。これにより OLE サーバーや OCX は STAR 以外のプログラムからでも使用できるようになっている。

構造解析に使用した解析入力データファイル名を示すフィールドを管理データテーブルに追加し、STAR 使用時にこのテーブルから解析入力データファイルを呼び出す機能を解析メニューの下に用意した。また、各コマンドおよびフォーム上の各コントロールに対するオンラインヘルプを作成した。

また、バージョン 7.1 では、エラー処理機能の強化およびグラフィック処理機能の強化を行った。エラー処理では、各コマンドへのエラー処理ルーチンの追加による STAR 作業中のシステムの異常終了の防止や、データ登録時のデータチェックおよびバックアップ機能の強化を図っている。グラフィック処理機能では、形状データ入力における旧バージョンの DXF ファイルの読み込み機能の追加や、AutoSketch のレイヤ機能を使った出力図の多層管理を実現した。

本説明書の構成は以下のようになっている。

- 1 章 概要
- 2 章 STAR データベースの構造
- 3 章 STAR のソースモジュール
- 4 章 AutoSketch による形状図の表示
- 5 章 外部プログラム
- 6 章 オンラインヘルプ
- 7 章 デジタイザ入力システム

1.1 動作環境

STAR システムは SQL Server をデータベース管理システムとするサーバー・クライアント方式のデータベースシステムである。したがって、データベースサーバーには、SQL Server 6.5 が動作する Windows NT Server 4.0 が必要である。クライアントとしては Windows 95 あるいは Windows NT 4.0 が動作するマシンが必要である。Windows 3.1、Windows NT 3.51 では動作しない。また、STAR システムからは以下のアプリケーションを起動するので、予めこれらのアプリケーションをインストールしておく必要がある。

- ・サーバー
 - ・ Windows NT Server 4.0
 - ・ SQL Server 6.5

- ・クライアント
 - ・ Windows 95, または Windows NT 4.0
 - ・ EXCEL 97
 - ・ ACCESS 97 (ACCESS 2.0 は使用不可)
 - ・ AutoSketch 2.2 (Windows 95/NT4.0 対応版)
 - ・ AutoCAD R13 (ディジタイザ入力システムを使用する場合のみ必要である)

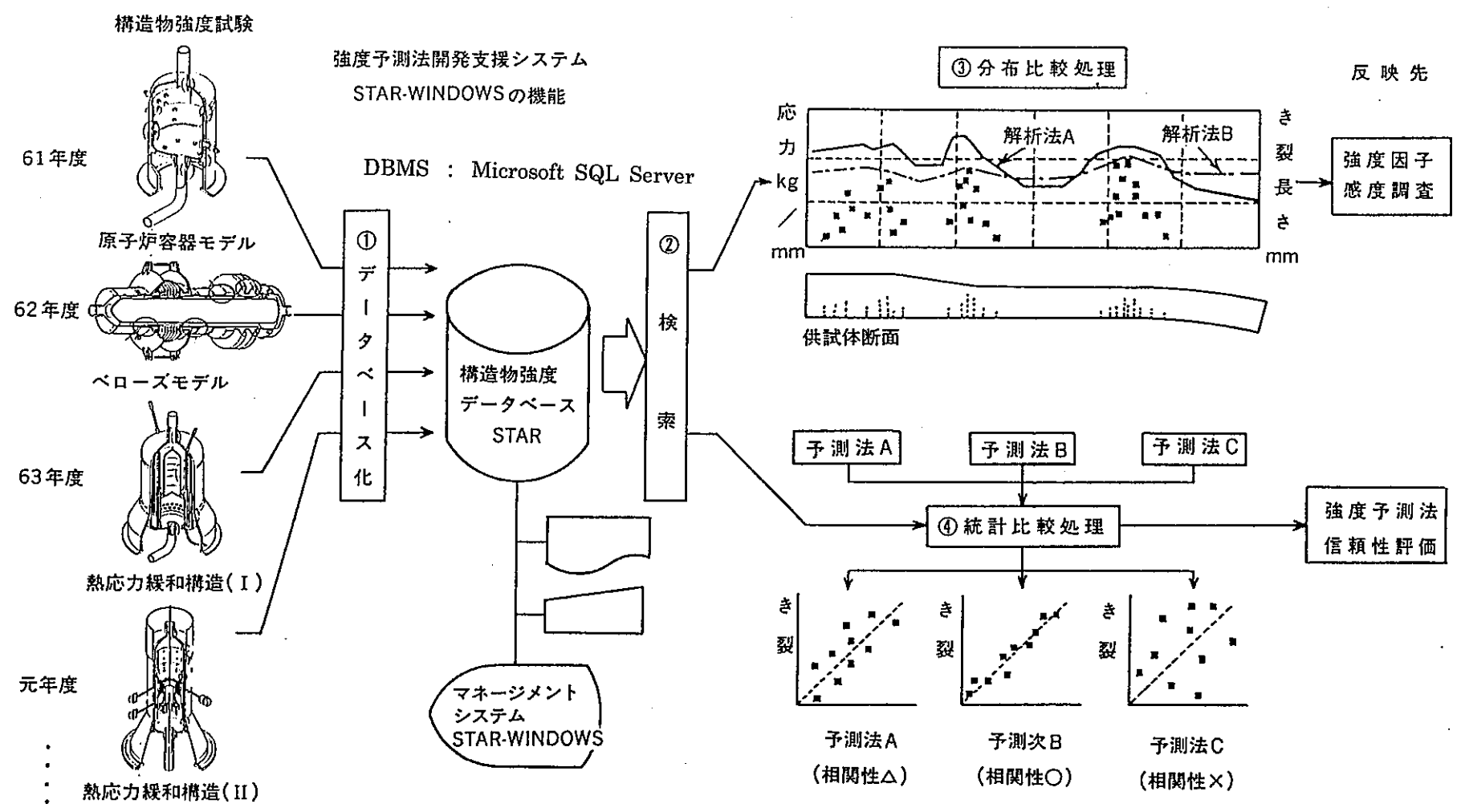


Fig 1.1 Concept of the STAR System

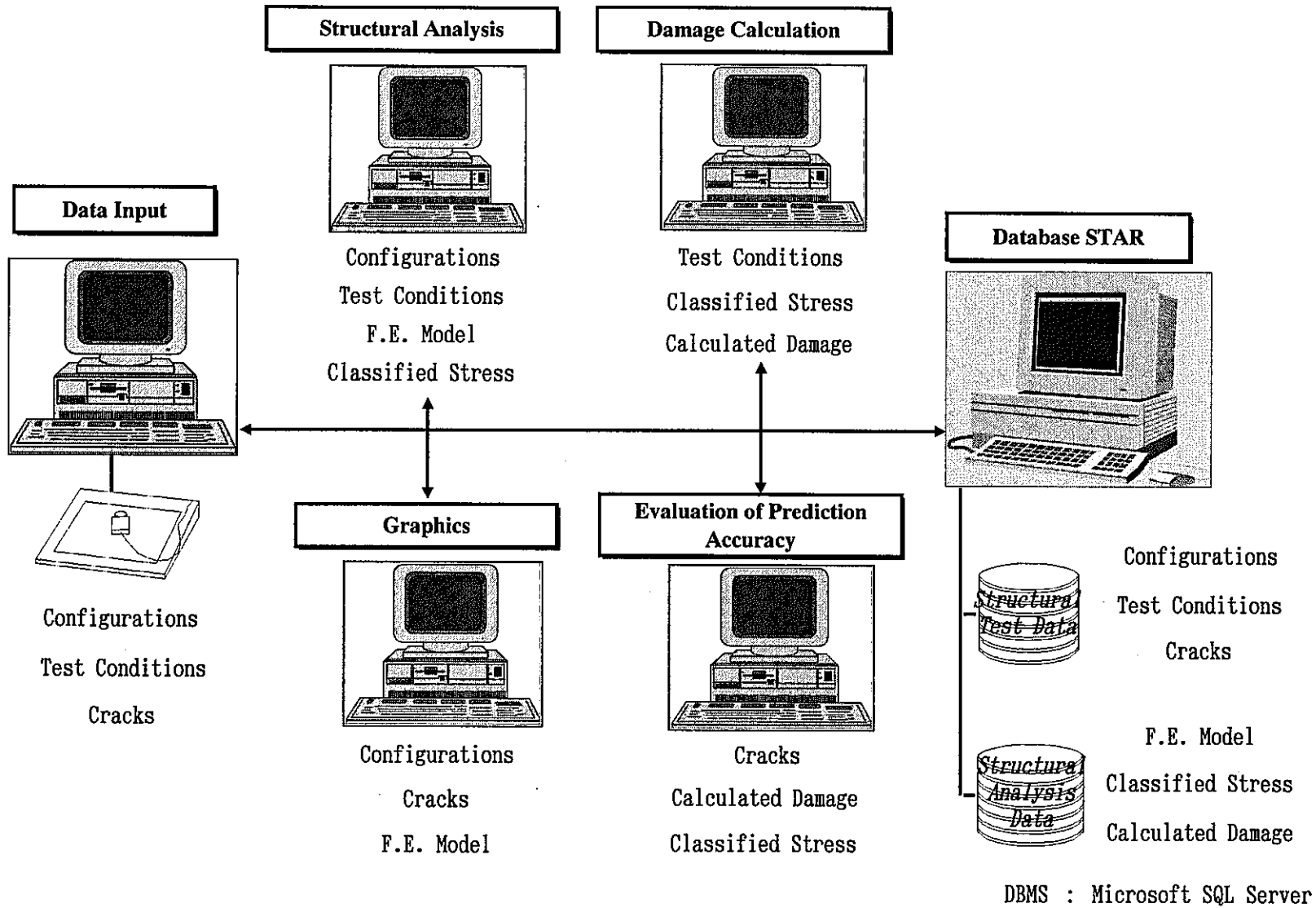


Fig1.2 Outline of the STAR Program

1.2 システム構成

1.2.1 クライアント構成

STAR クライアント・システムで使用するファイルの一覧とその概要を以下に示す。

Table 1.1 STAR System Files

ファイル名	ディレクトリ	各ファイルの機能概要
STAR.EXE	STAR	STARシステムのプログラム
STAR.HLP	STAR	STARのヘルプファイル
TCALCSV.EXE	WinSys	表計算用のOLEオートメーションのサーバプログラム
INTPSV.EXE	WinSys	補間処理用のOLEオートメーションのサーバプログラム
ACAD.LSP	ACAD	デジタイザ入力システムのLISPファイル
ACAD.MNU	ACAD	デジタイザ入力システムのメニュー/タブレット定義ファイル
ATTTEMP.TXT	ACAD	デジタイザ入力によるき裂データのフィルタ設定用シート
BASEBLK.DWG	ACAD	き裂属性入力用の図面
\$SHAPE.DXF	作業	AutoSketch用の形状、き裂、評価ライン、評価断面データ
\$SHAPE.MCR	作業	AutoSketch用のマクロファイル
VB5JP.DLL	WinSys	システムファイル
MSVBVM50.DLL	WinSys.	システムファイル
STDOLE2.TLB	WinSys	システムファイル
OLEAUT32.DLL	WinSys.	システムファイル
OLEPRO32.DLL	WinSys	システムファイル
ASYCFILT.DLL	WinSys.	システムファイル
CTL3D32.DLL	WinSys	システムファイル
COMCAT.DLL	WinSys	システムファイル
COMDLG32.OCX	WinSys	システムファイル
CMDLGJP.DLL	WinSys	システムファイル
TABCTL32.OCX	WinSys	システムファイル
TABCTJP.DLL	WinSys	システムファイル
BDLIST32.OCX	WinSys	システムファイル
DBLSTJP.DLL	WinSys	システムファイル
DBGRID32.OCX	WinSys	システムファイル
DBGRDJP.DLL	WinSys	システムファイル
DAO2535.TLB	DAO	システムファイル
DAO350.DLL	DAO	システムファイル
MSJTER35.DLL	WinSys.	システムファイル
MSJTNT35.DLL	WinSys.	システムファイル
MSVCRT40.DLL	WinSys.	システムファイル
GRID32.OCX	WinSys.	システムファイル
MFC40LOC.DLL	WinSys.	システムファイル
MFC40.DLL	WinSys.	システムファイル
MCI32.OCX	WinSys.	システムファイル
MCIJP.DLL	WinSys.	システムファイル
MSOUTL32.OCX	WinSys.	システムファイル
SPIN32.OCX	WinSys.	システムファイル
MSRDC20.OCX	WinSys.	システムファイル
RDC20JP.DLL	WinSys	システムファイル
MSRDO20.DLL	WinSys	システムファイル
RDO20JP.DLL	WinSys	システムファイル
RDOCURS.DLL	WinSys	システムファイル
ADME.DLL	WinSys.	システムファイル
DTCCM.DLL	WinSys.	システムファイル
DTCTRACE.DLL	WinSys.	システムファイル
DTCUTIL.DLL	WinSys.	システムファイル
MSDTCPRX.DLL	WinSys.	システムファイル
XOLEHLP.DLL	WinSys.	システムファイル
DAMAGE96.OCX	WinSys.	損傷プログラムのOLEコントロール

MSJET35.DLL	WinSys.	システムファイル
VBAJET32.DLL	WinSys.	システムファイル
VB5DB.DLL	WinSys.	システムファイル
MSRD2X35.DLL	WinSys.	システムファイル
MSREPL35.DLL	WinSys.	システムファイル
ODBCJT32.DLL	WinSys.	システムファイル
ODBCJI32.DLL	WinSys.	システムファイル
ODBCTL32.DLL	WinSys.	システムファイル
MSEXCL35.DLL	WinSys.	システムファイル
MSTEXT35.DLL	WinSys.	システムファイル

1.2.2 サーバー構成

STAR システム・サーバーでは種々のデータは一括して SQL Server 上の※データベースファイル（参考文献 SQLServer を参照のこと）に格納され、き裂・解析等のデータはそれぞれのテーブルに保存されている。バージョン 5 までは、試験部分ごとにファイルを分けて管理していたが、現在はすべての試験部分のデータをまとめてテーブルに格納している。たとえば、すべての試験部分のき裂データはデータベースの中の[き裂]テーブルに格納される。また、各データの管理はデータベースの中にある[管理テーブル]テーブルが行っている。以下に STAR システムで使用するテーブルの一覧を示す。

Table 1.2 Table of STAR Sysytem

テーブル名	内容
管理テーブル	登録されている試験部位等を管理するテーブル
解析ケーステーブル	登録されている解析ケースを管理するテーブル
評価法テーブル	登録されている損傷評価法を管理するテーブル
材料テーブル	登録されている材料の種類を管理するテーブル
き裂	き裂データのテーブル
試験条件	試験条件データのテーブル
温度	温度データのテーブル
形状	形状データのテーブル
解析	解析データのテーブル
時点	解析結果の各時点のデータのテーブル
表計算	表計算データのテーブル
(各種損傷値)	各種損傷値データのテーブル
き裂_S	統計処理用き裂データのテーブル
解析_S	統計処理用解析データのテーブル
時点_S	統計処理用時点データのテーブル
表計算_S	統計処理用表計算データのテーブル
(各種損傷値)_S	統計処理用各種損傷値データのテーブル

損傷値の場合は、各評価法ごとに複数のテーブルが存在する(例えば、TTS-DS テーブル、BDS テーブル等)。また、"_S"とあるテーブルはもとのテーブルから統計処理に必要な行を抽出したデータであることを示している。

1.3 ネットワークへの対応

STAR はバージョン 3 よりネットワークに対応している。いままで大洗工学センターにおいて実施された実験・数値解析のデータを SQL Server 上のデータベース上に置くことによって、ネットワークに接続された各ユーザから同時に STAR のデータベースを検索することが可能である。

このネットワーク対応により、STAR を使用するユーザを 2 階層に区分している。一つは「管理

者」モードで、データの入力/更新、検索など STAR の持つすべてのコマンドが実行できるモードである。もう一方は「一般ユーザ」モードで、データの閲覧、検索のみが可能である。以下にユーザ別のコマンド一覧表を示す。一般ユーザには使用できないコマンドは、表中で網掛け表示されている。

Table 1.3 STAR Commnads

	管理者	一般ユーザ
データ選択	試験部分選択 解析ケース選択 管理データ選択内容確認 試験部分形状図 STARの終了	試験部分選択 解析ケース選択 管理データ選択内容確認 試験部分形状図 STARの終了
実験データ	き裂データ入力 き裂データ確認 き裂補間処理 試験条件入力 試験条件確認 温度データ入力 温度データ確認 形状データ入力 形状データ確認	き裂データ入力 き裂データ確認 き裂補間処理 試験条件入力 試験条件確認 温度データ入力 温度データ確認 形状データ入力 形状データ確認
解析データ	解析結果入力 解析結果確認 表計算/全テーブル 表計算/現試験体 表計算/現解析ケース 表計算確認 損傷評価法選択 損傷値/全テーブル 損傷値/現試験体 損傷値/現解析ケース 損傷値確認 Qの計算 解析情報入力 解析情報確認	解析結果入力 解析結果確認 表計算/全テーブル 表計算/現試験体 表計算/現解析ケース 表計算確認 損傷評価法選択 損傷値/全テーブル 損傷値/現試験体 損傷値/現解析ケース 損傷値確認 Qの計算 解析情報入力 解析情報確認
分布比較	分布比較実行	分布比較実行
統計	き裂データ抽出 補間処理/全テーブル 補間処理/現試験体 補間処理/現解析ケース 統計用データ確認 統計処理実行	き裂データ抽出 補間処理/全テーブル 補間処理/現試験体 補間処理/現解析ケース 統計用データ確認 統計処理実行
STAR制御	管理データの一覧 管理データの登録 解析ケースの登録 損傷評価法の登録 材料の登録 環境設定 SQL STARの終了	管理データの一覧 管理データの登録 解析ケースの登録 損傷評価法の登録 材料の登録 環境設定 SQL STARの終了

STAR システムを使用するユーザが「管理者」あるいは「一般ユーザ」のどちらのモードになるのかは、SQL Server にログインする際に決まる。ユーザが STAR データベースに対してデータベースの更新を行うことができる権限を持っている場合は、「管理者」モードで STAR が起動され、STAR データベースの閲覧の権限しか持たない場合は「一般ユーザ」モードで起動する。

2. STARデータベースの構造

STAR システムでは種々のデータは SQL Server 上に格納され、き裂・解析等のデータはそれぞれのテーブルに分割されている。バージョン 5 までは、試験部分ごとに分けて EXCEL のワークシートで管理していたが、現在はすべての試験部分のデータをまとめてリレーショナルデータベースのテーブルに格納している。たとえば、すべての試験部分のき裂データは SQL Server の中の[き裂]テーブルに格納される。また、各データの管理は SQL Server の中にある[管理データテーブル]テーブルが行っている。[管理データテーブル]テーブルと[き裂]テーブルや[解析]テーブル等の各データテーブルとのリンクは、それぞれのテーブルに[参照 id]というフィールドを設けて、連結している。以下に STAR で使用するテーブルを示す。

Table 2.1 Table Contents

テーブル名	内容
管理データテーブル	登録されている試験部位等を管理するテーブル
解析ケーステーブル	登録されている解析ケースを管理するテーブル
評価法テーブル	登録されている損傷評価法を管理するテーブル
材料テーブル	登録されている材料の種類を管理するテーブル
き裂	き裂データのテーブル
試験条件	試験条件データのテーブル
温度	温度データのテーブル
形状	形状データのテーブル
解析	解析データのテーブル
時点	解析結果の各時点のデータのテーブル
表計算	表計算データのテーブル
(各種損傷値)	各種損傷値データのテーブル
き裂_S	統計処理用き裂データのテーブル
解析_S	統計処理用解析データのテーブル
時点_S	統計処理用時点データのテーブル
表計算_S	統計処理用表計算データのテーブル
(各種損傷値)_S	統計処理用各種損傷値データのテーブル

2.1 テーブル定義

以下に各テーブルのフィールド名とその型を示す。

◆ [管理データテーブル] テーブル

Table 2.2 Structure of Management Data Table

番号	フィールド名	データ型	備考
1	id	長整数(4)	データを一意に識別するためのID
2	試験装置	文字列(50)	試験装置の名称
3	試験体	文字列(50)	試験体の名称
4	試験部分	文字列(50)	試験部分の名称
5	評価ライン数	長整数(4)	評価ラインの数1か2か3
6	参照id	長整数(4)	他のテーブルとリレーションをとるためのフィールド
7	解析ケース	文字列(50)	解析ケース名

8	評価法id	長整数(4)	損傷評価法の番号
9	き裂id	長整数(4)	き裂テーブルとリレーションととるためのフィールド
10	試験条件id	長整数(4)	試験条件テーブルとリレーションととるためのフィールド
11	温度id	長整数(4)	温度テーブルとリレーションととるためのフィールド
12	形状id	長整数(4)	形状テーブルとリレーションととるためのフィールド
13	解析id	長整数(4)	解析テーブルとリレーションととるためのフィールド
14	表計算id	長整数(4)	表計算テーブルとリレーションととるためのフィールド
15	損傷値id	長整数(4)	各種損傷値テーブルとリレーションととるためのフィールド
16	き裂Sid	長整数(4)	き裂_Sテーブルとリレーションととるためのフィールド
17	解析Sid	長整数(4)	解析_Sテーブルとリレーションととるためのフィールド
18	表計算Sid	長整数(4)	表計算テーブルとリレーションととるためのフィールド
19	損傷値Sid	長整数(4)	各種損傷値_Sテーブルとリレーションととるためのフィールド
20	形状DXF	文字列(127)	形状データのDXFファイルのファイル名
21	FEメッシュ	文字列(127)	FEMAPデータのファイル名
22	FINAS入力	文字列(127)	FINAS入力データがあるディレクトリ名
23	Range1	文字列(127)	評価ライン1のRANGEデータのファイル名
24	Range2	文字列(127)	評価ライン2のRANGEデータのファイル名
25	Range3	文字列(127)	評価ライン3のRANGEデータのファイル名
26	FINAS出力	文字列(127)	FUNASの出力ファイルのファイル名
27	FINAS説明	文字列(127)	その他コメント

◆ [解析ケーステーブル] テーブル

Table 2.3 Structure of Analysis Case Table

番号	フィールド名	データ型	備考
1	id	長整数(4)	データを一意に識別するためのID
2	解析ケース	文字列(50)	解析ケースの名称
3	流体温度	文字列(50)	実験値か設計値か
4	熱伝達係数	文字列(50)	実験値か設計値か
5	メッシュモデル	文字列(50)	詳細形状か設計形状か
6	解析法	文字列(50)	解析法の名称

◆ [評価法テーブル] テーブル

Table 2.4 Structure of Evaluation Method Table

番号	フィールド名	データ型	備考
1	id	長整数(4)	データを一意に識別するためのID
2	評価法名	文字列(50)	評価法名
3	計算実施	論理型	損傷値計算でその評価法を計算する場合は真に設定
4	備考	文字列(80)	その他コメント
5	S p	倍精度実数(8)	表面応力
6	S n	倍精度実数(8)	等価線形応力
7	S i	倍精度実数(8)	初期応力
8	K	倍精度実数(8)	応力集中係数
9	K e d	倍精度実数(8)	K e の係数
10	K e	倍精度実数(8)	歪集中係数 K e
11	K e d g	倍精度実数(8)	総体的歪集中係数
12	K e d l	倍精度実数(8)	局所的歪集中係数
13	K e p	倍精度実数(8)	歪集中係数 K e p
14	E f	倍精度実数(8)	熱ピーク歪
15	E t	倍精度実数(8)	全歪範囲
16	Q n	倍精度実数(8)	弾性追従係数

17	Q w	倍精度実数(8)	溶接金属に対する弾性追従係数
18	Q l	倍精度実数(8)	局所的弾性追従係数
19	Q n u	倍精度実数(8)	多軸効果を表わす弾性追従係数
20	Q e f f	倍精度実数(8)	弾性追従係数 Q e f f
21	Q f	倍精度実数(8)	弾性追従係数 Q f
22	Q b m	倍精度実数(8)	母材に対する弾性追従係数
23	Neuber	倍精度実数(8)	NEUBER則
24	D c	倍精度実数(8)	疲労損傷値
25	D f	倍精度実数(8)	クリープ損傷値
26	D	倍精度実数(8)	クリープ疲労損傷値
27	D c r	倍精度実数(8)	許容値

◆ [材料テーブル] テーブル

Table 2.5 Structure of Materials Data Table

番号	フィールド名	データ型	備考
1	id	長整数(4)	データを一意に識別するためのID
2	材料名	文字列(50)	材料名
3	備考	文字列(80)	

◆ [き裂] および [き裂_S] テーブル

Table 2.6 Structure of Crack & Crack Supplement Data Table

番号	フィールド名	データ型	備考
1	id	長整数(4)	データを一意に識別するためのID
2	参照id	長整数(4)	管理データテーブルから参照されるID
3	データ番号	長整数(4)	解析テーブルを示すID
4	日付	日付	解析テーブルを示すID
5	ライン番号	長整数(4)	評価ラインの番号
6	起点r1	倍精度実数(8)	き裂の起点の r 座標
7	起点z1	倍精度実数(8)	き裂の起点の z 座標
8	起点θ1	倍精度実数(8)	き裂の起点の θ 座標
9	終点r2	倍精度実数(8)	き裂の終点の r 座標
10	終点z2	倍精度実数(8)	き裂の終点の z 座標
11	終点θ2	倍精度実数(8)	き裂の終点の θ 座標
12	き裂長さ	倍精度実数(8)	き裂の長さ
13	母・溶・HAZ	文字列(50)	母材か溶金かHAZか
14	溶接法	文字列(50)	
15	余盛有無	文字列(50)	
16	余盛形状	文字列(50)	
17	溶接備考	文字列(50)	
18	粒内・粒界	文字列(50)	
19	き裂形状	文字列(50)	
20	き裂発生サイクル	文字列(50)	
21	ストライエーション	文字列(50)	
22	ビッチマーク	文字列(50)	
23	統計処理アクティブ	長整数(4)	[き裂データ抽出]コマンドでは、このフィールドが1の値を抽出して、[き裂_S]テーブルに格納
24	結晶粒度	文字列(50)	
25	備考	文字列(127)	
26	L	倍精度実数(8)	評価ラインに沿った長さ[き裂補間処理]コマンドで計算

◆ [試験条件] テーブル

Table 2.7 Structure of Test Condition Table

番号	フィールド名	データ型	備考
1	id	長整数(4)	データを一意に識別するためのID
2	参照id	長整数(4)	管理データテーブルから参照されるID
3	試験装置	文字列(50)	
4	試験体	文字列(50)	
5	材料id	長整数(4)	材料名を示すID
6	評価部位	文字列(50)	
7	COLD温度	倍精度実数(8)	
8	HOT温度	倍精度実数(8)	
9	COLD保持時間	倍精度実数(8)	
10	HOT保持時間	倍精度実数(8)	
11	機械荷重	倍精度実数(8)	
12	サイクル数	長整数(4)	
13	データ数	長整数(4)	
14	関連資料名	文字列(127)	
15	関連資料番号	文字列(127)	
16	関連データセット	文字列(127)	
17	SMAT_TAG_NO	文字列(127)	
18	対	文字列(255)	

◆ [温度] テーブル

Table 2.8 Structure of Temperature Data Table

番号	フィールド名	データ型	備考
1	id	長整数(4)	データを一意に識別するためのID
2	参照id	長整数(4)	管理データテーブルから参照されるID
3	TIME1	倍精度実数(8)	
4	TEMP1	倍精度実数(8)	
5	CONV1	倍精度実数(8)	
6	TIME2	倍精度実数(8)	
7	TEMP2	倍精度実数(8)	
8	CONV2	倍精度実数(8)	
9	TIME3	倍精度実数(8)	
10	TEMP3	倍精度実数(8)	
11	CONV3	倍精度実数(8)	
12	TIME4	倍精度実数(8)	
13	TEMP4	倍精度実数(8)	
14	CONV4	倍精度実数(8)	
15	TIME5	倍精度実数(8)	
16	TEMP5	倍精度実数(8)	
17	CONV5	倍精度実数(8)	
18	TIME6	倍精度実数(8)	
19	TEMP6	倍精度実数(8)	
20	CONV6	倍精度実数(8)	
21	TIME7	倍精度実数(8)	
22	TEMP7	倍精度実数(8)	
23	CONV7	倍精度実数(8)	
24	TIME8	倍精度実数(8)	
25	TEMP8	倍精度実数(8)	
26	CONV8	倍精度実数(8)	
27	TIME9	倍精度実数(8)	
28	TEMP9	倍精度実数(8)	
29	CONV9	倍精度実数(8)	
30	TIME10	倍精度実数(8)	
31	TEMP10	倍精度実数(8)	
32	CONV10	倍精度実数(8)	

◆ [形状] テーブル

Table 2.9 Structure of Shape Data Table

番号	フィールド名	データ型	備考
1	id	長整数(4)	データを一意に識別するためのID
2	参照id	長整数(4)	管理データテーブルから参照されるID
3	起点r1	倍精度実数(8)	線分の起点の r 座標
4	起点z1	倍精度実数(8)	線分の起点の z 座標
5	終点r2	倍精度実数(8)	線分の終点の r 座標
6	終点z2	倍精度実数(8)	線分の終点の z 座標

◆ [解析] および [解析_§] テーブル

Table 2.10 Structure of Analysis & Analysis Supplement Data Table

番号	フィールド名	データ型	備考
1	id	長整数(4)	データを一意に識別するためのID
2	参照id	長整数(4)	管理データベースから参照されるID
3	解析id	長整数(4)	管理データベースから参照されるID
4	日付	日付	データを入力した日付
5	解析ケース	文字列(50)	解析ケース名
6	ライン番号	長整数(4)	評価ラインの番号
7	L	倍精度実数(8)	評価ラインに沿った長さ
8	Q	倍精度実数(8)	Q値[Qの計算]モードで計算
9	起点r1	倍精度実数(8)	評価断面の起点のr座標
10	起点z1	倍精度実数(8)	評価断面の起点のz座標
11	起点 θ 1	倍精度実数(8)	評価断面の起点の θ 座標
12	終点r2	倍精度実数(8)	評価断面の終点のr座標
13	終点z2	倍精度実数(8)	評価断面の終点のz座標
14	終点 θ 2	倍精度実数(8)	評価断面の終点の θ 座標
15	TEMPMAX	倍精度実数(8)	最高金属温度
16	時点1id	長整数(4)	時点1のデータを示すID
17	時点2id	長整数(4)	時点1のデータを示すID
18	試験条件id	長整数(4)	試験条件データを示すID

◆ [時点] および [時点_§] テーブル

Table 2.11 Structure of Point in Time & Point in Time Supplement Data Table

番号	フィールド名	データ型	備考
1	id	長整数(4)	解析の時点1id,時点2idとリンクされる
2	参照id	長整数(4)	管理データベースから参照されるID
3	解析id	長整数(4)	管理データベースから参照されるID
4	解析ケース	文字列(50)	解析ケース名
5	評価時点	倍精度実数(8)	
6	TEMP	倍精度実数(8)	
7	LTEMP	倍精度実数(8)	
8	MTEMP	倍精度実数(8)	
9	σ_r	倍精度実数(8)	
10	σ_z	倍精度実数(8)	
11	σ_θ	倍精度実数(8)	
12	τ_{rz}	倍精度実数(8)	
13	L σ_r	倍精度実数(8)	
14	L σ_z	倍精度実数(8)	
15	L σ_θ	倍精度実数(8)	
16	L τ_{rz}	倍精度実数(8)	
17	M σ_r	倍精度実数(8)	
18	M σ_z	倍精度実数(8)	
19	M σ_θ	倍精度実数(8)	
20	M τ_{rz}	倍精度実数(8)	
21	VMS	倍精度実数(8)	
22	TRES	倍精度実数(8)	
23	ϵ_r	倍精度実数(8)	
24	ϵ_z	倍精度実数(8)	
25	ϵ_θ	倍精度実数(8)	

26	γrz	倍精度実数(8)	
27	εpr	倍精度実数(8)	
28	εpz	倍精度実数(8)	
29	$\varepsilon p\theta$	倍精度実数(8)	
30	γprz	倍精度実数(8)	
31	εcr	倍精度実数(8)	
32	εcz	倍精度実数(8)	
33	$\varepsilon c\theta$	倍精度実数(8)	
34	γcrz	倍精度実数(8)	

◆ [表計算] および [表計算_S] テーブル

Table 2.12 Structure of Calculation & Calculation Supplement Data Table

番号	フィールド名	データ型	備考
1	id	長整数(4)	データを一意に識別するためのID
2	参照id	長整数(4)	管理データベースから参照されるID
3	解析id	長整数(4)	管理データベースから参照されるID
4	日付	日付	日付
5	解析ケース	文字列(50)	解析ケース名
6	ライン番号	長整数(4)	評価ラインの番号
7	L	倍精度実数(8)	評価ラインに沿った長さ
8	起点rl	倍精度実数(8)	評価断面の起点の r 座標
9	起点zl	倍精度実数(8)	評価断面の起点の z 座標
10	起点 θ 1	倍精度実数(8)	評価断面の起点の θ 座標
11	d σr	倍精度実数(8)	
12	d σz	倍精度実数(8)	
13	d $\sigma \theta$	倍精度実数(8)	
14	d τrz	倍精度実数(8)	
15	dL σr	倍精度実数(8)	
16	dL σz	倍精度実数(8)	
17	dL $\sigma \theta$	倍精度実数(8)	
18	dL τrz	倍精度実数(8)	
19	dM σr	倍精度実数(8)	
20	dM σz	倍精度実数(8)	
21	dM $\sigma \theta$	倍精度実数(8)	
22	dM τrz	倍精度実数(8)	
23	d σ 1	倍精度実数(8)	
24	d σ 2	倍精度実数(8)	
25	d σ 3	倍精度実数(8)	
26	dL σ 1	倍精度実数(8)	
27	dL σ 2	倍精度実数(8)	
28	dL σ 3	倍精度実数(8)	
29	dM σ 1	倍精度実数(8)	
30	dM σ 2	倍精度実数(8)	
31	dM σ 3	倍精度実数(8)	
32	$\Delta \sigma r$	倍精度実数(8)	
33	$\Delta \sigma z$	倍精度実数(8)	
34	$\Delta \sigma \theta$	倍精度実数(8)	
35	$\Delta \tau rz$	倍精度実数(8)	
36	$\Delta L\sigma r$	倍精度実数(8)	
37	$\Delta L\sigma z$	倍精度実数(8)	
38	$\Delta L\sigma \theta$	倍精度実数(8)	

39	$\Delta L\tau rz$	倍精度実数(8)	
40	$\Delta M\sigma r$	倍精度実数(8)	
41	$\Delta M\sigma z$	倍精度実数(8)	
42	$\Delta M\sigma\theta$	倍精度実数(8)	
43	$\Delta M\tau rz$	倍精度実数(8)	
44	$\Delta\sigma 1$	倍精度実数(8)	
45	$\Delta\sigma 2$	倍精度実数(8)	
46	$\Delta\sigma 3$	倍精度実数(8)	
47	$\Delta L\sigma 1$	倍精度実数(8)	
48	$\Delta L\sigma 2$	倍精度実数(8)	
49	$\Delta L\sigma 3$	倍精度実数(8)	
50	$\Delta M\sigma 1$	倍精度実数(8)	
51	$\Delta M\sigma 2$	倍精度実数(8)	
52	$\Delta M\sigma 3$	倍精度実数(8)	
53	ΔVMS	倍精度実数(8)	
54	$\Delta TRES$	倍精度実数(8)	
55	$\Delta RANK$	倍精度実数(8)	
56	$\Delta LVMS$	倍精度実数(8)	
57	$\Delta LTRES$	倍精度実数(8)	
58	$\Delta LRANK$	倍精度実数(8)	
59	$\Delta MVMS$	倍精度実数(8)	
60	$\Delta MTRES$	倍精度実数(8)	
61	$\Delta MRANK$	倍精度実数(8)	
62	$d\epsilon r$	倍精度実数(8)	
63	$d\epsilon z$	倍精度実数(8)	
64	$d\epsilon\theta$	倍精度実数(8)	
65	$d\gamma rz$	倍精度実数(8)	
66	$d\epsilon pr$	倍精度実数(8)	
67	$d\epsilon pz$	倍精度実数(8)	
68	$d\epsilon p\theta$	倍精度実数(8)	
69	$d\gamma prz$	倍精度実数(8)	
70	$d\epsilon cr$	倍精度実数(8)	
71	$d\epsilon cz$	倍精度実数(8)	
72	$d\epsilon c\theta$	倍精度実数(8)	
73	$d\gamma crz$	倍精度実数(8)	
74	$\Delta EEPCEQ$	倍精度実数(8)	
75	$\Delta EPEQ$	倍精度実数(8)	
76	$\Delta ECEQ$	倍精度実数(8)	

◆ [TTSDS] および [TTSDS_S] テーブル

Table 2.13 Structure of TTSDS Method Table

番号	フィールド名	データ型	備考
1	id	長整数(4)	データを一意に識別するためのID
2	参照id	長整数(4)	管理データテーブルから参照されるID
3	解析id	長整数(4)	解析テーブルを示すID
4	試験条件id	長整数(4)	解析テーブルを示すID
5	解析ケース	文字列(50)	解析ケース名
6	ライン番号	長整数(4)	評価ラインの番号
7	L	倍精度実数(8)	評価ラインに沿った長さ
8	起点r	倍精度実数(8)	評価断面の起点の r 座標
9	起点z	倍精度実数(8)	評価断面の起点の z 座標

10	起点 θ	倍精度実数(8)	評価断面の起点の θ 座標
11	VMS	倍精度実数(8)	(未使用)
12	E E P C E Q	倍精度実数(8)	(未使用)
13	K e	倍精度実数(8)	歪集中係数
14	E t	倍精度実数(8)	全歪範囲
15	D f	倍精度実数(8)	疲労損傷値
16	D c	倍精度実数(8)	クリープ損傷値
17	D	倍精度実数(8)	クリープ疲労損傷値

◆ [BDS] および [BDS_S] テーブル

Table 2.14 Structure of BDS Method Table

番号	フィールド名	データ型	備考
1	id	長整数(4)	データを一意に識別するためのID
2	参照id	長整数(4)	管理データベースから参照されるID
3	解析id	長整数(4)	解析テーブルを示すID
4	試験条件id	長整数(4)	解析テーブルを示すID
5	解析ケース	文字列(50)	解析ケース名
6	ライン番号	長整数(4)	評価ラインの番号
7	L	倍精度実数(8)	評価ラインに沿った長さ
8	起点r	倍精度実数(8)	評価断面の起点のr座標
9	起点z	倍精度実数(8)	評価断面の起点のz座標
10	起点 θ	倍精度実数(8)	評価断面の起点の θ 座標
11	S p	倍精度実数(8)	表面応力
12	S n	倍精度実数(8)	等価線形応力
13	K	倍精度実数(8)	応力集中係数
14	Q n	倍精度実数(8)	弾性追従係数
15	K e d	倍精度実数(8)	K e の係数
16	Neuber	倍精度実数(8)	NEUBER則
17	K e	倍精度実数(8)	歪集中係数
18	E f	倍精度実数(8)	熱ピーク歪
19	E t	倍精度実数(8)	全歪範囲
20	D f	倍精度実数(8)	疲労損傷値
21	D c	倍精度実数(8)	クリープ損傷値
22	D	倍精度実数(8)	クリープ疲労損傷値
23	D c r	倍精度実数(8)	許容値

◆ [BDSMD] および [BDSMD_S] テーブル

Table 2.15 Structure of BDSMD Method Table

番号	フィールド名	データ型	備考
1	id	長整数(4)	データを一意に識別するためのID
2	参照id	長整数(4)	管理データベースから参照されるID
3	解析id	長整数(4)	解析テーブルを示すID
4	試験条件id	長整数(4)	解析テーブルを示すID
5	解析ケース	文字列(50)	解析ケース名
6	ライン番号	長整数(4)	評価ラインの番号
7	L	倍精度実数(8)	評価ラインに沿った長さ
8	起点r	倍精度実数(8)	評価断面の起点のr座標
9	起点z	倍精度実数(8)	評価断面の起点のz座標
10	起点 θ	倍精度実数(8)	評価断面の起点の θ 座標
11	S p	倍精度実数(8)	表面応力

12	S n	倍精度実数(8)	等価線形応力
13	K	倍精度実数(8)	応力集中係数
14	Q n	倍精度実数(8)	弾性追従係数
15	K e d	倍精度実数(8)	K e の係数
16	Neuber	倍精度実数(8)	NEUBER則
17	K e	倍精度実数(8)	歪集中係数
18	E f	倍精度実数(8)	熱ピーク歪
19	E t	倍精度実数(8)	全歪範囲
20	S i	倍精度実数(8)	初期応力
21	Q f	倍精度実数(8)	弾性追従係数
22	D f	倍精度実数(8)	疲労損傷値
23	D c	倍精度実数(8)	クリープ損傷値
24	D	倍精度実数(8)	クリープ疲労損傷値
25	D c r	倍精度実数(8)	許容値

◆ [DDS] および [DDS_S] テーブル

Table 2.16 Structure of DDS Method Table

番号	フィールド名	データ型	備考
1	id	長整数(4)	データを一意に識別するためのID
2	参照id	長整数(4)	管理データテーブルから参照されるID
3	解析id	長整数(4)	解析テーブルを示すID
4	試験条件id	長整数(4)	解析テーブルを示すID
5	解析ケース	文字列(50)	解析ケース名
6	ライン番号	長整数(4)	評価ラインの番号
7	L	倍精度実数(8)	評価ラインに沿った長さ
8	起点r	倍精度実数(8)	評価断面の起点の r 座標
9	起点z	倍精度実数(8)	評価断面の起点の z 座標
10	起点θ	倍精度実数(8)	評価断面の起点の θ 座標
11	S p	倍精度実数(8)	表面応力
12	S n	倍精度実数(8)	等価線形応力
13	K	倍精度実数(8)	応力集中係数
14	Q n	倍精度実数(8)	弾性追従係数
15	K e d	倍精度実数(8)	K e の係数
16	Neuber	倍精度実数(8)	NEUBER則
17	K e	倍精度実数(8)	歪集中係数
18	E f	倍精度実数(8)	熱ピーク歪
19	E t	倍精度実数(8)	全歪範囲
20	S i	倍精度実数(8)	初期応力
21	Q f	倍精度実数(8)	弾性追従係数
22	D f	倍精度実数(8)	疲労損傷値
23	D c	倍精度実数(8)	クリープ損傷値
24	D	倍精度実数(8)	クリープ疲労損傷値

◆ [DDSWELD] および [DDSWELD_S] テーブル

Table 2.17 Structure of DDSWELD Method Table

番号	フィールド名	データ型	備考
1	id	長整数(4)	データを一意に識別するためのID
2	参照id	長整数(4)	管理データテーブルから参照されるID
3	解析id	長整数(4)	解析テーブルを示すID
4	試験条件id	長整数(4)	解析テーブルを示すID

5	解析ケース	文字列(50)	解析ケース名
6	ライン番号	長整数(4)	評価ラインの番号
7	L	倍精度実数(8)	評価ラインに沿った長さ
8	起点r	倍精度実数(8)	評価断面の起点の r 座標
9	起点z	倍精度実数(8)	評価断面の起点の z 座標
10	起点 θ	倍精度実数(8)	評価断面の起点の θ 座標
11	S p	倍精度実数(8)	表面応力
12	S n	倍精度実数(8)	等価線形応力
13	K	倍精度実数(8)	応力集中係数
14	K e d	倍精度実数(8)	K e の係数
15	Neuber	倍精度実数(8)	NEUBER則
16	K e	倍精度実数(8)	歪集中係数
17	E f	倍精度実数(8)	熱ピーク歪
18	E t	倍精度実数(8)	全歪範囲
19	S i	倍精度実数(8)	初期応力
20	Q b m	倍精度実数(8)	母材に対する弾性追従係数
21	Q w	倍精度実数(8)	溶接金属に対する弾性追従係数
22	Q e f f	倍精度実数(8)	弾性追従係数
23	D f	倍精度実数(8)	疲労損傷値
24	D c	倍精度実数(8)	クリープ損傷値
25	D	倍精度実数(8)	クリープ疲労損傷値

◆ [EQEF] および [EQEF_S] テーブル

Table 2.18 Structure of EQEF Method Table

番号	フィールド名	データ型	備考
1	id	長整数(4)	データを一意に識別するためのID
2	参照id	長整数(4)	管理データテーブルから参照されるID
3	解析id	長整数(4)	解析テーブルを示すID
4	試験条件id	長整数(4)	解析テーブルを示すID
5	解析ケース	文字列(50)	解析ケース名
6	ライン番号	長整数(4)	評価ラインの番号
7	L	倍精度実数(8)	評価ラインに沿った長さ
8	起点r	倍精度実数(8)	評価断面の起点の r 座標
9	起点z	倍精度実数(8)	評価断面の起点の z 座標
10	起点 θ	倍精度実数(8)	評価断面の起点の θ 座標
11	S p	倍精度実数(8)	表面応力
12	S n	倍精度実数(8)	等価線形応力
13	K	倍精度実数(8)	応力集中係数
14	K e d	倍精度実数(8)	K e の係数
15	K e	倍精度実数(8)	歪集中係数
16	K e d g	倍精度実数(8)	総体的歪集中係数
17	K e d l	倍精度実数(8)	局所的歪集中係数
18	E f	倍精度実数(8)	熱ピーク歪
19	E t	倍精度実数(8)	全歪範囲
20	S i	倍精度実数(8)	初期応力
21	Q n	倍精度実数(8)	総体的弾性追従係数
22	Q l	倍精度実数(8)	局所的弾性追従係数
23	Q n u	倍精度実数(8)	多軸効果を表わす弾性追従係数
24	Q e f f	倍精度実数(8)	弾性追従係数
25	D f	倍精度実数(8)	疲労損傷値
26	D c	倍精度実数(8)	クリープ損傷値

27	D	倍精度実数(8)	クリープ疲労損傷値
28	D c r	倍精度実数(8)	許容値

◆ [EQEFWELD] および [EQEFWELD_S] テーブル

Table 2.19 Structure of EQEFWELD Method Table

番号	フィールド名	データ型	備考
1	id	長整数(4)	データを一意に識別するためのID
2	参照id	長整数(4)	管理データベースから参照されるID
3	解析id	長整数(4)	解析テーブルを示すID
4	試験条件id	長整数(4)	解析テーブルを示すID
5	解析ケース	文字列(50)	解析ケース名
6	ライン番号	長整数(4)	評価ラインの番号
7	L	倍精度実数(8)	評価ラインに沿った長さ
8	起点r	倍精度実数(8)	評価断面の起点の r 座標
9	起点z	倍精度実数(8)	評価断面の起点の z 座標
10	起点 θ	倍精度実数(8)	評価断面の起点の θ 座標
11	S p	倍精度実数(8)	表面応力
12	S n	倍精度実数(8)	等価線形応力
13	K	倍精度実数(8)	応力集中係数
14	K e d	倍精度実数(8)	K e の係数
15	K e	倍精度実数(8)	歪集中係数
16	K e d g	倍精度実数(8)	総体的歪集中係数
17	K e d l	倍精度実数(8)	局所的歪集中係数
18	E f	倍精度実数(8)	熱ビーク歪
19	E t	倍精度実数(8)	全歪範囲
20	S i	倍精度実数(8)	初期応力
21	Q n	倍精度実数(8)	総体的弾性追従係数
22	Q l	倍精度実数(8)	局所的弾性追従係数
23	Q n u	倍精度実数(8)	多軸効果を表わす弾性追従係数
24	Q w	倍精度実数(8)	溶接金属に対する弾性追従係数
25	Q e f f	倍精度実数(8)	弾性追従係数
26	D f	倍精度実数(8)	疲労損傷値
27	D c	倍精度実数(8)	クリープ損傷値
28	D	倍精度実数(8)	クリープ疲労損傷値
29	D c r	倍精度実数(8)	許容値

◆ [EQEF15] および [EQEF15_S] テーブル

Table 2.20 Structure of EQEF15 Method Table

番号	フィールド名	データ型	備考
1	id	長整数(4)	データを一意に識別するためのID
2	参照id	長整数(4)	管理データベースから参照されるID
3	解析id	長整数(4)	解析テーブルを示すID
4	試験条件id	長整数(4)	解析テーブルを示すID
5	解析ケース	文字列(50)	解析ケース名
6	ライン番号	長整数(4)	評価ラインの番号
7	L	倍精度実数(8)	評価ラインに沿った長さ
8	起点r	倍精度実数(8)	評価断面の起点の r 座標
9	起点z	倍精度実数(8)	評価断面の起点の z 座標
10	起点 θ	倍精度実数(8)	評価断面の起点の θ 座標
11	S p	倍精度実数(8)	表面応力

12	S n	倍精度実数(8)	等価線形応力
13	K	倍精度実数(8)	応力集中係数
14	K e d	倍精度実数(8)	K e の係数
15	K e	倍精度実数(8)	歪集中係数
16	K e d g	倍精度実数(8)	総体的歪集中係数
17	K e d l	倍精度実数(8)	局所的歪集中係数
18	E f	倍精度実数(8)	熱ピーク歪
19	E t	倍精度実数(8)	全歪範囲
20	S i	倍精度実数(8)	初期応力
21	Q n	倍精度実数(8)	総体的弾性追従係数
22	Q l	倍精度実数(8)	局所的弾性追従係数
23	Q n u	倍精度実数(8)	多軸効果を表わす弾性追従係数
24	Q e f f	倍精度実数(8)	弾性追従係数
25	D f	倍精度実数(8)	疲労損傷値
26	D c	倍精度実数(8)	クリープ損傷値
27	D	倍精度実数(8)	クリープ疲労損傷値
28	D c r	倍精度実数(8)	許容値

◆ [EQEF167] および [EQEF167_S] テーブル

Table 2.21 Structure of EQEF167 Method Table

番号	フィールド名	データ型	備考
1	id	長整数(4)	データを一意に識別するためのID
2	参照id	長整数(4)	管理データテーブルから参照されるID
3	解析id	長整数(4)	解析テーブルを示すID
4	試験条件id	長整数(4)	解析テーブルを示すID
5	解析ケース	文字列(50)	解析ケース名
6	ライン番号	長整数(4)	評価ラインの番号
7	L	倍精度実数(8)	評価ラインに沿った長さ
8	起点r	倍精度実数(8)	評価断面の起点の r 座標
9	起点z	倍精度実数(8)	評価断面の起点の z 座標
10	起点 θ	倍精度実数(8)	評価断面の起点の θ 座標
11	S p	倍精度実数(8)	表面応力
12	S n	倍精度実数(8)	等価線形応力
13	K	倍精度実数(8)	応力集中係数
14	K e d	倍精度実数(8)	K e の係数
15	K e	倍精度実数(8)	歪集中係数
16	K e d g	倍精度実数(8)	総体的歪集中係数
17	K e d l	倍精度実数(8)	局所的歪集中係数
18	E f	倍精度実数(8)	熱ピーク歪
19	E t	倍精度実数(8)	全歪範囲
20	S i	倍精度実数(8)	初期応力
21	Q n	倍精度実数(8)	総体的弾性追従係数
22	Q l	倍精度実数(8)	局所的弾性追従係数
23	Q n u	倍精度実数(8)	多軸効果を表わす弾性追従係数
24	Q e f f	倍精度実数(8)	弾性追従係数
25	D f	倍精度実数(8)	疲労損傷値
26	D c	倍精度実数(8)	クリープ損傷値
27	D	倍精度実数(8)	クリープ疲労損傷値
28	D c r	倍精度実数(8)	許容値

◆ [INELAST] および [INELAST_S] テーブル

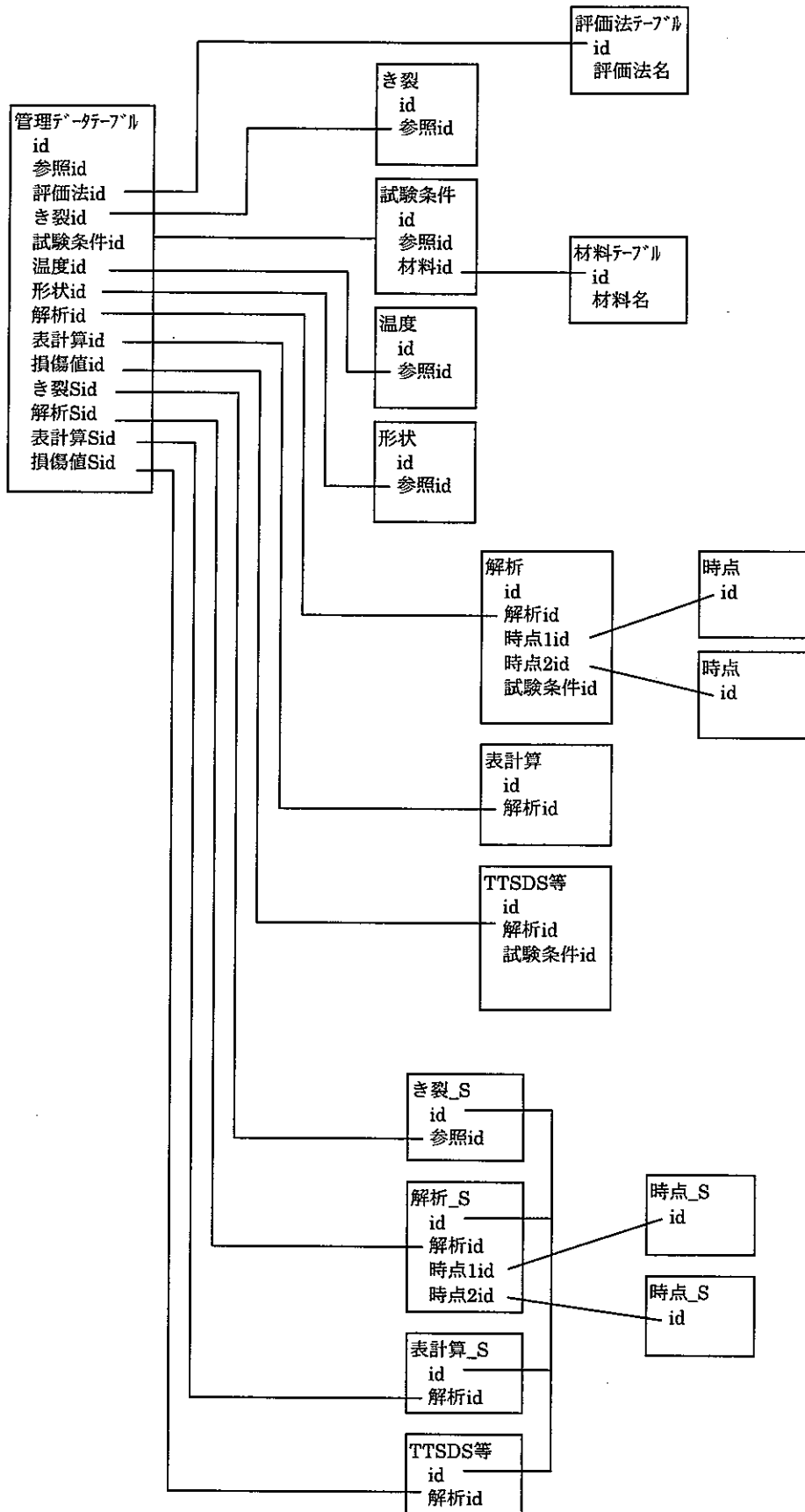
Table 2.22 Structure of INELAST Method Table

番号	フィールド名	データ型	備考
1	id	長整数(4)	データを一意に識別するためのID
2	参照id	長整数(4)	管理テーブルから参照されるID
3	解析id	長整数(4)	解析テーブルを示すID
4	試験条件id	長整数(4)	解析テーブルを示すID
5	解析ケース	文字列(50)	解析ケース名
6	ライン番号	長整数(4)	評価ラインの番号
7	L	倍精度実数(8)	評価ラインに沿った長さ
8	起点r	倍精度実数(8)	評価断面の起点の r 座標
9	起点z	倍精度実数(8)	評価断面の起点の z 座標
10	起点 θ	倍精度実数(8)	評価断面の起点の θ 座標
11	VMS	倍精度実数(8)	(未使用)
12	EEPCEQ	倍精度実数(8)	(未使用)
13	Ke	倍精度実数(8)	歪集中係数
14	Et	倍精度実数(8)	全歪範囲
15	Df	倍精度実数(8)	疲労損傷値
16	Dc	倍精度実数(8)	クリープ損傷値
17	D	倍精度実数(8)	クリープ疲労損傷値

2.2 テーブル結合

STAR の各テーブルの結合関係を示す。

Fig 2.1 Relation of STAR Table



2.3 管理データテーブル

[管理データテーブル]では、STAR にどのようなデータが現在登録されているかを管理している。以下に[管理データテーブル]の主なフィールドの詳細を説明する。

Table 2.23 Structure of Management Data Table

番号	フィールド名	備考
1	id	データを一意に識別するためのID
2	試験装置	試験装置の名称
3	試験体	試験体の名称
4	試験部分	試験部分の名称
5	評価ライン数	評価ラインの数1か2か3
6	参照id	他のテーブルとリレーションをとるためのフィールド
7	解析ケース	解析ケース名
8	評価法id	損傷評価法の番号
9	き裂id	き裂テーブルとリレーションをとるためのフィールド
10	試験条件id	試験条件テーブルとリレーションをとるためのフィールド
11	温度id	温度テーブルとリレーションをとるためのフィールド
12	形状id	形状テーブルとリレーションをとるためのフィールド
13	解析id	解析テーブルとリレーションをとるためのフィールド
14	表計算id	表計算テーブルとリレーションをとるためのフィールド
15	損傷値id	各種損傷値テーブルとリレーションをとるためのフィールド
16	き裂Sid	き裂_Sテーブルとリレーションをとるためのフィールド
17	解析Sid	解析_Sテーブルとリレーションをとるためのフィールド
18	表計算Sid	表計算テーブルとリレーションをとるためのフィールド
19	損傷値Sid	各種損傷値_Sテーブルとリレーションをとるためのフィールド

id: データを一意に識別するための番号で、ゼロからスタートする連番になっていなければならない。

試験装置: 試験装置の名称を示す。同じ試験装置を持つ行は分割されることなく1つにまとまっていなければならない。

試験体: 試験体の名称を示す。同じ試験装置および試験体を持つ行は分割されることなく1つにまとまっていなければならない。

試験部分: 試験部分の名称を示す。同じ試験装置、試験体および試験部分を持つ行は分割されることなく1つにまとまっていなければならない。

評価ライン数: 試験部分が持つ評価ラインの本数を示す。1、2または3が設定される。

参照id: 他のき裂、解析、表計算テーブル等とリレーションをとるためのフィールドである。試験部分ごとにユニークな番号が割り当てられる。STARはこの番号をもとにき裂、解析、表計算テーブルにアクセスする。

解析ケース: その試験部分に登録されている解析データのケース名を示す。試験部分を登録した直後は、ここにはハイフン(-)が入力されている。[解析結果入力]コマンドで解析データを登録すると、この行の下に新規の行が挿入され、挿入された行の[解析ケース]フィールドに新しい解析データのケースが設定される。なお、元の行の[解析ケース]フィールドはハイフンのままになっている。

評価法id: その解析ケースにおいて既に計算されている損傷値(評価法)の番号を示す。評価法の番号については、3.7.4項を参照のこと。解析データを登録した直後は、ここにはゼロ(0)が入力されている。損傷値の計算を行うと、この行の下に新規の行が挿入され、挿入された行の[評価法id]フィールドに新しく計算された評価法の番号が設定される。なお、元の行の[評価法id]フィールドはゼロのままになっている。

き裂id: 試験部分の登録直後はゼロになっているが、その試験部分のき裂データが登録されると、参照idと同じ番号が設定される。

試験条件id: 試験部分の登録直後はゼロになっているが、その試験部分の試験条件データが登録されると、参照idと同じ番号が設定される。

温度id: 試験部分の登録直後はゼロになっているが、その試験部分の温度データが登録されると、参照idと同じ番号が設定される。

形状id: 試験部分の登録直後はゼロになっているが、その試験部分の形状データが登録されると、参照idと同じ番号が設定される。

解析id: その試験部分に登録されている解析データを識別する番号が入る。試験部分を登録した直後は、ここにはゼロ(0)が入力されている。[解析結果入力]コマンドで解析データを登録すると、この行の下に新規の行が挿入され、挿入された行の[解析id]フィールドに新しい解析データを識別するためのユニークな番号が設定される。なお、元の行の[解析id]フィールドはゼロのままになっている。

表計算id: 解析データの登録直後はゼロになっているが、その解析データに対する表計算処理が実行されると、解析idと同じ番号が設定される。

損傷値id: 解析データの登録直後はゼロになっているが、その解析データに対する表計算処理が実行されると、この行の下に新規の行が挿入され、挿入された行の[損傷値id]フィールドに解析idと同じ番号が設定される。なお、元の行の[損傷値id]フィールドはゼロのままになっている。

き裂Sid: 試験部分の登録直後はゼロになっているが、その試験部分の統計用のき裂データが作成されると([き裂データ抽出])、参照idと同じ番号が設定される。

解析Sid: 解析データの登録直後はゼロになっているが、その解析データの統計用の解析データが作成されると([補間処理])、解析idと同じ番号が設定される。

表計算Sid: 解析データの登録直後はゼロになっているが、その解析データの統計用の表計算データが作成されると([補間処理])、解析idと同じ番号が設定される。

損傷値Sid: 解析データの登録直後はゼロになっているが、その解析データの統計用の損傷値データが作成されると([補間処理])、解析idと同じ番号が設定される。

3. STAR のソースモジュール

STAR クライアントで使用している Visual Basic のモジュール一覧を以下に示す。

Table 3.1 List of Source Module File

ファイル名	フォーム/モジュール名	概要
B_SELECT.FRM	frmBunSelect	分布比較において、評価ライン・検索クエリを選択
I_CND_RG.FRM	frmPriCondReg	試験条件データの登録
I_CRK_IP.FRM	frmPriCrackInt	き裂補間処理の進捗状況を表示
I_VW_DBG.FRM	frmPriViewDBG	実験データ登録時にファイル内容を一時的に DB グリッドコントロールに表示
I_VW_DXF.FRM	frmPriViewDXF	形状データ登録時に DXF ファイルの内容を一時的に表示
I_VW_GRD.FRM	frmPriViewGRD	実験データ登録時にファイル内容を一時的にグリッドコントロールに表示
N_ANL_CS.FRM	frmSecAnalCase	解析結果を格納する解析ケースを選択
N_ANL_RG.FRM	frmSecAnalReg	(現在未使用、X_SELDB3参照)
N_ANL_VW.FRM	frmSecAnalView	解析結果のファイル内容を一時的に DB グリッドコントロールに表示
N_DMG_CT.FRM	frmSecDmgCtrl	損傷値計算をする際にユーザに確認を促す画面
N_DMG_SL.FRM	frmSecDmgSel	損傷値計算を行う評価法を選択
N_DMG_VS.FRM	frmDmgViewSel	損傷値を表示する評価法を選択
N_FNS_VW.FRM	frmSecFinasView	解析情報の入力および確認
N_Q_SEL.FRM	frmSecQCalcSel	Q計算の解析ケースを選択
N_TBL_CT.FRM	frmSecTblCtrl	表計算をする際にユーザに確認を促す画面
S_BUBUN2.FRM	frmSelBubun2	これから処理を行う試験部分を選択
S_CASE2.FRM	frmSelCase2	これから処理を行う解析ケースを選択
S_VIEW.FRM	frmSelView	現在選択されている試験部分と解析ケースを表示
STAR.VBP		STAR システムのプロジェクトファイル
STAR_MDI.FRM	frmStarMain	STAR システムの親ウィンドウ
T_DAT_SL.FRM	frmTokDataSel	統計用のテーブルの確認
T_INT_CT.FRM	frmTokIntCtrl	統計補間処理の際にユーザに確認を促す画面
T_INT_SL.FRM	frmTokInterSel	統計補間処理をするテーブルを選択
T_SELECT.FRM	frmTokSelect	統計処理において、検索クエリを選択
U_CAS_AD.FRM	frmUtilCaseAdd	解析ケースを新規に登録
U_CASE.FRM	frmUtilCaseReg	解析ケースの登録・削除
U_HYK_AD.FRM	frmUtilHyokaAdd	損傷評価法を新規に登録
U_HYOKA.FRM	frmUtilHyokaReg	損傷評価法の登録・削除
U_KANRI2.FRM	frmUtilKanriReg2	管理データの登録・削除
U_KNR_A2.FRM	frmUtilKanriAdd2	管理データを新規に登録
U_MAT.FRM	frmUtilMatReg	材料データの登録・削除
U_SETUP.FRM	frmUtilSetup	STAR の環境を設定
U_SQL.FRM	frmUtilSQL	SQL 文を直接実行
U_VIEW.FRM	frmUtilKanriView	(現在未使用)
X_NORMAL.FRM	frmNormal	カレントの試験部分や解析ケースを表示する子ウィンドウ
X_RESGRD.FRM	frmResGRD	検索結果を表示する子ウィンドウ(グリッドコントロール)
X_RESOLE.FRM	frmResOLE	検索結果を表示する子ウィンドウ(OLEコントロール)
X_SELDB1.FRM	frmSelDB1	実験データのファイル名を入力
X_SELDB3.FRM	frmSelDB3	解析結果のファイル名を入力
X_USER2.FRM	frmUserCheck2	STAR のログイン画面
STAR1.BAS	modSTAR	各種のサブルーチンが収められた BASIC モジュール
STARC.BAS	modSTARC	各種の定数が定義された BASIC モジュール
DXFPREAD.BAS	modDXFPREAD	[試験部分形状図]で DXF ファイルを作成するモジュール
STAR.RTF		オンラインヘルプのソースファイル
STAR.HPJ		オンラインヘルプのプロジェクトファイル

3.1 各モジュールの概要

以下に、各モジュールの概要を示す。

◆ B_SELECT.FRM

分布比較処理の検索条件を選択する画面である。[分布比較]メニューの[分布比較実行]コマンドを実行した際に表示される。

Table 3.2 Summary of Source Program File "B_SELECT.FRM"

コントロール名	種類	概要
txtQueryDB	テキストボックス	検索クエリがあるデータベースファイル(MDB)を指定
cmdQDBChange	コマンドボタン	データベースファイル名をコンダイログを使って参照するボタン
cmbLine	コンボボックス	対象とする評価ラインを選択
cmbQuery	コンボボックス	検索クエリを選択
cmdOK	コマンドボタン	検索処理を実行検索結果を X_RESGRD.FRM(frmResGRD)または X_RESOLE.FRM(frmResOLE)に表示
cmdCancel	コマンドボタン	[分布比較実行]コマンドをキャンセル
CommonDialog1	コンダイログ	

◆ I_CND_RG.FRM

試験条件データの入力および確認の画面である。[実験データ]メニューの[試験条件入力]および[試験条件確認]コマンドを実行した際に表示される。

Table 3.3 Summary of Source Program File "I_CND_RG.FRM"

コントロール名	種類	概要
txtSochi	テキストボックス	試験装置名を入力・表示
txtTai	テキストボックス	試験体名を入力・表示
cmbMATid	コンボボックス	材料名を選択
txtBui	テキストボックス	評価部位を入力・表示
txtTCold	テキストボックス	COLD 温度を入力・表示
txtHCold	テキストボックス	COLD 保持時間を入力・表示
txtTHot	テキストボックス	HOT 温度を入力・表示
txtHHot	テキストボックス	HOT 保持時間を入力・表示
txtMload	テキストボックス	機械荷重を入力・表示
txtCycle	テキストボックス	サイクル数を入力・表示
txtNOData	テキストボックス	解析データの行数
txtRefName	テキストボックス	関連資料名を入力・表示
txtRefNo	テキストボックス	関連資料番号を入力・表示
txtRefDS	テキストボックス	関連データセットを入力・表示
txtSMAT	テキストボックス	Smat Tag No.を入力・表示
txtMemo	テキストボックス	メモを入力・表示
cmdOK	コマンドボタン	入力されたデータを[試験条件]テーブルに格納一般ユーザの場合はこのボタンは表示されない
cmdCancel	コマンドボタン	[試験条件入力]コマンドをキャンセル

◆ I_CRK_IP.FRM

き裂データの補間処理を行う際に進捗状況を表示する画面である。[実験データ]メニューの[き裂補間処理]コマンドを実行した際に表示される。

Table 3.4 Summary of Source Program File “I_CRK_IP.FRM”

コントロール名	種類	概要
lstError	リストボックス	き裂データの補間処理の進捗状況を表示
cmdClose	コマンドボタン	画面を閉ず

◆ I_VW_DBG.FRM

き裂データ、温度データ、形状データを登録する際にファイルの内容を一時的に DB グリッドコントロールを使って表示する画面である。[実験データ]メニューの[き裂データ入力]、[温度データ入力]、[形状データ入力]コマンドを実行し、ファイル名を入力した後に表示される。

Table 3.5 Summary of Source Program File “I_VW_DBG.FRM”

コントロール名	種類	概要
dbgSourceTBL	DB グリッド	dtaSourceDB の内容を表示
dtaSourceDB	データ	テーブルと連結されたデータコントロール
cmdOK	コマンド	表示されているデータを対応したテーブルに保存
cmdCancel	コマンド	何もしないでテーブルを閉ず

◆ I_VW_DXF.FRM

形状データを登録する際に DXF ファイルの内容を一時的にグリッドコントロールを使って表示する画面である。[形状データ入力]コマンドを実行し、ファイル名を入力した後に表示される。

Table 3.6 Summary of Source Program File “I_VW_DXF.FRM”

コントロール名	種類	概要
grdDXF	グリッド	データの内容を表示
cmdOK	コマンド	グリッドの内容をテーブルに保存
cmdCancel	コマンド	[形状データ入力]コマンドをキャンセルする

◆ I_VW_GRD.FRM

[き裂データ確認]、[温度データ確認]、[形状データ確認]、[解析結果確認]、[表計算確認]、[損傷値確認]コマンドを実行した際に、データをグリッドコントロールを使って表示する画面である。

Table 3.7 Summary of Source Program File "I_VW_GRD.FRM"

コントロール名	種類	概要
grdView	グリッド	データの内容を表示
mnuSave	メニュー	グリッドの内容をテーブルに保存
mnuClose	メニュー	フォームを閉じる
mnuAll	メニュー	すべてのセルを選択(反転表示)
mnuCopy	メニュー	選択されているセルの内容をクリップボードへコピー

◆ N_ANL_CS.FRM

解析結果をどの解析ケースとして登録するかを選択する画面である。[解析データ]メニューの[解析結果入力]コマンドを実行した際に表示される。

Table 3.8 Summary of Source Program File "N_ANL_CS.FRM"

コントロール名	種類	概要
lstCase	リストボックス	解析ケースの一覧を表示
cmdOK	コマンドボタン	解析結果を指定された解析ケースとして[解析]テーブルに格納
cmdCancel	コマンドボタン	画面を閉じる

◆ N_ANL_RG.FRM

(現在未使用、X_SELDB3 参照)

◆ N_ANL_VW.FRM

解析結果を登録する際にファイルの内容を一時的に DB グリッドコントロールを使って表示する画面である。[解析データ]メニューの[解析結果入力]コマンドを実行し、ファイル名を入力した後に表示される。

Table 3.9 Summary of Source Program File "N_ANL_VW.FRM"

コントロール名	種類	概要
dbgAnal1	DBグリッド	評価ライン1のデータ内容(Data1)を表示
dtaSourceDB1	データ	評価ライン1のデータと連結されたデータコントロール
dbgAnal2	DBグリッド	評価ライン2のデータ内容(Data2)を表示
dtaSourceDB2	データ	評価ライン2のデータと連結されたデータコントロール
dbgAnal3	DBグリッド	評価ライン3のデータ内容(Data3)を表示
dtaSourceDB3	データ	評価ライン3のデータと連結されたデータコントロール
cmdOK	コマンドボタン	N_ANL_CS.FRM(frmSecAnalCase)を表示
cmdCancel	コマンドボタン	何もしないで画面を閉じる

◆ N_DMG_CT.FRM

損傷値計算をする際にユーザに確認を促す画面である。[解析データ]メニューの[損傷値/全テーブル]、[損傷値/現試験体]および[損傷値/現解析ケース]コマンドを実行した際に表示される。

Table 3.10 Summary of Source Program File "N_DMG_CT.FRM"

コントロール名	種類	概要
lstTarget	リストボックス	損傷値計算の処理対象となるデータを一覧表示

cmdOK	コマンドボタン	N_DMG_SL.FRMで指定された損傷値処理を実行。実際の計算は損傷プログラム(DAMAGE95.OCX)が行う
cmdCancel	コマンドボタン	何もしないで画面を閉ず

◆ N_DMG_SL.FRM

損傷値計算を行う際にどの評価法の計算をするのかを選択する画面である。[解析データ]メニューの[損傷評価法選択]コマンドを実行した際に表示される。

Table 3.11 Summary of Source Program File “N_DMG_SL.FRM”

コントロール名	種類	概要
lstUnselected	リストボックス	未選択の評価法を一覧表示
lstSelected	リストボックス	選択の評価法を一覧表示
cmdSel	コマンドボタン	指定された評価法を選択ボックスへ移動
cmdUnsel	コマンドボタン	指定された評価法を未選択ボックスへ移動
cmdOK	コマンドボタン	選択された内容を「評価法」テーブルに格納
cmdCancel	コマンドボタン	何もしないで画面を閉ず

◆ N_DMG_VS.FRM

損傷値データを画面に表示する際にどの評価法のデータを表示させるかを選択する画面である。[解析データ]メニューの[損傷評価確認]コマンドを実行した際に表示される。

Table 3.12 Summary of Source Program File “N_DMG_VS.FRM”

コントロール名	種類	概要
lstHyoka	リストボックス	計算済みの評価法の一覧を表示
cmdOK	コマンドボタン	選択された評価法を画面に表示
cmdCancel	コマンドボタン	何もしないで画面を閉ず

◆ N_FNS_VW.FRM

FINAS 関連のデータファイル名を画面に表示する。[解析データ]メニューの[解析情報入力]および[解析情報確認]コマンドを実行した際に表示されるが、コマンドによって働きが違う。[解析情報入力]の場合は、表示されたセルの内容を更新できるが、[解析情報確認]ではできない。また、[解析情報確認]からはエクスプローラを起動することができる。

Table 3.13 Summary of Source Program File “N_FNS_VW.FRM”

コントロール名	種類	概要
grdFinas	リストボックス	FINAS 関連のファイル名の一覧を表示
cmdOK	コマンドボタン	表示されているデータ内容をテーブルに保存
cmdCancel	コマンドボタン	何もしないで画面を閉ず
cmdClose	コマンドボタン	何もしないで画面を閉ず

◆ N_Q_SEL.FRM

Qの計算をする際に弾性解析と非弾性解析のケースを選択する画面である。[解析データ]メニューの[Qの計算]コマンドを実行した際に表示される。処理結果は[解析]テーブルに格納される。

Table 3.14 Summary of Source Program File "N_Q_SEL.FRM"

コントロール名	種類	概要
cmbElas	コンボボックス	弾性解析の解析ケースの一覧を選択
cmbPlas	コンボボックス	非弾性解析の解析ケースの一覧を選択
cmbSave	コンボボックス	Qの結果を格納する解析ケースの一覧を選択
cmdOK	コマンドボタン	Qの計算を実行し、cmbSaveに示される解析ケースに結果を格納
cmdCancel	コマンドボタン	何もしないで画面を閉ず

◆ N_TBL_CT.FRM

表計算をする際にユーザに確認を促す画面である。[解析データ]メニューの[表計算/全テーブル]、[表計算/現試験体]および[表計算/現解析ケース]コマンドを実行した際に表示される。

Table 3.15 Summary of Source Program File "N_TBL_CT.FRM"

コントロール名	種類	概要
lstTarget	リストボックス	表計算の処理対象となるデータを一覧表示
cmdOK	コマンドボタン	表計算を実行し、結果を[表計算]テーブルに格納
cmdCancel	コマンドボタン	何もしないで画面を閉ず

◆ S_BUBUN2.FRM

これから処理を行う試験部分を選択する画面である。[データ選択]メニューの[試験部分選択]コマンドを実行した際に表示される。

Table 3.16 Summary of Source Program File "S_BUBUN2.FRM"

コントロール名	種類	概要
outBubun	アウトライン	試験装置・試験体・試験部分をツリー表示
cmdOK	コマンドボタン	指定された試験部分をアウトに設定
cmdCancel	コマンドボタン	何もしないで画面を閉ず

◆ S_CASE2.FRM

これから処理を行う試験部分の解析ケースを選択する画面である。[データ選択]メニューの[解析ケース選択]コマンドを実行した際に表示される。

Table 3.17 Summary of Source Program File "S_CASE2.FRM"

コントロール名	種類	概要
lstCase	リストボックス	解析ケースの一覧を表示
cmdOK	コマンドボタン	指定された解析ケースをアウトに設定
cmdCancel	コマンドボタン	何もしないで画面を閉ず

◆ S_VIEW.FRM

現在選択されている試験部分と解析ケースを表示する画面である。[データ選択]メニューの[管理データ選択内容確認]コマンドを実行した際に表示される。

Table 3.18 Summary of Source Program File "S_VIEW.FRM"

コントロール名	種類	概要
---------	----	----

txtSochi	テキストボックス	カートの試験装置名を表示
txtTai	テキストボックス	カートの試験体名を表示
txtBubun	テキストボックス	カートの試験部分名を表示
txtCase	テキストボックス	カートの解析ケース名を表示
cmdOK	コマンドボタン	画面をクリア

◆ STAR_MDI.FRM

STAR システムの親ウィンドウである。

◆ T_DAT_SL.FRM

現在選択されている試験部分と解析ケースの統計用の[き裂]、[解析]、[表計算]あるいは[損傷値]データを表示する画面である。[統計]メニューの[統計用データ確認]コマンドを実行した際に表示される。

Table 3.19 Summary of Source Program File “T_DAT_SL.FRM”

コントロール名	種類	概要
optCrk	オプションボタン	き裂データを表示させることを指示
optAnl	オプションボタン	解析データを表示させることを指示
optTbl	オプションボタン	表計算データを表示させることを指示
optDmg	オプションボタン	損傷値データを表示させることを指示
cmbDmgSel	コンボボックス	損傷値の場合は評価法を選択
cmdOK	コマンドボタン	統計用のデータを表示
cmdCancel	コマンドボタン	何もしないで、画面をクリア

◆ T_INT_CT.FRM

統計補間処理をする際にユーザに確認を促す画面である。[統計]メニューの[補間処理/全テーブル]、[補間処理/現試験体]および[補間処理/現解析ケース]コマンドを実行した際に表示される。

Table 3.20 Summary of Source Program File “T_INT_CT.FRM”

コントロール名	種類	概要
lstTarget	リストボックス	処理対象となるデータの一覧を表示
cmdOK	コマンドボタン	補間処理を実行し、統計用のテーブルに結果を格納
cmdCancel	コマンドボタン	何もしないで画面をクリア

◆ T_INT_SL.FRM

統計補間処理をする際に解析・表計算・損傷値のどれに対して処理をするかを選択する画面である。[統計]メニューの[補間処理/全テーブル]、[補間処理/現試験体]および[補間処理/現解析ケース]コマンドを実行した際に表示される。

Table 3.21 Summary of Source Program File "T_INT_SL.FRM"

コントロール名	種類	概要
optAnal	オプションボタン	補間処理対象となるテーブルを解析に設定
optTable	オプションボタン	補間処理対象となるテーブルを表計算に設定
optDamage	オプションボタン	補間処理対象となるテーブルを損傷値に設定
cmbDmgSel	コンボボックス	補間処理対象となる評価法を選択
cmdOK	コマンドボタン	T_INT_CT.FRM 画面へ移行
cmdCancel	コマンドボタン	何もしないで画面を閉ずる

◆ T_SELECT.FRM

統計処理の検索条件を選択する画面である。[統計]メニューの[統計処理実行]コマンドを実行した際に表示される。

Table 3.22 Summary of Source Program File "T_SELECT.FRM"

コントロール名	種類	概要
txtQueryDB	テキストボックス	検索クエリがあるデータベース名を指定
cmdQDBChange	コマンドボタン	データベース名をコンテキストメニューを使って参照するボタン
cmbFrom	コンボボックス	現在は未使用
cmbQuery	コンボボックス	検索クエリを選択
cmdOK	コマンドボタン	検索処理を実行検索結果を X_RESGRD.FRM(frmResGRD)あるいは X_RESOLE.FRM(frmResOLE)に表示
cmdCancel	コマンドボタン	[統計処理実行]コマンドをキャンセル
CommonDialog1	コンテキストメニュー	

◆ U_CAS_AD.FRM

解析ケースを新規に登録する画面である。[解析ケースの登録]ダイアログボックスで[追加]を選択した場合に表示される。

Table 3.23 Summary of Source Program File "U_CAS_AD.FRM"

コントロール名	種類	概要
optTempJkn	オプションボタン	流体温度を実験値に設定
optTempDgn	オプションボタン	流体温度を設計値に設定
optTrnsJkn	オプションボタン	熱伝達係数を実験値に設定
optTrnsDgn	オプションボタン	熱伝達係数を設計値に設定
optMeshDtl	オプションボタン	メッシュモデルを詳細形状に設定
optMeshDgn	オプションボタン	メッシュモデルを設計形状に設定
txtMethod	テキストボックス	解析法を入力
lstMethod	リストボックス	登録済みの解析法を一覧表示
cmdOK	コマンドボタン	指定された解析ケースを[解析ケーステーブル]に格納
cmdCancel	コマンドボタン	何もしないで画面を閉ずる

◆ U_CASE.FRM

解析ケースの登録・削除を行う画面である。[STAR 制御]メニューの[解析ケースの登録]コマンドを実行した際に表示される。

Table 3.24 Summary of Source Program File "U_CASE.FRM"

コントロール名	種類	概要
lstCase	リストボックス	登録済みの解析法を一覧表示
cmdAdd	コマンドボタン	U_CAS_AD.FRM に移行

cmdDelete	コマンドボタン	指定された解析ケースを[解析ケーステーブル]から削除。また、[管理データテーブル][解析][表計算]および各損傷値テーブルから関連するレコードを削除
cmdExit	コマンドボタン	画面を閉ず

◆ U_HYK_AD.FRM

損傷評価法を新規に登録する画面である。[損傷評価法の登録]ダイアログボックスで[追加]を選択した場合に表示される。

Table 3.25 Summary of Source Program File “U_HYK_AD.FRM”

コントロール名	種類	概要
txtName	テキストボックス	新しい評価法の名前を入力
txtBiko	テキストボックス	新しい評価法に対するコメントを入力
txtID	テキストボックス	新しい評価法の番号を入力
txtTemplate	テキストボックス	フィールド名が書かれた Excel ファイル名を入力
txtRange	テキストボックス	フィールド名の領域に付けられた名前を入力
cmdBrowse	コマンドボタン	テンプレートファイル名をコンダイアログを使って参照するボタン
cmdOK	コマンドボタン	新しい評価法を[評価法テーブル]に追加テンプレートファイルを参照して、その評価法専用の損傷値テーブルを STAR データベースの中に作成
cmdCancel	コマンドボタン	何もしないで画面を閉ず

◆ U_HYOKA.FRM

損傷評価法の登録・削除を行う画面である。[STAR 制御]メニューの[損傷評価法の登録]コマンドを実行した際に表示される。

Table 3.26 Summary of Source Program File “U_HYOKA.FRM”

コントロール名	種類	概要
lstHyoka	リストボックス	登録済みの評価法を一覧表示
cmdAdd	コマンドボタン	U_HYK_AD.FRM に移行
cmdDelete	コマンドボタン	指定された評価法を[評価法テーブル]から削除また、[管理データテーブル]から関連するレコードを削除さらにその評価法のテーブルを STAR データベースから削除
cmdExit	コマンドボタン	画面を閉ず

◆ U_KANRI2.FRM

管理データの登録・削除を行う画面である。[STAR 制御]メニューの[管理データの登録]コマンドを実行した際に表示される。

Table 3.27 Summary of Source Program File “U_KANRI2.FRM”

コントロール名	種類	概要
outBubun	アウトライン	試験装置・試験体・試験部分をツリー表示
cmdAdd	コマンドボタン	U_KNR_A2.FRM に移行
cmdDelete	コマンドボタン	指定されたデータのを[管理データテーブル]およびすべてのテーブルから削除上位の部位が指定された場合は、下位のデータも削除される
cmdExit	コマンドボタン	画面を閉ず

◆ U_KNR_A2.FRM

管理データを新規に登録する画面である。[管理データの登録]ダイアログボックスで[追加]を選択した場合に表示される。

Table 3.28 Summary of Source Program File "U_KNR_A2.FRM"

コントロール名	種類	概要
txtSochi	テキストボックス	試験装置名を入力
txtTai	テキストボックス	試験体名を入力
txtBubun	テキストボックス	試験部分名を入力
txtLine	テキストボックス	試験部分の評価ライン数を入力
txtID	テキストボックス	他の試験部分と区別するための番号が表示される
cmdReg	コマンドボタン	[管理データテーブル]にデータを追加
cmdCancel	コマンドボタン	何もしないで画面を閉じる

◆ U_MAT.FRM

材料名の登録・削除を行う画面である。[STAR 制御]メニューの[材料の登録]コマンドを実行した際に表示される。

Table 3.29 Summary of Source Program File "U_MAT.FRM"

コントロール名	種類	概要
lstMat	テキストボックス	試験装置名を入力
cmdAdd	コマンドボタン	新規に材料を追加する txtNewMat, txtNewBiko, txtNewID, cmdReg, cmdCancel を有効にする
cmdDelete	コマンドボタン	指定されたデータを[材料テーブル]から削除
cmdExit	コマンドボタン	画面を閉じる
txtNewMat	テキストボックス	新しい材料名を入力
txtNewBiko	テキストボックス	その材料についてのコメントを入力
txtNewID	テキストボックス	その材料の番号を入力
cmdReg	コマンドボタン	[材料テーブル]にデータを追加
cmdCancel	コマンドボタン	新規登録を取り消す

◆ U_SETUP.FRM

STAR の環境を設定する画面である。[STAR 制御]メニューの[環境設定]コマンドを実行した際に表示される。

Table 3.30 Summary of Source Program File "U_SETUP.FRM"

コントロール名	種類	概要
TxtDSN	テキストボックス	STAR データベースの窓口となるデータベース名を入力
TxtDBN	テキストボックス	STAR データベースの名前を入力
TxtQDB	テキストボックス	分布比較・統計の検索用のMDB ファイル名を入力
CmbBTView	コンボボックス	検索結果を表示方法を選択
TxtWorkDir	テキストボックス	一時作業ファイルを作成するディレクトリ名を入力
TxtWSketchDir	テキストボックス	AutoSketch があるディレクトリ名を入力
CmdOK	コマンドボタン	リストリを更新し、STAR を強制終了
CmdCancel	コマンドボタン	何もしないで画面を閉じる

◆ U_SQL.FRM

SQL 文を直接実行する画面である。[STAR 制御]メニューの[SQL]コマンドを実行した際に表示される。

Table 3.31 Summary of Source Program File "U_SQL.FRM"

コントロール名	種類	概要
txtSQL	テキストボックス	SQL 文を入力

chkSelect	チェックボックス	SELECT 文の場合はこのボックスをチェック。
cmdOK	コマンドボタン	SQL 文を実行。
cmdCancel	コマンドボタン	何もしないで画面を閉ず。
grdSelect	テキストボックス	SELECT 文の場合は、結果をここに表示。

◆ U_VIEW.FRM

(現在未使用)

登録されている管理データの一覧を表示する画面である。[STAR 制御]メニューの[管理データの一覧]コマンドを実行した際に表示される。

Table 3.32 Summary of Source Program File “U_VIEW.FRM”

コントロール名	種類	概要
dbgKanri	DBグリッド	dtaKanri を画面に表示
dtaKanri	データ	[管理データテーブル]と連結されたデータコントロール
cmdClose	コマンドボタン	画面を閉ず

◆ X_NORMAL.FRM

現在選択されている試験部分や解析ケースを表示する子ウィンドウである。ただし、実質的なメインモジュールである。

Table 3.33 Summary of Source Program File “X_NORMAL.FRM”

コントロール名	種類	概要
lblSochi	ラベル	現在選択されている試験装置名を表示
lblTai	ラベル	現在選択されている試験体名を表示
lblBubun	ラベル	現在選択されている試験部分名を表示
lblCase	ラベル	現在選択されている解析ケース名を表示

◆ X_RESGRD.FRM

分布比較および統計処理において検索・抽出されたデータをグリッドコントロールを使って表示する子ウィンドウである。

Table 3.34 Summary of Source Program File “X_RESGRD.FRM”

コントロール名	種類	概要
grdRes	グリッド	データの内容を表示
mnuClose	メニュー	フォームを閉ず
mnuAll	メニュー	すべてのセルを選択(反転表示)
mnuCopy	メニュー	選択されているセルの内容をクリップボードへコピー

◆ X_RESOLE.FRM

分布比較および統計処理において検索・抽出されたデータを OLE コントロールを使って表示する子ウィンドウである。このウィンドウは EXCEL とリンクしているので、ダブルクリックすると EXCEL が起動する。

Table 3.35 Summary of Source Program File “X_RESOLE.FRM”

コントロール名	種類	概要
oleCtrl	OLE	検索された結果を\$BUN.XLS または\$TOK.XLS に作成し、このファイルと OLE コントロールをリンク

◆ X_SELDB1.FRM

実験データ(き裂、温度、形状)のファイル名を入力する際に表示される画面である。[実験データ]メニューの[き裂データ入力]、[温度データ入力]および[形状データ入力]コマンドを実行した際に表示される。

Table 3.36 Summary of Source Program File “X_SELDB1.FRM”

コントロール名	種類	概要
lblCurFile	ラベル	ファイルボックスで選択されたファイル名を表示
filFileList	File ボックス	現在選択されているディレクトリ内のファイル名を一覧表示
lblCurDir	ラベル	現在選択されているディレクトリ名を表示
dirDirList	Dir ボックス	現在選択されているドライブ内のディレクトリ名を一覧表示
lstDataRange	リストボックス	現在選択されている EXCEL ファイル内の名前を一覧表示
cmbFileType	コンボボックス	ファイルの種類を選択
drvDrive	ドライブボックス	ドライブを選択
cmdOK	コマンドボタン	選択されたファイルを読み込み、データを一時的に画面に表示 (I_VW_DBG.FRM)
cmdCancel	コマンドボタン	何もしないで画面を閉じる

◆ X_SELDB3.FRM

解析結果のファイル名を入力する際に表示される画面である。[解析データ]メニューの[解析結果入力]コマンドを実行した際に表示される。

Table 3.37 Summary of Source Program File “X_SELDB3.FRM”

コントロール名	種類	概要
lblCurFile1	ラベル	選択された評価ライン1のファイル名を表示
filFile1	File ボックス	評価ライン1のファイル名を一覧表示
cmbRange1	コンボボックス	評価ライン1の EXCEL ファイル内の名前一覧を表示
lblCurFile2	ラベル	選択された評価ライン2のファイル名を表示
filFile2	File ボックス	評価ライン2のファイル名を一覧表示
cmbRange2	コンボボックス	評価ライン2の EXCEL ファイル内の名前一覧を表示
lblCurFile3	ラベル	選択された評価ライン3のファイル名を表示
filFile3	File ボックス	評価ライン3のファイル名を一覧表示
cmbRange3	コンボボックス	評価ライン3の EXCEL ファイル内の名前一覧を表示
dirDirectory	Dir ボックス	現在選択されているドライブ内のディレクトリ名を一覧表示
cmbFileType	コンボボックス	ファイルの種類を選択
drvDrive	ドライブボックス	ドライブを選択
cmdOK	コマンドボタン	選択されたファイルを読み込み、データを一時的に画面に表示 (N_ANL_VW.FRM)
cmdCancel	コマンドボタン	何もしないで画面を閉じる

◆ X_USER2.FRM

STAR 起動時に、SQL サーバーの STAR データベースに接続するためのログイン名とパスワードを入力する画面である。

Table 3.38 Summary of Source Program File “X_USER2.FRM”

コントロール名	種類	概要
txtUID	テキストボックス	ログイン名を入力
txtPWD	テキストボックス	パスワードを入力
cmdOK	コマンドボタン	STAR データベースに接続

cmdQuit	コマンドボタン	STARを終了
cmdSetup	コマンドボタン	セットアップ画面を表示(U_SETUP.FRM)

◆ STAR1.BAS

各種のサブルーチンが収められた BASIC モジュールである。

◆ STARC.BAS

各種の定数が収められた BASIC モジュールである。

◆ DXFREAD.BAS

[形状データ入力]コマンドで DXF ファイルを読み込むための BASIC モジュールである。

◆ STAR.RTF

オンラインヘルプのソースファイル。MS-WORD のリッチテキスト形式。

◆ STAR.HPJ

オンラインヘルプのソースファイルをコンパイルする際のプロジェクトファイル。

3.2 STAR 7.1 の変更点

STAR 7.1 では以下に示す機能の強化を行っている。

(1) AutoSketch 用 DXF ファイル(SHAPE_1.DXF)の読み込みルーチンの作成

[形状データ入力]コマンドで DXF ファイルを読み込む場合、読み込に可能な DXF ファイルはデジタルタイザ入力システムで出力した DXF ファイルに限られている。今回、以前のバージョンにおいて、[試験部分形状図]コマンドで出力した DXF 形式の形状データも、[形状データ入力]コマンドで読み込めるように改良した。[形状データ入力]コマンドの操作において、特に変更になることはなく、データのデジタルタイザ入力システムで出力した DXF ファイルも、以前のバージョンにおいて[試験部分形状図]コマンドで出力した DXF ファイルも、同じように扱える。DXF ファイルを読み込むルーチンは DXFREAD2.BAS に収められている。

DXF ファイルは、HEADER・TABLES・BLOCKS・ENTITIES の 4 つのセクションから構成されるが、DXFREAD2 では HEADER と TABLES の情報は無視し、形状データ作成に必要な BLOCKS と ENTITIES のみの情報を読み込む。また、DXF で規定されているすべてのエンティティタイプに対応はしていない。DXFREAD2 で対応しているエンティティタイプを以下に示す。

3.39 Entity Type Table

エンティティタイプ	内容
BLOCK	複合図形の定義
INSERT	複合図形の挿入
LINE	線分データ
POLYLINE	ポリラインデータ

VERTEX	ポリラインの各頂点のデータ
SEQEND	ポリラインのシーケンスの終り

(2)レイヤ機能を利用した各要素の個別表示機能

現在、複数に別れている DXF ファイル(SHAPE、LINE、SECT、CRACK 等)を 1 つにまとめ、各要素を識別させるために、その中にレイヤ情報を付加した。これにより AutoSketch において、レイヤ指定による表示が可能となる。詳細は 4 章を参照のこと。

(3)エラートラップルーチンの付加

各コマンドにエラートラップルーチンを設定し、エラーが発生した場合はそのコマンドを強制終了するように改良した。Visual Basic の On Error Goto ステートメントを用いて、各コマンドにエラー処理ルーチンを用意した。あるモジュール内でエラーが発生した場合、処理はエラー処理ルーチンに移行する。エラー処理ルーチンでは、最初にどのモジュールでエラーが発生したかを示すダイアログを表示し、次にエラーの詳細情報を表示する。最後に、閉じられていないファイルとテーブルをクローズして、コマンドを強制終了させる。

[管理データの登録]、[解析ケースの登録]、[損傷評価法の登録]、および[材料の登録]コマンドを除いて、エラー処理ルーチンは、単にコマンドを強制終了させるだけの機能しか持たず、エラーの内容によってはデータベースの整合性がとれなくなる場合もありえる。

(4)データ登録時の入力ファイルのフォーマットチェック

き裂データ・温度データ・形状データ・解析結果データ(Excel、CSV 形式等)を読み込む際に、それらの入力データに所定のフィールドが存在するかのチェック機能を付加した。各入力ファイルで必要とされるフィールド名を以下に示す。入力ファイルに以下に示すフィールド名が存在しない場合は、エラーメッセージを表示し、処理を中止する。

3.40 Table of Field Item

番号	き裂データ XLS, CSV	温度データ XLS, CSV, STF	形状データ XLS, CSV, STF	解析結果	
				XLS, CSV	STF
0	日付	TIME1	MODEL-R	起点 r1	R1
1	ライン番号	TEMP1	MODEL-Z	起点 z1	Z1
2	起点 r1	CONV1		起点 θ 1	H1
3	起点 z1			終点 r2	R2
4	起点 θ 1			終点 z2	Z2
5	終点 r2			終点 θ 2	H2
6	終点 z2			TEMPMAX	TEMPMAX
7	き裂長さ			TIME1	TIME1
8	母・溶・HAZ			TEMP1	TEMP1
9	溶接法			LTEMP1	LTEMP1
10	余盛有無			MTEMP1	MTEMP1
11	余盛形状			σ r1	STRS11
12	溶接備考			σ z1	STRS21
13	粒内・粒界			σ θ 1	STRS31
14	き裂形状			τ rz1	STRS41
15	き裂発生サイクル			L σ r1	LSTRS11
16	スライエーション			L σ z1	LSTRS21
17	ピークマーク			L σ θ 1	LSTRS31
18	統計処理アケイブ			L τ rz1	LSTRS41
19	結晶粒度			M σ r1	MSTRS11

20	備考			$M\sigma z1$	MSTRS21
21				$M\sigma\theta 1$	MSTRS31
22				$M\tau rz1$	MSTRS41
23				$\varepsilon r1$	EE11
24				$\varepsilon z1$	EE21
25				$\varepsilon\theta 1$	EE31
26				$\gamma rz1$	EE41
27				$\varepsilon pr1$	EP11
28				$\varepsilon pz1$	EP21
29				$\varepsilon p\theta 1$	EP31
30				$\gamma prz1$	EP41
31				$\varepsilon cr1$	EC11
32				$\varepsilon cz1$	EC21
33				$\varepsilon c\theta 1$	EC31
34				$\gamma crz1$	EC41

(5)計算処理(表計算・損傷値・補間処理)に入力側データの有無の確認

表計算・損傷値・補間処理の計算処理の際、元になるデータテーブルが既に登録されているかのチェック機能を見直した。以下にそれぞれの計算処理において必要とされるデータを示す。表計算・損傷値・補間処理において、現解析ケースの計算処理では、対応するデータが存在しない場合、そのメッセージを画面に表示する。また、現試験体および全テーブルに関する処理では、その処理をスキップし、次の部位の処理に移行する。

Table 3.41 Table of Calculation Item

表計算	損傷計算	Qの計算	補間処理
解析結果	解析結果	解析結果(格納先)	き裂_S
	試験条件	表計算	解析、表計算、損傷値

(6)管理データ登録(管理データ・解析ケース・評価法等)時のバックアップテーブルの作成

管理データ・解析ケース・損傷評価法・材料の登録削除処理の際、処理実施前に元のテーブル(管理データテーブル、解析ケーステーブル、評価法テーブル、材料テーブル)のバックアップを作成する。処理中に何らかのエラーが発生した場合、処理を中止してバックアップしておいたテーブルを復元する。エラーの有無に係わらず、全処理が終了したのちバックアップテーブルは削除される。評価法を削除する場合は、評価法テーブルの他に削除対象の損傷値テーブルもバックアップする。

試験部位(試験装置、試験体、試験部分)および解析ケースを削除する場合、管理データテーブルや解析ケーステーブルから削除対象となる行を削除するが、解析・表計算・各損傷値等のデータテーブルからも対象となるデータが削除される。その削除処理の際、管理データテーブルや解析ケーステーブルのバックアップは作成するが、データテーブルのバックアップは行えない。

Table 3.42 Correspond Table of Management Data – Backup Data

管理テーブル名	バックアップテーブル名
管理データテーブル	管理データテーブル BACKUP
解析ケーステーブル	解析ケーステーブル BACKUP
評価法テーブル	評価法テーブル BACKUP
材料テーブル	材料テーブル BACKUP

(7)起動時の環境(作業・AutoSketch ディレクトリ、検索用 MDB 等の有無)の確認

[環境設定]コマンドにより、入力された環境データは、レジストリの HKEY_CURRENT_USER / Software / VB and VBA Program Settings / STAR7 / STAR に登録される。以下に STAR で使用しているエントリ名を示す。

Table 3.43 Entry Item Table

エントリ名	内容
BTView	検索結果の表示方法(0:グリッドコントロール、1:Excel)
DatabaseName	SQL サーバに登録されている STAR のデータベース名
DataSourceName	データベースにアクセスするための ODBC のデータソース名
MaxRetrieve	検索結果を表示するさいの最大行数
QueryDB	データベースを検索するためのクエリーが定義された MDB ファイル名
WorkDir	作業ファイルを書込むためのワークディレクトリ名
WSketchDir	AutoSketch の実行ファイルが存在するディレクトリ名

STAR 起動時に、これらの情報を読み取りチェックする。エントリデータに不備があった場合の処理方法の一覧を以下に示す。

Table 3.44 List of Entry Process

エントリ名	未設定の場合	間違ったデータの場合
BTView	0	0
DatabaseName	設定画面の表示	(ログイン時にエラーとなる)
DataSourceName	設定画面の表示	(ログイン時にエラーとなる)
MaxRetrieve	300	
QueryDB	設定画面の表示	設定画面の表示
WorkDir	設定画面の表示	設定画面の表示
WSketchDir		設定画面の表示

3.3 STAR ソースプログラムのインストール

STAR 7.1 は Visual Basic 5.0(SP3)で作成されている。以下に STAR 7.1 のソースファイルをフロッピーからハードディスクにインストール際の手順を示す。

(1) Damage96.OCX 内部で使用している以下の DLL ファイルを Windows の SYSTEM フォルダにコピーする。

・ MATLIB32.DLL	・ MSFRT10.DLL	・ MSVCRT10.DLL
・ MFC40D.DLL	・ MFCO40D.DLL	・ MSVC40D.DL
・ MFC42D.DLL	・ MFCO42D.DLL	・ MSVCRTD.DLL

(2) Damage96.OCX を Windows の SYSTEM フォルダにコピーする。

(3) Visual Basic 5.0 をインストールする。

(4) Visual Basic 4.0 のコンポーネントをインストール

- ・ VB5.0 の CD の¥VB5.0¥TOOLS.JPN¥CONTROLS の全ファイルを Windows の SYSTEM フォルダにコピーする。
- ・ REGSVR32.EXE を使って、各コントロールをレジストリに登録する
 - REGSVR32 ANIBTN32.OCX
 - REGSVR32 GAUGE32.OCX
 - REGSVR32 GRID32.OCX
 - REGSVR32 KEYSTA32.OCX
 - REGSVR32 MSOUTL32.OCX
 - REGSVR32 SPIN32.OCX
 - REGSVR32 THREED32.OCX
- ・ コントロールのデザイン時のライセンスに登録する
 - REGEDIT VBCTRLS.REG

(5) VB5.0 の Service Pack 3 をインストール。SP3 はマイクロソフトのホームページよりダウンロード可能である。

(6) STAR 7.1 用のフォルダを作成し、STAR のソースプログラムのフロッピーの SOURCE フォルダにあるすべてのファイル(サブフォルダも含めて)をコピーする。

(7) TCALCSV.EXE と INTPSV.EXE プログラムの登録する。

- ・ フロッピーからコピーした表計算および補間処理用の OLE サーバプログラム(TCALCSV.EXE と INTPSV.EXE)を¥Program Files¥Common Files¥OleSvr フォルダにコピーする。
- ・ VB5.0 を起動し、STAR を読み込み、[プロジェクト]/[参照設定]コマンドを使って、これらのサーバプログラムを登録する。

4 AutoSketch による形状図表示

4.1 概要

[データ選択]メニューの[試験部分形状図]コマンドでは試験部分の輪郭・き裂・評価ライン・評価断面の表示を行える。STAR 4.0 までは形状図の表示は STAR 独自の WINDOWS アプリケーション(SHAPE.EXE)を使用していたが、5.0 以降では市販アプリケーションである AutoSketch 2.2 を使用して形状図表示を行っている。形状図プログラム(SHAPE.EXE)は形状図表示の専用プログラムであり、その他の特殊な機能はないもの、ソースプログラムの行数から見れば 3000 行以上にもおよぶ C 言語で書かれた大きなプログラムであった。このプログラムを修正したり、新規の機能の追加を行う場合にかなりの労力がかかっていた。

そこで、形状図プログラム(SHAPE.EXE)の代替としての市販アプリケーションである AutoSketch を採用することにした。AutoSketch はディジタイザ入力システムで使用している AutoCAD と同じオートデスク社が販売しているアプリケーションであり、両アプリケーション間でデータの互換性がある。AutoCAD で入力したデータが DXF ファイルを介して AutoSketch でも表示することができる。

AutoSketch のデフォルトの図形データのファイル形式は SKD 形式だが、これはバイナリ形式で AutoSketch 専用のもであり、AutoCAD との互換性はない。両アプリケーションで扱えるファイル形式として DXF 形式というフォーマットがある。この書式はテキスト形式なので、ユーザが修正したり、作成することもできる。もちろん、EXCEL や VB アプリケーションから図形データを DXF 形式のファイルに出力することもできる。STAR の[試験部分形状図]コマンドでは以下に示すデータを作業ディレクトリの \$\$SHAPE.DXF ファイルに出力する。

Table 4.1 Correspond Table of STAR Data

データ種別	対応テーブル
形状データ	[形状]テーブル
評価ラインデータ	[解析]テーブル
評価断面データ	[解析]テーブル
き裂データ	[き裂]テーブル

AutoSketch ではこの DXF ファイルを[ファイル]メニューの[読み込み]コマンドで読み込み、表示を行う。ズームやプリンタ出力といった処理は AutoSketch の機能をそのまま使用できる。

STAR 7.0 では、上記のデータを個々の DXF ファイル(\$SHAPE.DXF, \$CRACK.DXF, \$LINE_1.DXF 等)に分けて出力し、AutoSketch 側で必要な情報を選択して読み込むようになっていた。STAR 7.1 では、これらの複数に別れている DXF ファイルを 1 つにまとめて出力するように変更している。その際、各要素が識別できるようにそれぞれ個別のレイヤ情報を付加させている。これにより、AutoSketch の機能により、レイヤ指定による表示が可能になった。

4.2 AutoSketch の起動

STAR の[試験部分形状図]コマンドが実行されると、STAR システムはデータベースから形状図表示に必要なデータの抽出を行い、抽出されたデータを STAR のワークディレクトリに DXF 形式のファイルとして掃き出し、その後 AutoSketch を自動的に起動し、これらのファイルを読み込む。AutoSketch を起動する前に、STAR は以下に示す AutoSketch 制御用のマクロファイル (\$SHAPE.MCR)を作業ディレクトリに作成する。これは 3 行目に示された形状データ(フルパス名)をオープンし、ロードした図が表示されるように全体表示を行うマクロファイルである。

\$SHAPE.MCRファイルの例

```
REM r2.1 - AutoSketch for Windows Release 2.1 Macro File.
IMPORT
SET DXFFILENAME $SHAPE
SET DXFDIR C:\STARWORK\
DialogBoxReturn 1
DialogBoxReturn 1
ZOOMFULL
```

さらに、STAR は AutoSketch を起動する際に、マクロの自動実行機能によって上記のマクロファイルを、AutoSketch を実行させることにより画面上に形状図を表示する。

4.3 画層

AutoSketch では画層を扱うことができる。画層とはトレーシングペーパーのようなもので、それぞれの絵や部品を別々のトレーシングペーパーに描き、後でそれらを重ねて表示する機能のことである。[試験部分形状図]コマンドで DXF ファイルを作成する際、個々の情報は画層に分けて出力している。STAR では画層を次のように使用している。

Table 4.2 Correspond Table of Shape Data

画層名	表示内容	表示色
CrackXXX	き裂(XXXは角度。角度の数だけ画層が存在する)	赤
Line1	評価ライン1	黒
Line2	評価ライン2	黒
Line3	評価ライン3	黒
Sect1	評価断面1	黒
Sect2	評価断面2	黒
Sect3	評価断面3	黒
Shape	形状データ	黒

4.4 入出力仕様

DXF ファイルの一般的なファイルは以下のセクションから構成される。通常、図形データは BLOCKS セクションと ENTITIES セクションに納められる。しかしながら、STAR の形状データを描画するためには、HEADER セクションと TABLES セクションのデータは必須ではないので、STAR から掃き出される DXF ファイルには、これらのセクションを付加していない。

- ・ HEADER セクション
- ・ TABLES セクション
- ・ BLOCKS セクション
- ・ ENTITIES セクション

◆形状データ

形状データは Shape 画層に格納されている。形状データは線分の形でブロックセクションに定義されている。ブロック名は SHAPE になる。

◆き裂データ

き裂データはその角度ごとに個々の画層に格納され、画層名は CrackXXX になる。XXX は断面の角度を示す。き裂データは線分の形でブロックセクションに定義されている。ブロック名は CRACK になる。

◆評価ライン

評価ラインデータは評価ラインごとに、Line1・Line2・Line3 画層に格納される。評価ラインは何百もの線分で構成されているので、一つ一つ移動するのは現実的ではない。AutoCAD や AutoSketch にはいくつかの図形データを一つにまとめて扱う複合図形という機能がある。複数のデータを複合図形として定義することによって、その複合図形の移動が線分一つを操作するのと同じ方法で行うことができる。複合図形はブロックセクションに名前を付けて定義(作成)し、エンティティセクションにはその参照する複合図形の名前だけを挿入する。

評価ラインデータはポリラインエンティティを使い線分を定義している。線分の太さを変えられるエンティティはポリラインだけである。ブロック名は HLINE1、HLINE2、HLINE3 になる。

◆評価断面データ

評価断面データはラインごとに、Sect1・Sect2・Sect3 画層に格納される。評価断面も評価ラインと同様に複合図形の形で格納されている。ブロック名は HSECT1、HSECT2、HSECT3 になる。

以下に、ブロック名と画層名の表を示す。

Table 4.3 Correspond Table of Block Data

データ内容	ブロック名	画層名	表示色
き裂	CRACK	CrackXXX	赤
評価ライン1	HLINE1	Line1	黒
評価ライン2	HLINE2	Line2	黒
評価ライン3	HLINE3	Line3	黒
評価断面1	HSECT1	Sect1	黒
評価断面2	HSECT2	Sect2	黒
評価断面3	HSECT3	Sect3	黒
形状データ	SHAPE	Shape	黒

5. 外部プログラム

5.1 表計算プログラム

表計算処理は、STAR 本体とは独立した TCALCSV.EXE が行う。この TCALCSV.EXE は STAR 本体にあった表計算機能を OLE サーバーとして、部品化したものである。処理の手順としては、STAR から試験部分の解析・時点データを TCALCSV.EXE に転送し、TCALCSV では転送されたデータに対して表計算処理を行う。STAR は TCALCSV から表計算結果を読み込み、STAR データベースの表計算テーブルに格納する。データの転送および計算は行単位ではなく試験部分単位でまとめて行われる。

5.1.1 モジュール構成

TCALCSV.EXE は以下のモジュールで構成されている。

Table 5.1 Summary of Program Module “TCALCV.EXE”

モジュール名	概要
TCALCSV.FRM	ダミーのフォームモジュール
TCALCSV.CLS	プロパティ・メソッドが定義されたモジュール
TCALCSV.BAS	Mainだけが定義された標準モジュール
STAR.C.BAS	各種定数を定義した標準モジュール。STAR.EXEと共用

5.1.2 プロシージャ

プロシージャはすべて TCALCSV.CLS に定義されている。TCALCSV で使用しているプロシージャの概要を以下に示す。

Table 5.2 Summary of Procedure Module “TCALCSV.CLS”

プロシージャ名	タイプ	入出力	概要
AnalCols	プロパティ	Get/Set	解析テーブルのフィールド数の取得/設定(未使用)
AnalRows	プロパティ	Get/Set	解析テーブルの行数の取得/設定この値を設定すると自動的に時点および表計算の行数も同じ値に設定される
JitenCols	プロパティ	Get/Set	時点テーブルのフィールド数の取得/設定(未使用)
JitenRows	プロパティ	Get/Set	時点テーブルの行数の取得/設定
Max3	関数		3つの引数から最小のものを選択
Min3	関数		3つの引数から最大のものを選択
TableCols	プロパティ	Get/Set	表計算テーブルのフィールド数の取得/設定(未使用)
TableRows	プロパティ	Get/Set	表計算テーブルの行数の取得/設定
Exec	メソッド		表計算の実行

5.1.3 表計算式

Exec メソッドには以下に示す 4 つの引数がある。

Table 5.3 Summary of Array Table

番号	引き数名	概要
1	AD (300,100)	解析データが入った2次元配列 第1引数: レコド、第2引数: フィールド
2	JD1 (300,100)	評価時点1の各データが入った2次元配列 第1引数: レコド、第2引数: フィールド
3	JD2 (300,100)	評価時点2の各データが入った2次元配列 第1引数: レコド、第2引数: フィールド
4	TD (300,100)	表計算結果を格納する2次元配列 第1引数: レコド、第2引数: フィールド

Exec メソッドが実行されると、AD,JD1,JD2 の配列データから表計算を行い、結果を TD 配列に格納する。以下に、表計算で使用している計算式を示す。

Table 5.4 Explanation of Excel Method

番号	フィールド名	計算式
0	id	=解析!id
1	参照id	=解析!参照id
2	日付	=解析!日付
3	解析ケース	=解析!解析ケース
4	ライン番号	=解析!ライン番号
5	L	=解析!L
6	起点r1	=解析!起点r1
7	起点z1	=解析!起点z1
8	起点θ1	=解析!起点θ1
9	dσr	=時点1!σr-時点2!σr
10	dσz	=時点1!σz-時点2!σz
11	dσθ	=時点1!σθ-時点2!σθ
12	dτrz	=時点1!τrz-時点2!τrz
13	dLσr	=時点1!Lσr-時点2!Lσr
14	dLσz	=時点1!Lσz-時点2!Lσz
15	dLσθ	=時点1!Lσθ-時点2!Lσθ
16	dLτrz	=時点1!Lτrz-時点2!Lτrz
17	dMσr	=時点1!Mσr-時点2!Mσr
18	dMσz	=時点1!Mσz-時点2!Mσz
19	dMσθ	=時点1!Mσθ-時点2!Mσθ
20	dMτrz	=時点1!Mτrz-時点2!Mτrz
21	dσ1	=(dσr+dσz)/2 + SQRT(((dσr-dσz)/2)^2 + dτrz^2)
22	dσ2	=(dσr+dσz)/2 - SQRT(((dσr-dσz)/2)^2 + dτrz^2)
23	dσ3	=dσθ
24	dLσ1	=(dLσr+dLσz)/2 + SQRT(((dLσr-dLσz)/2)^2 + dLτrz^2)
25	dLσ2	=(dLσr+dLσz)/2 - SQRT(((dLσr-dLσz)/2)^2 + dLτrz^2)
26	dLσ3	=dLσθ
27	dMσ1	=(dMσr+dMσz)/2 + SQRT(((dMσr-dMσz)/2)^2 + dMτrz^2)
28	dMσ2	=(dMσr+dMσz)/2 - SQRT(((dMσr-dMσz)/2)^2 + dMτrz^2)
29	dMσ3	=dMσθ
30	Δσr	=ABS(dσr)
31	Δσz	=ABS(dσz)
32	Δσθ	=ABS(dσθ)
33	Δτrz	=ABS(dτrz)
34	ΔLσr	=ABS(dLσr)
35	ΔLσz	=ABS(dLσz)
36	ΔLσθ	=ABS(dLσθ)
37	ΔLτrz	=ABS(dLτrz)
38	ΔMσr	=ABS(dMσr)
39	ΔMσz	=ABS(dMσz)
40	ΔMσθ	=ABS(dMσθ)
41	ΔMτrz	=ABS(dMτrz)

42	$\Delta \sigma 1$	=ABS($\Delta \sigma 1$)
43	$\Delta \sigma 2$	=ABS($\Delta \sigma 2$)
44	$\Delta \sigma 3$	=ABS($\Delta \sigma 3$)
45	$\Delta L \sigma 1$	=ABS($\Delta L \sigma 1$)
46	$\Delta L \sigma 2$	=ABS($\Delta L \sigma 2$)
47	$\Delta L \sigma 3$	=ABS($\Delta L \sigma 3$)
48	$\Delta M \sigma 1$	=ABS($\Delta M \sigma 1$)
49	$\Delta M \sigma 2$	=ABS($\Delta M \sigma 2$)
50	$\Delta M \sigma 3$	=ABS($\Delta M \sigma 3$)
51	ΔVMS	=SQRT(((1/2)*((d σ r-d σ z)^2+(d σ z-d σ θ)^2+(d σ θ -d σ r)^2+6*d τ rz^2))
52	$\Delta TRES$	=MAX(ABS(d σ 1-d σ 2),ABS(d σ 2-d σ 3),ABS(d σ 3-d σ 1))
53	$\Delta RANK$	=MAX($\Delta \sigma 1$, $\Delta \sigma 2$, $\Delta \sigma 3$)
54	$\Delta LVMS$	=SQRT(((1/2)*((dL σ r-dL σ z)^2+(dL σ z-dL σ θ)^2+(dL σ θ -dL σ r)^2+6*dL τ rz^2))
55	$\Delta LTRES$	=MAX(ABS(dL σ 1-dL σ 2),ABS(dL σ 2-dL σ 3),ABS(dL σ 3-dL σ 1))
56	$\Delta LRANK$	=MAX($\Delta L \sigma 1$, $\Delta L \sigma 2$, $\Delta L \sigma 3$)
57	$\Delta MVMS$	=SQRT(((1/2)*((dM σ r-dM σ z)^2+(dM σ z-dM σ θ)^2+(dM σ θ -dM σ r)^2+6*dM τ rz^2))
58	$\Delta MTRES$	=MAX(ABS(dM σ 1-dM σ 2),ABS(dM σ 2-dM σ 3),ABS(dM σ 3-dM σ 1))
59	$\Delta MRANK$	=MAX($\Delta M \sigma 1$, $\Delta M \sigma 2$, $\Delta M \sigma 3$)
60	d ϵ r	=時点1! ϵ r-時点2! ϵ r
61	d ϵ z	=時点1! ϵ z-時点2! ϵ z
62	d ϵ θ	=時点1! ϵ θ -時点2! ϵ θ
63	d γ rz	=時点1! γ rz-時点2! γ rz
64	d ϵ pr	=時点1! ϵ pr-時点2! ϵ pr
65	d ϵ pz	=時点1! ϵ pz-時点2! ϵ pz
66	d ϵ p θ	=時点1! ϵ p θ -時点2! ϵ p θ
67	d γ prz	=時点1! γ prz-時点2! γ prz
68	d ϵ cr	=時点1! ϵ cr-時点2! ϵ cr
69	d ϵ cz	=時点1! ϵ cz-時点2! ϵ cz
70	d ϵ c θ	=時点1! ϵ c θ -時点2! ϵ c θ
71	d γ crz	=時点1! γ crz-時点2! γ crz
72	$\Delta EEPCEQ$	=((1/3)*SQRT(2*((d ϵ r+d ϵ pr+d ϵ cr)-(d ϵ z+d ϵ pz+d ϵ cz))^2 + ((d ϵ z+d ϵ pz+d ϵ cz)-(d ϵ θ +d ϵ p θ +d ϵ c θ))^2 + ((d ϵ θ +d ϵ p θ +d ϵ c θ)-(d ϵ r+d ϵ pr+d ϵ cr))^2 + (3/2)*(d γ rz+d γ prz+d γ crz)^2))
73	$\Delta EPEQ$	=((1/3)*SQRT(2*((d ϵ pr-d ϵ pz)^2 + (d ϵ pz-d ϵ p θ)^2 + (d ϵ p θ -d ϵ pr)^2 + (3/2)*d γ prz^2))
74	$\Delta ECEQ$	=((1/3)*SQRT(2*((d ϵ cr-d ϵ cz)^2 + (d ϵ cz-d ϵ c θ)^2 + (d ϵ c θ -d ϵ cr)^2 + (3/2)*d γ crz^2))

5.1.4 TCALCSV の実行例

以下に、Visual Basic からの TCALCSV.EXE の実行例を示す。TCALCSV.EXE のクラス名は"TCALCSV.TCALC"になる。

```

Dim Anal(300, 100) As Variant
Dim Jiten1(300, 100) As Variant
Dim Jiten2(300, 100) As Variant
Dim Table(300, 100) As Variant
Dim obj As Object

Set obj = CreateObject("TCALCSV.TCALC")
obj.AnalRows = 10          '解析結果が10レコードと仮定

'解析データをAnal配列へ格納

'時点データをJiten1,Jiten2の各配列へ格納

obj.Exec Anal, Jiten1, Jiten2, Table

'Table配列(計算結果)のデータを表計算テーブルに格納

Set obj = Nothing
    
```

5.2 補間処理プログラム

補間処理は、STAR 本体とは独立した INTPSV.EXE が行う。この INTPSV.EXE は STAR 本体にあった補間処理機能を OLE サーバーとして、部品化したものである。処理の手順としては、STAR から試験部分単位で解析、表計算あるいは損傷値データを INTPSV.EXE に転送し、INTPSV では転送されたデータに対して補間処理を行う。STAR は INTPSV から補間処理結果を読み込み、STAR データベースの対応した各統計用のテーブルに格納する。データの転送および計算は行単位ではなく試験部分単位でまとめて行われる。

5.2.1 モジュール構成

INTPSV.EXE は以下のモジュールで構成されている。

Table 5.5 Explanation of Module Program "INTPSV.EXE"

モジュール名	概要
INTPSV.FRM	ダミーのフォームモジュール
INTPSV.CLS	プロパティ・メソッドが定義されたモジュール
INTPSV.BAS	Mainだけが定義された標準モジュール

5.2.2 プロシージャ

プロシージャはすべて INTPSV.CLS に定義されている。INTPSV で使用しているプロシージャの概要を以下に示す。

Table 5.6 Summary of Procedure Module

プロシージャ名	タイプ	入出力	概要
CrackSRows	プロパティ	Get/Set	統計用のき裂テーブルの行数の取得/設定
Max3	関数		3つの引数から最小のものを選択
Min3	関数		3つの引数から最大のものを選択
TargetCols	プロパティ	Get/Set	対象の計算テーブルのフィールド数の取得/設定(未使用)
TargetRows	プロパティ	Get/Set	対象の計算テーブルの行数の取得/設定
Exec	メソッド		補間処理の実行

5.2.3 処理

き裂データの座評点と計算データ(解析、表計算、各損傷値)の座評点は必ずしも(ほとんど)一致していることはない。補間処理では、計算データを統計用のき裂データに合わせ、計算データの各フィールドを補間処理し、その結果を統計用の計算テーブルに格納する。

Exec メソッドには以下に示す 3つの引数がある。

Table 5.7 Summary of Excel Method Parameter

番号	引き数名	概要
1	CrkS (600,3)	統計用き裂データが入った2次元配列 第1引数: コート、第2引数: フィールド CrkS(i, 0): ライン番号 CrkS(i, 1): L CrkS(i, 2): id
2	Tgt (3,300,100)	評価時点1の各データが入った2次元配列 第1引数: ライン番号、第2引数: コート、第3引数: フィールド
3	TgtS (600,100)	評価時点2の各データが入った2次元配列 第1引数: コート、第2引数: フィールド

Exec メソッドが実行されると、CrkS,Tgt の配列データから補間処理計算を行い、結果を TgtS 配列に格納する。

5.2.4 INTPSV の実行例

以下に、Visual Basic からの INTPSV.EXE の実行例を示す。INTPSV.EXE のクラス名は”INTPSV.INTP”になる。

```
Dim CrkS(600, 3) As Variant
Dim Tgt(3, 300, 100) As Variant
Dim TgtS(600, 100) As Variant
Dim obj As Object

Set obj = CreateObject("INTPSV.TCALC")

obj.CrackSRows = 10                '統計用き裂が10ノードと仮定

obj.TargetRows = 100              '対象の計算データが10ノードと仮定
obj.TargetCols = 20 + 1          '対象の計算データが10フィールドと仮定

'統計用き裂データをCrkS配列へ格納

'対象の計算データをTgt配列へ格納

obj.Exec CrkS, Tgt, TgtS

'TgtS配列(補間結果)のデータを対応した統計用のテーブルに格納

Set obj = Nothing
```

5.3 損傷プログラム

損傷値計算は、STAR 本体とは独立した DAMAGE96.OCX が行う。この DAMAGE96.OCX は損傷値計算の機能を部品化した OLE カスタムコントロールである。処理の手順としては、STAR から試験部分の試験条件データおよび解析・時点データを DAMAGE96.OCX に転送し、DAMAGE96 では転送されたデータに対して損傷値計算を行う。STAR は DAMAGE96 から計算結果を読み込み、STAR データベースの各損傷値テーブルに格納する。データの転送および計算は行単位で行われる。

5.3.1 プロシージャ

DAMAGE96.OCX の主なプロパティとメソッドを以下に示す。

Table 5.8 summary of Procedure Module “DAMAGE96.OCX”

プロシージャ名	タイプ	入出力	概要
SetMethod	メソッド		評価法の番号を設定
SetTestCondition	メソッド		試験条件データをDAMAGE96へ転送
SetRangeAnalysis	メソッド		解析(Range)データをDAMAGE96へ転送
CalcDamage	メソッド		損傷値計算の実行
GetDamageResult	メソッド		計算結果をDAMAGE96から転送

以下に、メソッドの概要を示す。

- Long SetMethod(Long method_id)

metho_id: 評価法の番号

Table 5.9 Evaluation Method Table

番号	評価法	番号	評価法
1	TTSDS	8	EQEF
2	BDS	9	EQEFWELD
3	BDSMD	10	(PNC)
4	DDS	11	EQEF15
5	DDSWELD	12	EQEF167
6	(TYPE1)	13	INELAST
7	(TYPE2)	14	(SIEQ167)

(注)括弧付きの評価法はSTARでは使用できない。

- Long SetTestCondition(Long la, Double da)

la: 試験条件を収めた配列(整数型)

Table 5.10 Explanation of Module Parameter "SetTestCondition"

番号	内容
0	材料番号
1	サイクル数

da: 試験条件を収めた配列(実数型)

Table 5.11 Explanation of Module Parameter "SetTestCondition"

番号	内容
0	Cold温度
1	Hot温度
2	Cold保持時間
3	Hot保持時間
4	機械荷重

- Long SetRangeAnalysis(Double dh, Double dr1, Double dr2)

dh: 解析データを収めた配列

Table 5.12 Explanation of Module Parameter "SetRangeAnalysis"

番号	内容
0	最高金属温度
1	Q値

dr1, dr2: 時点1および2のデータの配列

Table 5.13 Explanation of Module Parameter "SetRangeAnalysis"

番号	内容	番号	内容
0	表面温度	15	弾性歪 ϵ_r
1	等価線形温度	16	弾性歪 ϵ_z
2	平均温度	17	弾性歪 ϵ_θ

3	表面応力 σ_r	18	弾性歪 γ_{rz}
4	表面応力 σ_z	19	塑性歪 ϵ_{pr}
5	表面応力 σ_θ	20	塑性歪 ϵ_{pz}
6	表面応力 τ_{rz}	21	塑性歪 $\epsilon_{p\theta}$
7	等価線形応力 $L\sigma_r$	22	塑性歪 γ_{prz}
8	等価線形応力 $L\sigma_z$	23	ｸﾘ-7°歪 ϵ_{cr}
9	等価線形応力 $L\sigma_\theta$	24	ｸﾘ-7°歪 ϵ_{cz}
10	等価線形応力 $L\tau_{rz}$	25	ｸﾘ-7°歪 $\epsilon_{c\theta}$
11	平均応力 $M\sigma_r$	26	ｸﾘ-7°歪 γ_{crz}
12	平均応力 $M\sigma_z$		
13	平均応力 $M\sigma_\theta$		
14	平均応力 $M\tau_{rz}$		

・ Long CalcDamage(Void)

引数なし:

・ Long GetDamageResult(Double daResult)

daResult: 計算結果データの配列

Table 5.14 Explanation of Module Parameter "GetDamageResult"

番号	内容	番号	内容
0	表面応力 S_p	12	溶接金属に対する弾性追従係数 Q_w
1	等価線形応力 S_n	13	局所的弾性追従係数 Q_l
2	初期応力 S_i	14	多軸効果を現す弾性追従係数 Q_{nu}
3	応力集中係数 K	15	弾性追従係数 Q_{eff}
4	K_e 係数 K_{ed}	16	弾性追従係数 Q_f
5	歪集中係数 K_e	17	母材に対する弾性追従係数 Q_{bm}
6	総体的歪集中係数 K_{edg}	18	Neuber則 NEUBER
7	局所的歪集中係数 K_{edl}	19	疲労損傷値 D_f
8	歪集中係数 K_p	20	ｸﾘ-7°損傷値 D_c
9	熱比°-ｸ歪 E_f	21	疲労ｸﾘ-7°損傷値 D
10	全歪範囲 E_t	22	許容値 D_{cr}
11	弾性追従係数 Q_n		

5.3.2 DAMAGE96の例

DAMAGE96.OCX を使用する場合は、この OCX をフォームに貼り付けて使用する。以下に STAR から(コンテナアプリケーション)から損傷プログラムをコントロールする例を示す。

```
Dim lRet As Long
Dim metho_id As Long
Dim la(2) As Long
Dim da(5) As Double
Dim dh(2) As Double
Dim dr1(27) As Double
Dim dr2(27) As Double
Dim DaResult(23) As Double
```

```
method_id = 2 'BDS
lRet = m_theDMG.SetMethod(method_id)
```

試験条件データを読み込みデータをlaとdaに格納

lRet = m_theDMG.SetTestCondition(la, da)

For i =すべての行に対して

解析データと時点データを読み込みdh,dr1,dr2に格納

lRet = m_theDMG.SetRangeAnalysis(dh, dr1, dr2)

lRet = m_theDMG.CalcDamage

lRet = m_theDMG.GetDamageResult(daResult)

daResultのデータを対応した損傷値テーブルに格納

Next i

6. オンラインヘルプ

STAR にはオンラインヘルプが用意されている。オンラインヘルプは[ヘルプ]メニューの[インデックス]コマンドからアクセスでき、STAR の各コマンドの概要を調べることができる。また、現在表示されているダイアログのどれかのコントロールにマウスを合せて右ボタンをおすと、そのコントロールについての説明が表示される。

この章ではヘルプファイルを作成する際の手順等について説明する。

6.1 ヘルプファイルの概要

STAR のヘルプファイルは STAR.HLP というファイル名になるが、以下の 2 つのソースファイルから作成される。

Table 6.1 Summary of Help File “STAR.HLP”

ファイル名	概要
STAR.RTF	オンラインヘルプのソースファイル。MS-WORD のリッチテキスト形式。
STAR.HPJ	オンラインヘルプのソースファイルをコンパイルする際のプロジェクトファイル。

ヘルプファイルは MS-WORD のリッチテキスト機能を使って作成する。MS-WORD ではヘルプファイルを作成するための様々な機能が用意されているが、STAR のヘルプファイルではそれらの機能の一部を使っているにすぎません。Visual Basic の各コントロールには WhatsThisHelpID というプロパティが用意されており、ここにはこのコントロール固有のヘルプ用の番号を記述する。そして、ヘルプファイルの中にこのコントロールの番号に対応したメッセージを記述するということになる。これがヘルプファイルの基本である。他にも検索機能、ヒストリ機能、移動(>>,<<)といった機能をヘルプファイルに持たせることができるが、STAR では使っていない。

6.2 ヘルプの番号

Visual Basic の各コントロールには WhatsThisHelpID というプロパティが用意されており、ここにはこのコントロール固有のヘルプ用の番号を記述する。以下に、STAR の各コントロールに割当てた番号を示す。

Table 6.2 Explanation of Help Number “WhatsThisHelpID”

コマンド名	ファイル名	ヘルプ番号	コントロール 1	コントロール 2	コントロール 3
試験部分選択	S_BUBUN2.FRM	10102	lblBubun	outBubun	
		10103	cmdOK		
		10104	cmdCancel		
解析ケース選択	S_CASE2.FRM	10202	lblCase	lstCase	
		10203	cmdOK		
		10204	cmdCancel		
選択データ内容確認	S_VIEW.FRM	10302	lblSochi	txtSochi	

		10304	lblTai	txtTai	
		10306	lblBubun	txtBubun	
		10308	lblCase	txtCase	
		10309	cmdOK		
き裂補間処理	I_CRK_IP.FRM	20102	lblError	lstError	
		20103	cmdOK		
試験条件入力/確認	I_CND_RG.FRM	20202	lblSochi	txtSochi	
		20204	lblTai	txtTai	
		20206	lblMATid	cmbMATid	
		20208	lblBui	txtBui	
		20210	lblTCold	txtTCold	
		20212	lblHCold	txtHCold	
		20214	lblTHot	txtTHot	
		20216	lblHHot	txtHHot	
		20218	lblMLoad	txtMLoad	
		20220	lblCycle	txtCycle	
		20222	lblNOData	txtNOData	
		20224	lblRefName	txtRefName	
		20226	lblRefNo	txtRefNo	
		20228	lblRefDS	txtRefDS	
		20230	lblSMAT	txtSMAT	
		20232	lblMemo	txtMemo	
		20233	cmdOK		
		20234	cmdClose		
き裂/温度/形状データ入力	I_VW_DBG.FRM	20602	dbgSourceTBL		
		20603	cmdOK		
		20604	cmdCancel		
形状データ入力	I_VW_DXF.FRM	20501	grdDXF		
		20502	cmdOK		
		20503	cmdCancel		
き裂/温度/形状データ確認 解析結果/表計算/損傷値確認	I_VW_GRD.FRM	20300	frmPriViewGRD		
		20301	grdView		
		20302	txtEdit		
き裂/温度/形状データ入力	X_SELDB1.FRM	20401	lblFile		
		20402	lblCurFile		
		20403	filFileLst		
		20404	lblDirectory		
		20405	lblCurDir		
		20406	dirDirList		
		20408	lblDataRange	lstDataRange	
		20410	lblFileType	cmdFileType	
		20412	lblDrive	drvDrive	
		20413	cmdOK		
		20414	cmdCancel		
解析結果入力	X_SELDB3.FRM	30101	lblLine1		
		30102	lblCurFile1		
		30103	filFile1		
		30104	lblLine2		
		30105	lblCurFile2		
		30106	filFile2		
		30107	lblLine3		
		30108	lblCurFile3		
		30109	filFile3		
		30111	lblRangel	cmbRangel	
		30113	lblRange2	cmbRange2	
		30115	lblRange3	cmbRange3	
		30117	lblDirectory	dirDirectory	

		30119	lblFileType	cmbFileType	
		30121	lblDrive	drvDrive	
		30122	cmdOK		
		30123	cmdCancel		
//	N_ANL_CS.FRM	30202	lblCase	lstCase	
		30203	cmdOK		
		30204	cmdCancel		
表計算	N_TBL_CT.FRM	30302	lstTerget		
		30304	cmdOK		
		30305	cmdCancel		
損傷評価法選択	N_DNG_SL.FRM	30402	lblUnselected	lstUnselected	
		30404	lblSelected	lstSelected	
		30405	cmdSel		
		30406	cmdUnsel		
		30407	cmdOK		
		30408	cmdCancel		
損傷値	N_DMG_CT.FRM	30502	lblterget	lstTerget	
		30504	cmdOK		
		30505	cmdCancel		
損傷値確認	N_DMG_VS.FRM	30602	lblHyoka	lstHyoka	
		30603	cmdOK		
		30604	cmdCancel		
Qの計算	N_Q_SEL.FRM	30702	lblElas	cmbElas	
		30704	lblPlas	cmbPlas	
		30706	lblSave	cmbSave	
		30707	cmdOK		
		30708	cmdCancel		
解析情報入力/確認	N_FNS_VW.FRM	30801	grdFinas		
		30802	txtEdit		
		30803	cmdOK		
		30804	cmdCancel		
		30805	cmdClose		
解析結果入力	N_ANL_VW.FRM	30905	dbgAna11		
		30906	dbgAna12		
		30907	dbgAna13		
		30908	cmdOK		
		30909	cmdCancel		
分布比較処理	B_SELECT.FRM	40102	fraQueryDB	txtQueryDB	
		40103	cmdQDBChange		
		40105	fraLine	cmbLine	
		40107	fraQuery	cmbQuery	
		40108	cmdOK		
		40109	cmdCancel		
補間処理	T_INT_SL.FRM	50101	fraTerget	optAnal	opttable
			optDamage		
		50105	cmdDmgSel		
		50106	cmdOK		
		50107	cmdCancel		
//	T_INT_CT.FRM	50202	lblTerger	lstTerget	
		50204	cmdOK		
		50205	cmdCancel		
統計用データ確認	T_DAT_SL.FRM	50301	fraTerget	optCrk	optAnl
			optTbl	optDmg	
		50306	cmbDmgSel		
		50307	cmdOK		
		50308	cmdCancel		
統計処理	T_SELECT.FRM	50402	fraQueryDB	txtQueryDB	

		50403	cmdQDBChange		
		50407	fraQuery	cmbQuery	
		50408	cmdOK		
		50409	cmdCancel		
管理データの登録	U_KANRI2.FRM	60102	lblBubun	outBubun	
		60103	cmdAdd		
		60104	cmdDelete		
		60105	cmdExit		
//	U_KNR_A2.FRM	60203	lblSochi	txtSochi	
		60205	lblTai	txtTai	
		60207	lblBubun	txtBubun	
		60209	lblLine	cmbLine	
		60211	lblID	txtID	
		60212	cmdReg		
		60203	cmdCancel		
解析ケースの登録	U_CASE.FRM	60302	lblCase	lstCase	
		60303	cmdAdd		
		60304	cmdDelete		
		60305	cmdExit		
//	U_CAS_AD.FRM	60402	fraTemp	optTempJkn	optTempDgn
		60405	fraTrns	optTrnsJkn	optTrnsDgn
		60408	fraMesh	optMeshDtl	optMeshDtl
		60412	lblMesh	txtMesh	
		60413	lstMesh		
		60414	cmdOK		
		60415	cmdCancel		
損傷評価法の登録	U_HYOKA.FRM	60502	lblHyoka	lstHyoka	
		60503	cmdAdd		
		60504	cmdDelete		
		60507	cmdExit		
//	U_HYK_AD.FRM	60602	lblName	txtName	
		60604	lblBiko	txtBiko	
		60606	lblID	txtID	
		60609	lblTemplate	txtTemplate	
		60611	lblRange	txtRange	
		60612	cmdBrowse		
		60613	cmdOK		
		60614	cmdClose		
材料の登録	U_MAT.FRM	60702	lblMat	lstMat	
		60703	cmdAdd		
		60704	cmdDelete		
		60705	cmdExit		
		60707	lblNewMat	txtNewMat	
		60709	lblNewBiko	txtNewBiko	
		60711	lblNewID	txtNewID	
		60712	cmdReg		
		60713	cmdCancel		
環境設定	U_SETUP.FRM	60802	lblDSN	txtDSN	
		60804	lblDBN	txtDBN	
		60806	lblQDB	txtQDB	
		60807	cmdBrowse		
		60809	lblBTView	cmbBTView	
		60818	lblMaxRet	txtMaxRet	
		60811	lblWorkDir	txtWorkDir	
		60813	lblWSketchDir	txtWSketchDir	
		60815	cmdOK		
		60816	cmdCancel		

SQL	U_SQL.FRM	60902	lblSQL	txtSQL	
		60903	chkSelect		
		60904	cmdOK		
		60905	cmdCancel		
		60906	grdSelect		
起動時	X_USER2.FRM	70102	lblUID	txtUID	
		70104	lblPWD	txtPWD	
		70105	cmdOK		
		70106	cmdQuid		
		70107	cmdSetup		
常時	X_NORMAL.FRM	70202	lblSochiX	lblSochi	
		70204	lblTaiX	lblTai	
		70206	lblBubunX	lblBubun	
		70208	lblCaseX	lblCase	
分布比較・統計	X_RESGRD.FRM	70400			
		70401	grdRes		

STAR には各コントロールに割当てたヘルプ以外にも、メニューバーから参照する階層構造を持ったヘルプも用意されている。以下にメニューヘルプの番号を示す。

Table 6.3 Explanation of Help Number

ヘルプ番号			内容
階層 0	階層 1	階層 2	
0			[ヘルプ]メニューの[インテックス]コマンドが実行されたときに表示。
	1000		[データ選択]メニューについて
		1010	[試験部分選択]コマンドについて
		1020	[解析ケース選択]コマンドについて
		1030	[管理データ内容確認]コマンドについて
		1040	[試験部分形状図]コマンドについて
		1050	[STAR 終了]コマンドについて
	2000		[実験データ]メニューについて
		2010	[き裂データ入力]コマンドについて
		2020	[き裂データ確認]コマンドについて
		2030	[き裂補間処理]コマンドについて
		2040	[試験条件入力]コマンドについて
		2050	[試験条件確認]コマンドについて
		2060	[温度データ入力]コマンドについて
		2070	[温度データ確認]コマンドについて
		2080	[形状データ入力]コマンドについて
		2090	[形状データ確認]コマンドについて
	3000		[解析データ]メニューについて
		3010	[解析結果入力]コマンドについて
		3020	[解析結果確認]コマンドについて
		3030	[表計算/全テーブル]コマンドについて
		3040	[表計算/現試験体]コマンドについて
		3050	[表計算/現解析ケース]コマンドについて
		3060	[表計算確認]コマンドについて
		3070	[損傷評価法の選択]コマンドについて
		3080	[損傷値/全テーブル]コマンドについて
		3090	[損傷値/現試験体]コマンドについて
		3100	[損傷値/現解析ケース]コマンドについて
		3110	[損傷値確認]コマンドについて

		3120	[Qの計算]コマンドについて
		3130	[解析情報入力]コマンドについて
		3140	[解析情報確認]コマンドについて
	4000		[分布比較]メニューについて
		4010	[分布比較処理]コマンドについて
	5000		[統計]メニューについて
		5010	[き裂データ抽出]コマンドについて
		5020	[補間処理/全テーブル]コマンドについて
		5030	[補間処理/現試験体]コマンドについて
		5040	[補間処理/現解析ケース]コマンドについて
		5050	[統計用データ確認]コマンドについて
		5060	[統計処理]コマンドについて
	6000		[STAR 制御]メニューについて
		6010	[管理データの一覧]コマンドについて
		6020	[管理データの登録]コマンドについて
		6030	[解析ケースの登録]コマンドについて
		6040	[損傷評価法の登録]コマンドについて
		6050	[材料の登録]コマンドについて
		6060	[環境設定]コマンドについて
		6070	[SQL]コマンドについて
		6080	[STAR 終了]コマンドについて

6.3 リッチテキストファイル

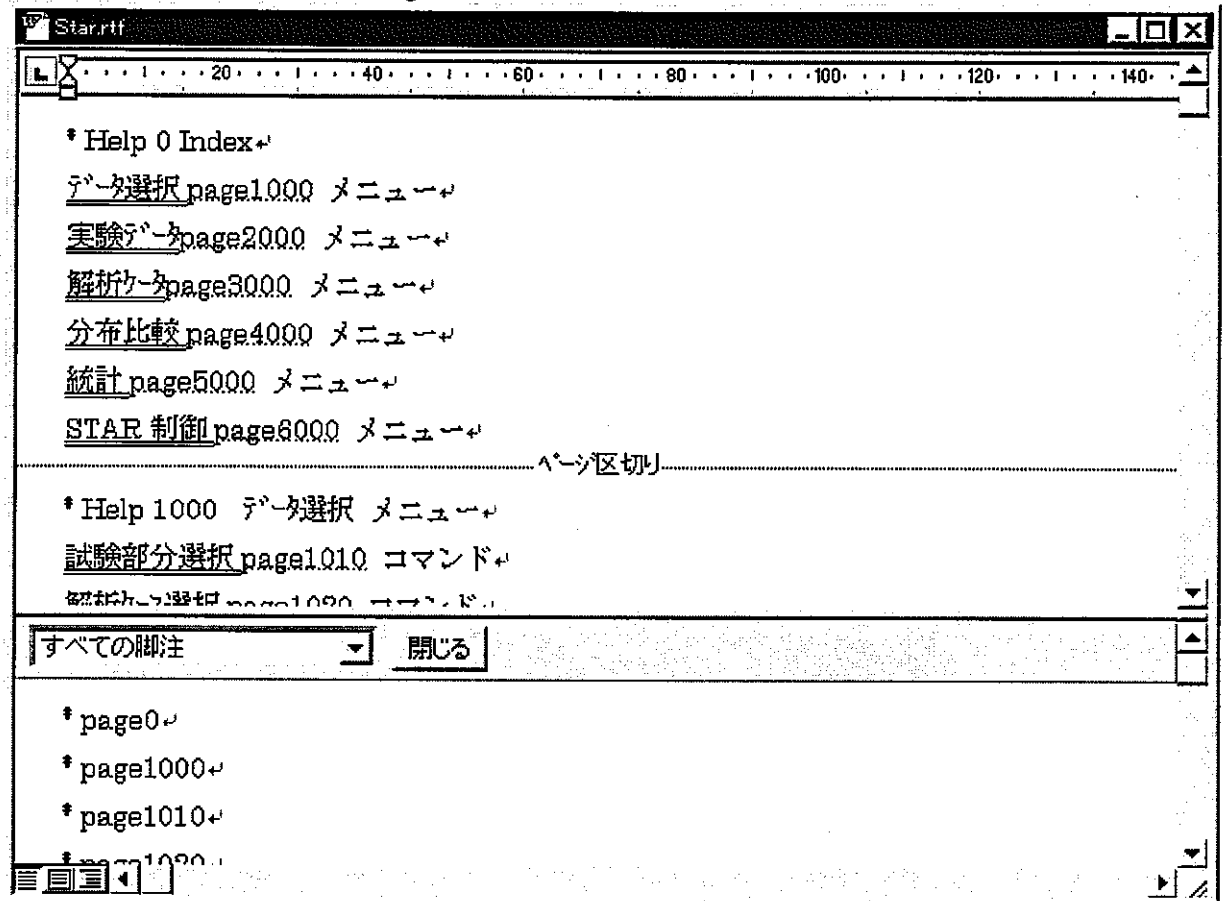
ヘルプファイルは MS-WORD を使って作成する。データを保存する際には、ファイルの種類をリッチテキスト形式(RTF)にする。MS-WORD で単に文章を記述しただけでは、そのファイルをヘルプファイルとして使うことはできない。文章を記述する際にいくつかの決まりごとがある。つまり、文章の中にアプリケーションからヘルプを参照できるようにするための各種の属性を設定しなければならない。以下にヘルプファイルを記述する際の注意を示す。

1. 準備: [ツール]/[オプション]コマンドを実行し、表示タグの”隠し文字”にチェックマークを付ける。
2. 脚注番号の設定: 各ページの先頭にキャレットを移動し、[挿入]/[脚注]コマンドを実行する。”脚注と文末脚注”ダイアログの”任意設定の脚注番号”に半角文字の”#”を入力して、OK ボタンを押す。WORD のウィンドウの下部に脚注用の編集領域が表示されるので、”#”マークの後にそのページの名前(page1, page2 等)を記述する。本文に”#”マークが付加され脚注が設定されたことがわかる。
3. ジャンプの設定: ジャンプを現す文字列を強調表示させ、[書式]/[文字]コマンドを実行する。文字ダイアログのフォントタグの”下線”コンボボックスから”二重線”を設定し、OK ボタンを押す。強調表示した文字列に二重下線が付加される。次に具体的な飛び先を指定する。今の文字列の一番後ろにキャレットを移動し、[書式]/[文字]コマンドを実行する。文字ダイアログのフォントタグで下線なしに設定し、さらに”隠し文字”にチェックマークを付け、OK ボタンを押す。次に、そのキャレ

トの位置の飛び先のページ名(page1, page2 等)を入力する。破線がついた文字が表示されるはずである。

上記の作業をすべてのページ、すべてのジャンプに対して設定する。以下に STAR.RTF の編集画面の例を示す。

Fig 6.1 Edit Menu "STAR.RTF"



6.4 プロジェクトファイル

メニューファイルをコンパイルする際に、コンパイルの手順や条件を記述したプロジェクトファイルを使用する。リッチテキストファイルの 1 ページが各コントロールに対応したヘルプになる。コントロールにはヘルプを参照するためにそのヘルプの番号(数値)を記述し、リッチテキストファイルには、各ページに固有の脚注番号を定義している。プロジェクトファイルではコントロールのヘルプ番号とリッチテキストファイルの脚注番号の関連付けを行っている。以下にプロジェクトファイルの一部を示す。

```
#define page10201 10201
#define page10202 10202
```

```
#define page10203 10203
```

```
#define page10204 10204
```

1行目の例は、Visual Basic から 10201 版のヘルプの要求が発生した場合に、リッチテキストファイルの脚注番号 page10201 を表示することを示している。

6.5 ヘルプファイルのコンパイル

ヘルプコンパイラは HC31.EXE を使用する。このコンパイラは Visual Basic に用意されている。コンパイルはいたって簡単である。リッチテキストファイル(STAR.RTF)とプロジェクトファイル(STAR.HPJ)をどこかのディレクトリに作成し、DOS プロンプトから以下のコマンドを実行するだけである。

```
HC31 STAR.HPJ
```

コンパイルが終了するとヘルプファイル(STAR.HLP)が作成されるので、STAR がインストールされたディレクトリへヘルプファイルをコピーする。

7. デジタイザ入力システム

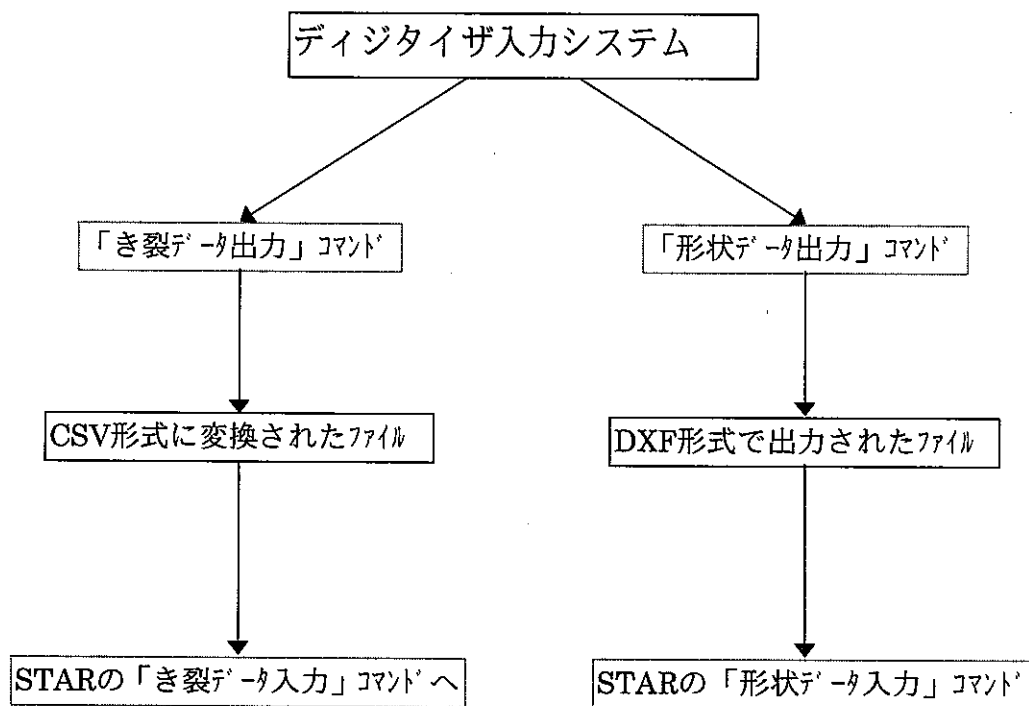
STARバージョン7では、デジタイザ入力システムはWINDOWS版のAutoCAD(AutoCAD R13 for Windows)に対応している。AutoCAD で出力したき裂や形状データの DXF ファイルを AutoSketch で読み込むことが可能である。ただし、AutoSketch の DXF ファイル読み込みの仕様がサブセットであるため、AutoCAD で作成した DXF ファイルが必ずしも AutoSketch で読めるわけではない。AutoCAD で図を作成する際には注意が必要である。

なお、一般ユーザはデジタイザ入力システムを扱うことはできない。

7.1 出力ファイルの流れ

デジタイザ入力システムで入力した形状およびき裂データファイルの STAR システムでの位置づけを以下に示す。

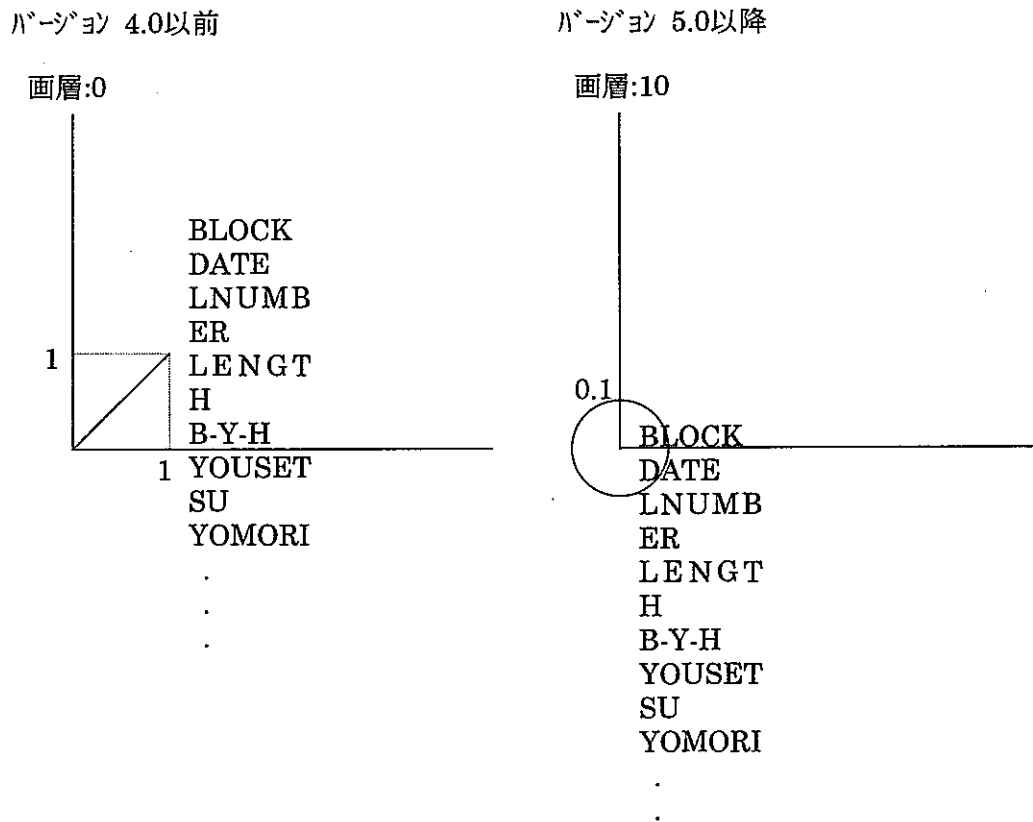
Fig 7.1 Output File Flow



7.2 BASEBLK.DWGの仕様

BASEBLK.DWG ファイルの仕様は STAR 5.0 から以下のように変更されている。

Fig 7.2 Specification of Shape File “BASEBLK.DWG”



謝 辞

管理プログラムのシステム設計、プログラム設計及びコーディングについては株式会社CRC総合研究所 本橋 賀津彦 氏に多大なるご協力をいただきました。ここに謝意を表します。

参考文献

- (1) マイクロソフト㈱ 『Microsoft Windows 95 ファーストステップガイド』, (1995)
- (2) マイクロソフト㈱ 『Microsoft Office for Windows 95 活用ガイド』, (1995)
- (3) マイクロソフト㈱ 『Microsoft Access for Windows 95 活用ガイド』, (1995)
- (4) マイクロソフト㈱ 『Microsoft Access for Windows 95 アプリケーション開発ガイド』, (1995)
- (5) マイクロソフト㈱ 『Microsoft Visual Basic 4.0 プログラミングガイド』, (1995)
- (6) マイクロソフト㈱ 『Microsoft Visual Basic 4.0 ランゲージリファレンス』, (1995)
- (7) マイクロソフト㈱ 『Microsoft Visual Basic 4.0 プロフェッショナルプログラミングガイド』, (1995)
- (8) マイクロソフト㈱ 『Microsoft Visual Basic 4.0 カスタムコントロールリファレンス』, (1995)
- (9) マイクロソフト㈱ 『Microsoft Visual Basic 4.0 クライアントサーバアプリケーション作成ガイド』, (1995)
- (10) オートデスク ㈱ 『AutoCAD Release R13 ユーザリファレンス』, (1995)
- (11) オートデスク ㈱ 『AutoCAD Release R13 コマンドリファレンス』, (1995)
- (12) オートデスク ㈱ 『AutoCAD Release R13 カスタムガイド』, (1995)
- (13) オートデスク ㈱ 『AUTOSKETCH FOR WINDOWS ユーザリファレンス』, (1995)
- (14) オートデスク ㈱ 『AUTOSKETCH FOR WINDOWS 操作ガイド』, (1995)
- (15) ㈱翔泳社 『ビジュアルベシックマガジン vol.1』 “Windowsヘルプファイル作成とヘルプガード”, (1995)
- (16) 啓学出版㈱ 『標準SQLプログラミング』, (1995)
- (17) マイクロソフト㈱ 『SQL Server リファレンス』, (1995)
- (18) 笠原他、 『高速炉機器設計用構造物強度データベースの開発』 原子力学会、
1990年秋の大会, (1990)
- (19) 笠原他、 『構造物の熱応力強度データベースSTARの開発』 機械学会、
第68期通常総会, (1991)
- (20) N.Kasahara et al., 『Database System of Structural Strength Tests to Validate Design
Methods of FBR Components』, Trans.of SmiRT11, SD1

付 録

A. STAR本体ソースリスト

STAR. EXE

```

Attribute VB_Name = "frmBunSelect"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Option Explicit

```

```
Private Sub cmbLine_Click()
```

```

    If cmbLine.ListIndex >= 0 And cmbQuery.ListIndex >= 0 Then
        cmdOK.Enabled = True
    End If

```

```
End Sub
```

```
Private Sub cmbQuery_Click()
```

```

    If cmbLine.ListIndex >= 0 And cmbQuery.ListIndex >= 0 Then
        cmdOK.Enabled = True
    End If

```

```
End Sub
```

```
Private Sub cmdCancel_Click()
```

```

    modSTAR.tmpFormCloseType = vbCancel
    Unload Me

```

```
End Sub
```

```
Private Sub cmdCancel_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
```

```

    If Button = 2 Then cmdCancel.ShowWhatsThis

```

```
End Sub
```

```
Private Sub cmdOK_Click()
```

```

    On Error GoTo Err1

```

```

    If cmbLine.ListIndex < 0 Then
        MsgBox "評価ライン番号が指定されていません。"
        Exit Sub
    End If

```

```

    If cmbQuery.ListIndex < 0 Then
        MsgBox "分布比較方法が選択されていません。"
        Exit Sub
    End If

```

```

    modSTAR.CurBunLineNo = cmbLine.ListIndex
    modSTAR.CurBunQueryNo = cmbQuery.ListIndex
    modSTAR.CurBunQueryName$ = cmbQuery.List(cmbQuery.ListIndex)

```

```

    modSTAR.tmpFormCloseType = vbOK

```

```

    Unload Me

```

```

    Exit Sub

```

```
Err1:
```

```

    MsgBox "Error in frmBunSelect/cmdOK"
    Call ShowErrorMessage
    modSTAR.tmpFormCloseType = vbCancel

```

```
End Sub
```

```
Private Sub cmdOK_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
```

```

    If Button = 2 Then cmdOK.ShowWhatsThis

```

```
End Sub
```

```
Private Sub cmdQDBChange_Click()
```

```

    Dim dbQuery As Database
    Dim i As Long
    Dim LCount As Long
    Dim Buf$

```

```

    CommonDialog1.Filter = "Access (*.MDB)|*.mdb"

```

```

CommonDialog1.Flags = cdIOFNReadOnly

On Error Resume Next
CommonDialog1.ShowOpen
On Error GoTo 0

' 復帰ボタンのチェック
If Err.Number <> 0 Then
    If Err.Number = 32755 Then ' ユーザがキャンセルを選択
        Exit Sub
    Else
        MsgBox "[ファイルを開く]ダイアログボックスで予期しないエラーが発生しました。"
        Exit Sub
    End If
End If

If CommonDialog1.FileName <> "" Then

    ' 入力ファイル名の取得
    txtQueryDB.Text = CommonDialog1.FileName
    modSTAR.QueryDB$ = CommonDialog1.FileName

    ' cmbQueryリストボックスの更新
    LCount = cmbQuery.ListCount
    For i = LCount - 1 To 0 Step -1
        cmbQuery.RemoveItem i
    Next i

    Set dbQuery = Workspaces(0).OpenDatabase(modSTAR.QueryDB$)
    For i = 0 To dbQuery.QueryDefs.Count - 1
        Buf$ = dbQuery.QueryDefs(i).Name
        If Left$(Buf$, 2) = "分布" Then
            cmbQuery.AddItem dbQuery.QueryDefs(i).Name
        End If
    Next i
    dbQuery.Close

End If
End Sub

Private Sub cmdQueryDelete_Click()
    Dim dbQuery As Database
    Dim dtBUN_TTL As Recordset
    Dim Ret As Long
    Dim QName$, TName$, SQL$
    '
    '
    ' If cmbQuery.ListIndex < 0 Then Exit Sub
    '
    '
    ' TName$ = cmbQuery.List(cmbQuery.ListIndex)
    ' Ret = MsgBox("「" & TName$ & "」を削除します。よろしいですか?", vbYesNo + vbDefaultButton2)
    ' If Ret = vbYes Then
    '     QName$ = modSTAR.QueryDB$
    '     SQL$ = "SELECT * FROM 分布比較 WHERE (タイトル = '" & TName$ & "');"
    '     Set dbQuery = Workspaces(0).OpenDatabase(QName$)
    '     Set dtBUN_TTL = dbQuery.OpenRecordset(SQL$, dbOpenDynaset)
    '     If dtBUN_TTL.RecordCount > 0 Then
    '         dtBUN_TTL.Delete
    '         cmbQuery.RemoveItem cmbQuery.ListIndex
    '         cmbQuery.ListIndex = -1
    '     End If
    '     dtBUN_TTL.Close
    '     dbQuery.Close
    ' End If
End Sub

Private Sub cmdQueryEdit_Click()
    ' If cmbQuery.ListIndex >= 0 Then
    '     modSTAR.tmpBunQName$ = cmbQuery.List(cmbQuery.ListIndex)
    '     frmBunDesign.Show 1
    ' End If
End Sub

Private Sub cmdQDBChange_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then cmdQDBChange.ShowWhatsThis

```

End Sub

```
Private Sub Form_Load()
    Dim dbQuery As Database
    Dim i As Long
    Dim QName$, Buf$

    On Error GoTo Err1

    QName$ = modSTAR.QueryDB$

    ' 検索条件テキストの初期値
    txtQueryDB.Text = QName$

    ' 評価ラインのリストを作成
    For i = 1 To modSTAR.NoOfLine
        cmbLine.AddItem "LINE-" & i
    Next i
    cmbLine.ListIndex = modSTAR.CurBunLineNo

    ' 検索条件のリストを作成
    If QName$ <> modSTAR.StarDB$ Then
        Set dbQuery = Workspaces(0).OpenDatabase(QName$)
    Else
        Set dbQuery = dbSTAR
    End If

    For i = 0 To dbQuery.QueryDefs.Count - 1
        Buf$ = dbQuery.QueryDefs(i).Name
        If Left$(Buf$, 2) = "分布" Then
            cmbQuery.AddItem dbQuery.QueryDefs(i).Name
        End If
    Next i

    If QName$ <> modSTAR.StarDB$ Then
        dbQuery.Close
    End If
    cmbQuery.ListIndex = modSTAR.CurBunQueryNo

    If cmbLine.ListIndex < 0 Or cmbQuery.ListIndex < 0 Then
        cmdOK.Enabled = False
    End If

    modSTAR.tmpFormCloseType = vbCancel

    Exit Sub
```

```
Err1:
    MsgBox "Error in frmBunSelect/Load"
    Call ShowErrorMessage
    modSTAR.tmpFormCloseType = vbCancel
    Unload Me ' 強制エラー
```

End Sub

```
Private Sub fraLine_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
```

```
    If Button = 2 Then fraLine.ShowWhatsThis
```

End Sub

```
Private Sub fraQuery_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
```

```
    If Button = 2 Then fraQuery.ShowWhatsThis
```

End Sub

```
Private Sub fraQueryDB_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
```

```
    If Button = 2 Then fraQueryDB.ShowWhatsThis
```

End Sub

```
Private Sub txtQueryDB_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
```

```
    If Button = 2 Then txtQueryDB.ShowWhatsThis
```

End Sub


```

Attribute VB_Name = "frmPriCondReg"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Option Explicit

Private MaxID As Long

Private Sub cmdCancel_Click()
    Unload Me
End Sub

Private Sub cmdCancel_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then cmdCancel.ShowWhatsThis
End Sub

Private Sub cmdOK_Click()
    Dim dtCOND As rdoResultset
    Dim SQL$
    Dim hCOND As Integer

    On Error GoTo Err1

    If txtSochi.Text = "" Then
        MsgBox "試験装置が入力されていません。"
        Exit Sub
    End If

    If txtTai.Text = "" Then
        MsgBox "試験体が入力されていません。"
        Exit Sub
    End If

    If cmbMATid.ListIndex = -1 Then
        MsgBox "材料名が入力されていません。"
        Exit Sub
    End If

    If txtBui.Text = "" Then
        MsgBox "評価部位が入力されていません。"
        Exit Sub
    End If

    If txtTCold.Text = "" Then
        MsgBox "COLD温度が入力されていません。"
        Exit Sub
    End If

    If txtHCold.Text = "" Then
        MsgBox "COLD保持時間が入力されていません。"
        Exit Sub
    End If

    If txtTHot.Text = "" Then
        MsgBox "HOT温度が入力されていません。"
        Exit Sub
    End If

    If txtHHot.Text = "" Then
        MsgBox "HOT保持時間が入力されていません。"
        Exit Sub
    End If

    If txtMLoad.Text = "" Then
        MsgBox "機械荷重が入力されていません。"
        Exit Sub
    End If

    If txtCycle.Text = "" Then
        MsgBox "サイクル数が入力されていません。"
        Exit Sub
    End If

    Screen.MousePointer = 11

    'データ用のテーブルから既存の試験条件テーブルを削除しておく
    SQL$ = "DELETE 試験条件 WHERE (参照id = " & modSTAR.nmID & ");"
    dbSTAR5.Execute SQL$

    '試験条件テーブルにデータを記入
    SQL = "SELECT * FROM 試験条件 WHERE "
    & "(参照id = " & modSTAR.nmID & ");"
    Set dtCOND = dbSTAR5.OpenResultset(SQL$, rdOpenDynamic, rdConcurLock)
    Set dtCOND = xOpenResultset(SQL$, rdOpenDynamic, rdConcurLock, hCOND)
    If hCOND = -1 Then GoTo Err1

    'If modSTAR.tmpCondInpCtrl$ = "REG" Then

```

```

' dtCOND.AddNew
Else
' dtCOND.Edit
End If
If modSTAR.tmpCondInpCtrl$ = "SHOW" Then
SQL$ = "DELETE 試験条件 WHERE (参照id = " & modSTAR.nmID & ");"
dbSTAR5.Execute SQL$
End If
dtCOND.AddNew
dtCOND![id] = MaxID
dtCOND![参照id] = modSTAR.nmID '参照id
dtCOND![試験装置] = txtSochi.Text '試験装置
dtCOND![試験体] = txtTai.Text '試験体
dtCOND![材料id] = cmbMATid.ListIndex + 1 '材料id
dtCOND![評価部位] = txtBui.Text '評価部位
dtCOND![COLD温度] = Val(txtTCold.Text) 'COLD温度
dtCOND![COLD保持時間] = Val(txtHCold.Text) 'COLD保持時間
dtCOND![HOT温度] = Val(txtTHot.Text) 'HOT温度
dtCOND![HOT保持時間] = Val(txtHHot.Text) 'HOT保持時間
dtCOND![機械荷重] = Val(txtMLoad.Text) '機械荷重
dtCOND![サイクル数] = Val(txtCycle.Text) 'サイクル数
dtCOND![テ-タ数] = Val(txtNOData.Text) 'テ-タ数
dtCOND![関連資料名] = txtRefName.Text '関連資料名
dtCOND![関連資料番号] = txtRefNo.Text '関連資料番号
dtCOND![関連テ-タット] = txtRefDS.Text '関連テ-タット
dtCOND![SMAT_TAG_NO] = txtSMAT.Text 'Smat Tag no.
dtCOND![メモ] = txtMemo.Text 'メモ
dtCOND.Update

'試験条件テ-プルのロード
dtCOND.Close
xCloseResultset dtCOND, hCOND

' Call SetKanriXID("試験条件id", modSTAR.nmID, modSTAR.nmID, "-", 0)
Call SetKanriXID("試験条件id", modSTAR.nmID, modSTAR.nmID, "-", 0)

Screen.MousePointer = 0

Unload Me
Exit Sub

Err1:
Screen.MousePointer = 0
MsgBox "Error in frmPriCondReg/cmdOK"
Call ShowErrorMessage
On Error Resume Next
dtCOND.Close
xCloseResultset dtCOND, hCOND

End Sub

Private Sub cmdOK_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
If Button = 2 Then cmdOK.ShowWhatsThis
End Sub

Private Sub Form_Load()
Dim dtCOND As rdoResultset
Dim dtMAT As rdoResultset
Dim SQL$
Dim vImp(5) As Variant
Dim hCOND As Integer
Dim hMAT As Integer

On Error GoTo Err1

Screen.MousePointer = 11

SQL$ = "SELECT 材料名 FROM 材料テ-プルのリスト ORDER BY id;"
' Set dtMAT = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
Set dtMAT = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hMAT)
If hMAT = -1 Then GoTo Err1

Do Until dtMAT.EOF
cmbMATid.AddItem dtMAT![材料名]
dtMAT.MoveNext
Loop

' dtMAT.Close
xCloseResultset dtMAT, hMAT

txtSochi.Text = ""
txtTai.Text = ""
cmbMATid.Text = ""
cmbMATid.ListIndex = -1
txtBui.Text = ""
txtTCold.Text = ""
txtHCold.Text = ""
txtTHot.Text = ""
txtHHot.Text = ""
txtMLoad.Text = ""

```

```

txtCycle.Text = ""
txtNOData.Text = ""
txtRefName.Text = ""
txtRefNo.Text = ""
txtRefDS.Text = ""
txtSMAT.Text = ""
txtMemo.Text = ""

If modSTAR.tmpCondInpCtrl$ = "REG" Then      '登録
Caption = "試験条件データの入力"
MaxID = GetMaxID("試験条件", "id") + 1

ElseIf modSTAR.tmpCondInpCtrl$ = "SHOW" Then '確認
Caption = "試験条件データの確認"

SQL$ = "SELECT * FROM 試験条件 WHERE "
      & "(参照id = " & modSTAR.nmID & ")";
' Set dtCOND = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
Set dtCOND = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hCOND)
If hCOND = -1 Then GoTo Err1

If dtCOND.EOF = False Then
MaxID = dtCOND![id]
txtSochi.Text = dtCOND![試験装置]
txtTai.Text = dtCOND![試験体]
cmbMATid.ListIndex = dtCOND![材料id] - 1
txtBui.Text = dtCOND![評価部位]
txtTCold.Text = dtCOND![COLD温度]
txtHCold.Text = dtCOND![COLD保持時間]
txtTHot.Text = dtCOND![HOT温度]
txtHHot.Text = dtCOND![HOT保持時間]
txtMLoad.Text = dtCOND![機械荷重]
txtCycle.Text = dtCOND![サイクル数]
txtNOData.Text = dtCOND![データ数]
txtRefName.Text = dtCOND![関連資料名]
txtRefNo.Text = dtCOND![関連資料番号]
txtRefDS.Text = dtCOND![関連データセット]
txtSMAT.Text = dtCOND![SMAT_TAG_NO]
txtMemo.Text = dtCOND![メモ]
vImp(1) = dtCOND![関連資料名]
vImp(2) = dtCOND![関連資料番号]
vImp(3) = dtCOND![関連データセット]
vImp(4) = dtCOND![SMAT_TAG_NO]
vImp(5) = dtCOND![メモ]
If IsNull(vImp(1)) = False Then txtRefName.Text = vImp(1)
If IsNull(vImp(2)) = False Then txtRefNo.Text = vImp(2)
If IsNull(vImp(3)) = False Then txtRefDS.Text = vImp(3)
If IsNull(vImp(4)) = False Then txtSMAT.Text = vImp(4)
If IsNull(vImp(5)) = False Then txtMemo.Text = vImp(5)

If modSTAR.SuperVisor = False Then
cmdOK.Enabled = False
cmdOK.Visible = False
cmdCancel.Caption = "閉じる"

txtSochi.Locked = True
txtTai.Locked = True
cmbMATid.Enabled = False
txtBui.Locked = True
txtTCold.Locked = True
txtHCold.Locked = True
txtTHot.Locked = True
txtHHot.Locked = True
txtMLoad.Locked = True
txtCycle.Locked = True
txtNOData.Locked = True
txtRefName.Locked = True
txtRefNo.Locked = True
txtRefDS.Locked = True
txtSMAT.Locked = True
txtMemo.Locked = True
End If

End If

'dtCOND.Close
xCloseResultset dtCOND, hCOND

End If

Screen.MousePointer = 0
Exit Sub

Err1:
Screen.MousePointer = 0
MsgBox "Error in frmPriCondReg/Load"
Call ShowErrorMessage
On Error Resume Next
'dtMAT.Close
'dtCOND.Close
xCloseResultset dtMAT, hMAT
xCloseResultset dtCOND, hCOND
Unload Me      '強制エラー発生

End Sub

```

```
Private Sub lblBui_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblBui.ShowWhatsThis
End Sub
```

```
Private Sub lblCycle_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblCycle.ShowWhatsThis
End Sub
```

```
Private Sub lblHCold_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblHCold.ShowWhatsThis
End Sub
```

```
Private Sub lblHHot_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblHHot.ShowWhatsThis
End Sub
```

```
Private Sub lblMATid_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblMATid.ShowWhatsThis
End Sub
```

```
Private Sub lblMemo_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblMemo.ShowWhatsThis
End Sub
```

```
Private Sub lblMLoad_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblMLoad.ShowWhatsThis
End Sub
```

```
Private Sub lblNOData_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblNOData.ShowWhatsThis
End Sub
```

```
Private Sub lblRefDS_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblRefDS.ShowWhatsThis
End Sub
```

```
Private Sub lblRefName_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblRefName.ShowWhatsThis
End Sub
```

```
Private Sub lblRefNo_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblRefNo.ShowWhatsThis
End Sub
```

```
Private Sub lblSMAT_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblSMAT.ShowWhatsThis
End Sub
```

```
Private Sub lblSochi_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
```

```
    If Button = 2 Then lblSochi.ShowWhatsThis
End Sub

Private Sub lblTai_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblTai.ShowWhatsThis
End Sub

Private Sub lblTCold_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblTCold.ShowWhatsThis
End Sub

Private Sub lblTHot_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblTHot.ShowWhatsThis
End Sub

Private Sub txtBui_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then txtBui.ShowWhatsThis
End Sub

Private Sub txtCycle_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then txtCycle.ShowWhatsThis
End Sub

Private Sub txtHCold_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then txtHCold.ShowWhatsThis
End Sub

Private Sub txtHHot_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then txtHHot.ShowWhatsThis
End Sub

Private Sub txtMemo_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then txtMemo.ShowWhatsThis
End Sub

Private Sub txtMLoad_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then txtMLoad.ShowWhatsThis
End Sub

Private Sub txtNOData_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then txtNOData.ShowWhatsThis
End Sub

Private Sub txtRefDS_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then txtRefDS.ShowWhatsThis
End Sub

Private Sub txtRefName_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then txtRefName.ShowWhatsThis
```

End Sub

Private Sub txtRefNo_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)

 If Button = 2 Then txtRefNo.ShowWhatsThis

End Sub

Private Sub txtSMAT_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)

 If Button = 2 Then txtSMAT.ShowWhatsThis

End Sub

Private Sub txtSochi_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)

 If Button = 2 Then txtSochi.ShowWhatsThis

End Sub

Private Sub txtTai_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)

 If Button = 2 Then txtTai.ShowWhatsThis

End Sub

Private Sub txtTCold_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)

 If Button = 2 Then txtTCold.ShowWhatsThis

End Sub

Private Sub txtTHot_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)

 If Button = 2 Then txtTHot.ShowWhatsThis

End Sub

```
Attribute VB_Name = "frmPriCrackInt"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Option Explicit
```

```
Private Sub cmdClose_Click()
    Unload Me
End Sub
```

```
Private Sub cmdClose_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then cmdClose.ShowWhatsThis
End Sub
```

```
Private Sub Form_Load()
    Dim dtANAL As rdoResultset
    Dim dtCRACK As rdoResultset
    Dim SQL$
    Dim i As Long
    Dim j As Long
    Dim k As Long
    Dim lcase As Long
    Dim CrackRec As Long
    Dim AnalRec As Long
    Dim nCr1 As Long
    Dim nCz1 As Long
    Dim nCL As Long
    Dim CrackL As Single
    Dim Cz10, AzJ0, AzJ1 As Single
    Dim Cr10, ArJ0, ArJ1 As Single
    Dim AlJ0, AlJ1 As Single
    Dim RD, RM, ZD, ZM As Single
    Dim hANAL As Integer
    Dim hCRACK As Integer

    On Error GoTo Err1

    Screen.MousePointer = 11

    For k = 0 To modSTAR.NoOffline - 1

        SQL$ = "SELECT 起点r1, 起点z1, L FROM 解析 WHERE " -
            & "(" & "解析id = " & modSTAR.nmAID & ")" & " AND " -
            & "(" & "ライン番号 = " & k + 1 & ")" -
            & " ORDER BY L;"
        ' Set dtANAL = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
        Set dtANAL = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hANAL)
        If hANAL = -1 Then GoTo Err1
        AnalRec = 0
        Do Until dtANAL.EOF
            For i = 0 To dtANAL.rdoColumns.Count - 1
                Anal(AnalRec, i) = dtANAL.rdoColumns(i).Value
                ' Debug.Print AnalRec; i; Anal(AnalRec, i)
            Next i
            AnalRec = AnalRec + 1
            dtANAL.MoveNext
        Loop
        xCloseResultset dtANAL, hANAL

        ' SQL$ = "SELECT 起点r1, 起点z1, L FROM 开裂 WHERE ("
        SQL$ = "SELECT * FROM 开裂 WHERE ("
            & "(" & "参照id = " & modSTAR.nmID & ")" & " AND " -
            & "(" & "ライン番号 = " & k + 1 & ")" -
            & " ORDER BY ライン番号, L;"
            ' Updateでエラーになる
        ' Set dtCRACK = dbSTAR5.OpenResultset(SQL$, rdOpenDynamic, rdConcurLock)
        Set dtCRACK = xOpenResultset(SQL$, rdOpenDynamic, rdConcurLock, hCRACK)
        If hCRACK = -1 Then GoTo Err1
        nCr1 = FindFieldPosition(dtCRACK, "起点r1")
        nCz1 = FindFieldPosition(dtCRACK, "起点z1")
        nCL = FindFieldPosition(dtCRACK, "L")
        CrackRec = 0
        Do Until dtCRACK.EOF
            For i = 0 To dtCRACK.rdoColumns.Count - 1
                Crack(CrackRec, i) = dtCRACK.rdoColumns(i).Value
                ' Debug.Print CrackRec; i; Crack(CrackRec, i)
            Next i
            CrackRec = CrackRec + 1
            dtCRACK.MoveNext
        Loop
        xCloseResultset dtCRACK, hCRACK

        ' If dtANAL.EOF = False And dtCRACK.EOF = False Then
        ' If AnalRec > 0 And CrackRec > 0 Then
            ' dtCRACK.MoveLast
            ' CrackRec = dtCRACK.RowCount
            ' dtANAL.MoveLast
```

```

'AnalRec = dtANAL.RowCount
Debug.Print CrackRec, AnalRec

'dtANAL.MoveFirst
'dtCRACK.MoveFirst

ArJO = Anal(0, 0) '起点r1
AzJO = Anal(0, 1) '起点z1
AlJO = Anal(0, 2) 'L

For i = 0 To CrackRec - 1

  Icase = 0
  'dtCRACK.Edit
  'CrIO = dtCRACK![起点r1]
  'CzIO = dtCRACK![起点z1]
  CrIO = Crack(i, nCr1) '起点r1
  CzIO = Crack(i, nCz1) '起点z1
  Debug.Print CrIO, CzIO
  'dtANAL.MoveFirst
  'ArJO = dtANAL![起点r1]
  'AzJO = dtANAL![起点z1]
  'AlJO = dtANAL![L]
  'dtANAL.MoveNext

  For j = 1 To AnalRec - 1

    'ArJ1 = dtANAL![起点r1]
    'AzJ1 = dtANAL![起点z1]
    'AlJ1 = dtANAL![L]
    ArJ1 = Anal(j, 0) '起点r1
    AzJ1 = Anal(j, 1) '起点z1
    AlJ1 = Anal(j, 2) 'L

    If (CzIO - AzJO) * (AzJ1 - CzIO) >= 0 Then
      If (CrIO - ArJO) * (ArJ1 - CrIO) >= 0 Then
        'dtCRACK![L] = AlJO + (CzIO - AzJO) * (AlJ1 - AlJO) / (AzJ1 - AzJO)
        Crack(i, nCL) = AlJO + (CzIO - AzJO) * (AlJ1 - AlJO) / (AzJ1 - AzJO)
        Icase = 1
        'Debug.Print "CrIO,CzIO "; CrIO, CzIO
        'Debug.Print "ArJO,AzJO "; ArJO, AzJO
        'Debug.Print "ArJ1,AzJ1 "; ArJ1, AzJ1
        'Debug.Print i; j; Icase; dtCRACK![L]
        Debug.Print i; j; Icase; Crack(i, nCL)
        Exit For
      End If

      RD = Abs(ArJ1 - ArJO) / 2
      RM = (ArJ1 + ArJO) / 2
      If Abs(CrIO - RM) <= 5 + RD Then
        'dtCRACK![L] = AlJO + (CzIO - AzJO) * (AlJ1 - AlJO) / (AzJ1 - AzJO)
        Crack(i, nCL) = AlJO + (CzIO - AzJO) * (AlJ1 - AlJO) / (AzJ1 - AzJO)
        Icase = 2
        'Debug.Print i; j; Icase; dtCRACK![L]
        Debug.Print i; j; Icase; Crack(i, nCL)
        Exit For
      End If
    End If

    ArJO = ArJ1
    AzJO = AzJ1
    AlJO = AlJ1
  'dtANAL.MoveNext
Next j

If Icase = 0 Then

'dtANAL.MoveFirst
'ArJO = dtANAL![起点r1]
'AzJO = dtANAL![起点z1]
'AlJO = dtANAL![L]
'dtANAL.MoveNext
ArJO = Anal(0, 0) '起点r1
AzJO = Anal(0, 1) '起点z1
AlJO = Anal(0, 2) 'L

For j = 1 To AnalRec - 1

  'ArJ1 = dtANAL![起点r1]
  'AzJ1 = dtANAL![起点z1]
  'AlJ1 = dtANAL![L]
  ArJ1 = Anal(j, 0) '起点r1
  AzJ1 = Anal(j, 1) '起点z1
  AlJ1 = Anal(j, 2) 'L

  If (CrIO - ArJO) * (ArJ1 - CrIO) >= 0 Then
    If (CzIO - AzJO) * (AzJ1 - CzIO) >= 0 Then
      'dtCRACK![L] = AlJO + (CrIO - ArJO) * (AlJ1 - AlJO) / (ArJ1 - ArJO)
      Crack(i, nCL) = AlJO + (CrIO - ArJO) * (AlJ1 - AlJO) / (ArJ1 - ArJO)
      Icase = 3
      'Debug.Print i; j; Icase; dtCRACK![L]
      Debug.Print i; j; Icase; Crack(i, nCL)
      Exit For
    End If

    ZD = Abs(AzJ1 - AzJO) / 2
    ZM = (AzJ1 + AzJO) / 2

```



```

    If Abs(CzIO - ZM) <= 5 + ZD Then
      'dtCRACK![L] = AIJO + (CrIO - ArJO) * (AIJ1 - AIJO) / (ArJ1 - ArJO)
      Crack(i, nCL) = AIJO + (CrIO - ArJO) * (AIJ1 - AIJO) / (ArJ1 - ArJO)
      Icase = 4
      'Debug.Print i; j; Icase; dtCRACK![L]
      Debug.Print i; j; Icase; Crack(i, nCL)
      Exit For
    End If
  End If

  ArJO = ArJ1
  AzJO = AzJ1
  AIJO = AIJ1
  'dtANAL.MoveNext
Next j

End If

If Icase = 0 Then
  lstError.AddItem "Line-" & k + 1 & ", 起点r1=" & CrIO & ", 起点z1=" & CzIO & " が補間出来ません。"
End If

' dtCRACK.Update
' dtCRACK.Bookmark = dtCRACK.LastModified

' dtCRACK.MoveNext
Next i

'元のデータを削除
' Call DeleteDataR("き裂", modSTAR.nmID)
SQL$ = "DELETE き裂 WHERE (" & modSTAR.nmID & " AND " & "(参照id = " & modSTAR.nmID & ") AND " & "(ライン番号 = " & k + 1 & "));"
dbSTAR5.Execute SQL$

'配列のデータを再登録
SQL$ = "SELECT * FROM き裂 WHERE (" & modSTAR.nmID & " AND " & "(参照id = " & modSTAR.nmID & ") AND " & "(ライン番号 = " & k + 1 & "));"
Set dtCRACK = xOpenResultset(SQL$, rdOpenDynamic, rdConcurLock, hCRACK)
If hCRACK = -1 Then GoTo Err1
For i = 0 To CrackRec - 1
  dtCRACK.AddNew
  For j = 0 To dtCRACK.rdoColumns.Count - 1
    If IsNull(Crack(i, j)) Then
      dtCRACK.rdoColumns(j).Value = Null
    Else
      dtCRACK.rdoColumns(j).Value = Crack(i, j)
    End If
  Next j
  dtCRACK.Update
Next i
xCloseResultset dtCRACK, hCRACK

Else

  ' If dtCRACK.EOF = True Then lstError.AddItem "該当するき裂データがありません。"
  ' If dtANAL.EOF = True Then lstError.AddItem "該当する解析データがありません。"
  If CrackRec = 0 Then lstError.AddItem "該当するき裂データ(ライン" & k + 1 & ")がありません。"
  If AnalRec = 0 Then lstError.AddItem "該当する解析データ(ライン" & k + 1 & ")がありません。"

End If

' dtCRACK.Close
' dtANAL.Close
' xCloseResultset dtCRACK, hCRACK
' xCloseResultset dtANAL, hANAL
Next k

Screen.MousePointer = 0

If lstError.ListCount = 0 Then
  lstError.AddItem "補間エラーはありませんでした。"
End If

Exit Sub

Err1:
Screen.MousePointer = 0
MsgBox "Error in frmPriCrackInt/Load"
Call ShowErrorMessage
On Error Resume Next
' dtANAL.Close
' dtKANRI.Close
xCloseResultset dtANAL, hANAL
xCloseResultset dtCRACK, hCRACK
Unload Me '強制エラー

End Sub

Private Sub lblError_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
  If Button = 2 Then lblError.ShowWhatsThis

```

End Sub

Private Sub lstError_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)

 If Button = 2 Then lstError.ShowWhatsThis

End Sub

```
Attribute VB_Name = "frmPriViewDBG"  
Attribute VB_GlobalNameSpace = False  
Attribute VB_Creatable = False  
Attribute VB_PredeclaredId = True  
Attribute VB_Exposed = False  
Option Explicit
```

```
Private Sub cmdCancel_Click()  
    modSTAR.tmpFormCloseType = vbCancel  
    Unload Me  
End Sub
```

```
Private Sub cmdOK_Click()  
    modSTAR.tmpFormCloseType = vbOK  
    Unload Me  
End Sub
```

```
Private Sub Form_Load()  
    Caption = modSTAR.tmpFormName$  
    modSTAR.tmpFormCloseType = vbCancel  
  
    If modSTAR.SuperVisor = False Then  
        cmdOK.Enabled = False  
        cmdCancel.Caption = "閉じる(&C)"  
    End If  
  
    dtaSourceDB.Connect = modSTAR.tmpDBConnect$  
    dtaSourceDB.DatabaseName = modSTAR.tmpDBDatabaseName$  
    dtaSourceDB.RecordSource = modSTAR.tmpDBRecordSource$  
    If modSTAR.SuperVisor = True Or  
        modSTAR.tmpFormOpenType = vbOKCancel Then  
        dtaSourceDB.RecordsetType = vbRSTypeDynaset  
    Else  
        dtaSourceDB.RecordsetType = vbRSTypeSnapshot  
    End If  
End Sub
```

```
Private Sub Form_Resize()  
    dbgSourceTBL.Width = ScaleWidth  
    dbgSourceTBL.Height = ScaleHeight - cmdOK.Height * 1.5  
  
    cmdOK.Top = ScaleHeight - cmdOK.Height * 1  
    cmdOK.Width = ScaleWidth / 2  
  
    cmdCancel.Top = ScaleHeight - cmdCancel.Height * 1  
    cmdCancel.Left = ScaleWidth / 2  
    cmdCancel.Width = ScaleWidth / 2  
End Sub
```

```
Attribute VB_Name = "frmPriViewDXF"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
```

```
Private Sub cmdCancel_Click()
    modSTAR.tmpFormCloseType = vbCancel
    Unload Me
End Sub
```

```
Private Sub cmdOK_Click()
    modSTAR.tmpFormCloseType = vbOK
    Unload Me
End Sub
```

```
Private Sub Form_Load()
    Dim i As Long
    Dim LMax As Long
    Dim FName$, Buf$

    Screen.MousePointer = 11

    FName$ = modSTAR.tmpDBDatabaseName$

    'DXFファイルの行数を取得
    Open FName$ For Input As #1
    LMax = 0
    Do While Not EOF(1)
        Line Input #1, Buf$
        LMax = LMax + 1
    Loop
    Close #1

    'クリットの行数を設定
    grdDXF.Rows = LMax + 1
    grdDXF.Cols = 2

    '
    Open FName$ For Input As #1
    For i = 1 To LMax
        grdDXF.Row = i

        grdDXF.Col = 0
        grdDXF.Text = i

        Line Input #1, Buf$
        grdDXF.Col = 1
        grdDXF.Text = Buf$
    Next i
    Close #1

    Screen.MousePointer = 0
End Sub
```

```
Private Sub Form_Resize()
    grdDXF.Width = ScaleWidth
    grdDXF.Height = ScaleHeight - cmdOK.Height * 1.5

    grdDXF.ColWidth(1) = grdDXF.Width - grdDXF.ColWidth(0)

    cmdOK.Top = ScaleHeight - cmdOK.Height * 1
    cmdOK.Width = ScaleWidth / 2

    cmdCancel.Top = ScaleHeight - cmdCancel.Height * 1
    cmdCancel.Left = ScaleWidth / 2
    cmdCancel.Width = ScaleWidth / 2
End Sub
```

```

Attribute VB_Name = "frmPriViewGRD"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Option Explicit

Private SRow As Long
Private SCol As Long
Private OrgStr$
Private bChanged As Boolean

Private Sub Form_Load()
    Dim dtTMP As rdoResultset
    Dim Rows As Long
    Dim Cols As Long
    Dim i As Long
    Dim j As Long
    Dim hTMP As Integer

    On Error GoTo Err1

    Screen.MousePointer = 11

    Caption = modSTAR.tmpFormName$

    Debug.Print modSTAR.tmpViewSQL$
    'Set dtTMP = dbSTAR5.OpenResultset(modSTAR.tmpViewSQL$, rdOpenStatic, rdConcurReadOnly)
    Set dtTMP = xOpenResultset(modSTAR.tmpViewSQL$, rdOpenStatic, rdConcurReadOnly, hTMP)
    If hTMP = -1 Then GoTo Err1

    If dtTMP.EOF = False Then

        dtTMP.MoveLast
        Rows = dtTMP.RowCount
        Cols = dtTMP.rdoColumns.Count
        Debug.Print Rows, Cols

        dtTMP.MoveFirst

        gridView.Rows = Rows + 2
        gridView.Cols = Cols + 1

        For j = 0 To dtTMP.rdoColumns.Count - 1
            gridView.ColWidth(j + 1) = 1000 'デフォルトは630twip
            gridView.Col = j + 1
            gridView.Row = 0
            gridView.Text = dtTMP.rdoColumns(j).Name
            gridView.Row = 1
            gridView.Text = dtTMP.rdoColumns(j).Name
        Next j

        i = 1
        Do Until dtTMP.EOF
            gridView.Row = i + 1
            gridView.Col = 0
            gridView.Text = i
            For j = 0 To dtTMP.rdoColumns.Count - 1
                gridView.Col = j + 1
                If IsNull(dtTMP.rdoColumns(j).Value) = False Then
                    gridView.Text = dtTMP.rdoColumns(j).Value
                End If
            Next j
            i = i + 1
            dtTMP.MoveNext
        Loop
    End If

    'dtTMP.Close
    xCloseResultset dtTMP, hTMP

    mnuFileSave.Enabled = modSTAR.SuperVisor '保存メニューの設定
    bChanged = False '保存問合わせ

    Screen.MousePointer = 0
    Exit Sub 'Err
    'Err
    'Err
Err1:
    Screen.MousePointer = 0
    MsgBox "Error in frmPriViewGRD/Load" 'Err
    Call ShowErrorMessage 'Err
    On Error Resume Next 'Err
    'dtTMP.Close 'Err
    xCloseResultset dtTMP, hTMP 'Err
    modSTAR.tmpFormCloseType = vbCancel 'Err
    Unload Me '強制エラー
End Sub

Private Sub Form_QueryUnload(Cancel As Integer, UnloadMode As Integer)
    Dim Ret As Long

    If bChanged = True Then
        Ret = MsgBox("データが変更されています。保存しますか?", vbYesNoCancel, "STAR")
    End If

```

```

    If Ret = vbYes Then Call mnuFileSave_Click 'データを保存
    If Ret = vbCancel Then Cancel = 1
End If

```

```
End Sub
```

```
Private Sub Form_Resize()
```

```
    grdView.Width = ScaleWidth
    grdView.Height = ScaleHeight
```

```
End Sub
```

```
Private Sub grdView_Click()
```

```
    Dim SRow, SCol As Long
    Dim RowPos As Long
    Dim ColPos As Long
    Dim RowHeight As Long
    Dim ColWidth As Long
    Dim OrgStr$
```

```
    If modSTAR.SuperVisor = False Then Exit Sub
    If grdView.Row <= 1 Then '行1と2は変更不可
        txtEDIT.Visible = False
        Exit Sub
    End If
```

```
    OrgStr = grdView.Text
    SRow = grdView.Row
    SCol = grdView.Col
    RowPos = grdView.RowPos(SRow)
    ColPos = grdView.ColPos(SCol)
    RowHeight = grdView.RowHeight(SRow)
    ColWidth = grdView.ColWidth(SCol)
    txtEDIT.Top = RowPos
    txtEDIT.Left = ColPos
    txtEDIT.Height = RowHeight
    txtEDIT.Width = ColWidth
    txtEDIT.Text = OrgStr$
    txtEDIT.Visible = True
    'Debug.Print "SelChange"; grdView.Row; grdView.Col
```

```
End Sub
```

```
Private Sub grdView_KeyDown(KeyCode As Integer, Shift As Integer)
```

```
    If KeyCode = 67 And Shift = 2 Then 'Ctrl+C
        Clipboard.SetText grdView.Clip, vbCFText 'クリップボードへ
    End If
```

```
End Sub
```

```
Private Sub grdView_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
```

```
    If Button = 2 Then grdView.ShowWhatsThis
```

```
End Sub
```

```
Private Sub mnuEditAll_Click()
```

```
    grdView.SelStartRow = 1
    grdView.SelStartCol = 1
    grdView.SelEndRow = grdView.Rows - 1
    grdView.SelEndCol = grdView.Cols - 1
```

```
End Sub
```

```
Private Sub mnuEditCopy_Click()
```

```
    Clipboard.SetText grdView.Clip, vbCFText
```

```
End Sub
```

```
Private Sub mnuFileClose_Click()
```

```
    Dim Ret As Long
```

```
    On Error GoTo Err1
```

```
    If bChanged = True Then
        Ret = MsgBox("データが変更されています。保存しますか?", vbYesNoCancel, "STAR")
        If Ret = vbYes Then Call mnuFileSave_Click 'データを保存
        If Ret = vbCancel Then Exit Sub
    End If
```

```

modSTAR.tmpFormCloseType = vbCancel
Unload Me

Exit Sub

Err1:
MsgBox "Error in frmPriViewGRD/mnuFileClose"
Call ShowErrorMessage
modSTAR.tmpFormCloseType = vbCancel
Unload Me

End Sub

Private Sub mnuFileSave_Click()
Dim dtDATA As rdoResultset
Dim dtANL As rdoResultset
Dim dtJT1 As rdoResultset
Dim dtJT2 As rdoResultset
Dim SQL$
Dim FRows As Long
Dim FCols As Long
Dim i As Long
Dim j As Long
Dim k As Long
Dim ACols As Long
Dim JCols As Long
Dim Ret As Long
Dim hDATA As Integer
Dim hANL As Integer
Dim hJT1 As Integer
Dim hJT2 As Integer

On Error GoTo Err1

Screen.MousePointer = 11

If Left(modSTAR.tmpFormName$, 2) = "解析" Then '解析

'元のデータを削除
Call DeleteAnalRC("解析", "時点", modSTAR.nmAID, modSTAR.CurrentCase$)
Call DeleteAnalA("解析", "時点", modSTAR.nmAID)
Debug.Print "解析,時点テーブルを削除"

'Gridのデータを再登録
SQL$ = "SELECT * FROM 解析 "
      & "WHERE (解析id = " & modSTAR.nmAID & ");"
' Set dtANL = dbSTAR5.OpenResultset(SQL$, rdOpenDynamic, rdConcurLock)
Set dtANL = xOpenResultset(SQL$, rdOpenDynamic, rdConcurLock, hANL)
If hANL = -1 Then GoTo Err1

SQL$ = "SELECT 時点.* "
      & "FROM 解析 INNER JOIN 時点 ON 解析.時点1id = 時点.id " _
      & "WHERE (解析.解析id = " & modSTAR.nmAID & ");"
' Set dtJT1 = dbSTAR5.OpenResultset(SQL$, rdOpenDynamic, rdConcurLock)
Set dtJT1 = xOpenResultset(SQL$, rdOpenDynamic, rdConcurLock, hJT1)
If hJT1 = -1 Then GoTo Err1

SQL$ = "SELECT 時点.* "
      & "FROM 解析 INNER JOIN 時点 ON 解析.時点2id = 時点.id " _
      & "WHERE (解析.解析id = " & modSTAR.nmAID & ");"
' Set dtJT2 = dbSTAR5.OpenResultset(SQL$, rdOpenDynamic, rdConcurLock)
Set dtJT2 = xOpenResultset(SQL$, rdOpenDynamic, rdConcurLock, hJT2)
If hJT2 = -1 Then GoTo Err1

' If dtANL.EOF = False Then

FRows = grdView.Rows - 2
ACols = dtANL.rdoColumns.Count
JCols = dtJT1.rdoColumns.Count
Debug.Print FRows; ACols; JCols

For i = 0 To FRows - 1
grdView.Row = i + 2
k = 1
Debug.Print "保存 "; i + 1

'解析データの更新
dtANL.Edit
dtANL.AddNew
For j = 0 To ACols - 1
grdView.Col = k
If IsNull(grdView.Text) Or grdView.Text = "" Then
dtANL.rdoColumns(j).Value = Null
Else
dtANL.rdoColumns(j).Value = grdView.Text
End If
k = k + 1
Next j
dtANL.Update
dtANL.Bookmark = dtANL.LastModified

'時点1データの更新

```

```

'dtJT1.Edit
dtJT1.AddNew
For j = 0 To JCols - 1
  grdView.Col = k
  If IsNull(grdView.Text) Or grdView.Text = "" Then
    dtJT1.rdcColumns(j).Value = Null
  Else
    dtJT1.rdcColumns(j).Value = grdView.Text
  End If
  k = k + 1
Next j
dtJT1.Update
'dtJT1.Bookmark = dtJT1.LastModified

'時点データの更新
dtJT2.Edit
dtJT2.AddNew
For j = 0 To JCols - 1
  grdView.Col = k
  If IsNull(grdView.Text) Or grdView.Text = "" Then
    dtJT2.rdcColumns(j).Value = Null
  Else
    dtJT2.rdcColumns(j).Value = grdView.Text
  End If
  k = k + 1
Next j
dtJT2.Update
'dtJT2.Bookmark = dtJT2.LastModified

'dtANL.MoveNext
'dtJT1.MoveNext
'dtJT2.MoveNext
Next i

'End If

'dtANL.Close
'dtJT1.Close
'dtJT2.Close
xCloseResultset dtANL, hANL
xCloseResultset dtJT1, hJT1
xCloseResultset dtJT2, hJT2

Else
'解析以外

'元のデータを削除
dbSTAR5.Execute modSTAR.tmpDeleteSQL$
Debug.Print modSTAR.tmpDeleteSQL$

'Gridのデータを再登録
SQL$ = modSTAR.tmpViewSQL$
Set dtDATA = dbSTAR5.OpenResultset(SQL$, rdOpenDynamic, rdConcurLock)
Set hDATA = xOpenResultset(SQL$, rdOpenDynamic, rdConcurLock, hDATA)
If hDATA = -1 Then GoTo Err1

'If dtDATA.EOF = False Then

'テーブルのサイズを取得
FCols = dtDATA.rdcColumns.Count
'dtDATA.MoveLast
'FRows = dtDATA.RowCount
FRows = grdView.Rows - 2
Debug.Print "FCols,FRows = "; FCols; FRows

'データの取得
dtDATA.MoveFirst
For i = 0 To FRows - 1
  grdView.Row = i + 2
  dtDATA.Edit
  Debug.Print "Row = "; i
  dtDATA.AddNew
  For j = 0 To FCols - 1
    Debug.Print " Col = "; j; grdView.Text; dtDATA.rdcColumns(j).Type
    grdView.Col = j + 1
    If IsNull(grdView.Text) Or grdView.Text = "" Then
      dtDATA.rdcColumns(j).Value = Null
    Else
      dtDATA.rdcColumns(j).Value = grdView.Text
    End If
  Next j
  dtDATA.Update
  Debug.Print "Update"
  'dtDATA.Bookmark = dtDATA.LastModified
  'dtDATA.MoveNext
Next i

'End If

'dtDATA.Close
xCloseResultset dtDATA, hDATA

End If

Screen.MousePointer = 0
bChanged = False

Exit Sub
'Err

```



```
Err1:                                     'Err
  Screen.MousePointer = 0                 'Err
  MsgBox "Error in frmPriViewGRD/mnuFileSave" 'Err
  Call ShowErrorMessage                   'Err
  On Error Resume Next
  dtANL.Close
  dtJT1.Close
  dtJT2.Close
  dtDATA.Close
  xCloseResultset dtANL, hANL
  xCloseResultset dtJT1, hJT1
  xCloseResultset dtJT2, hJT2
  xCloseResultset dtDATA, hDATA
```

```
End Sub
```

```
Private Sub mnuHelp_Click()
  CommonDialog1.HelpCommand = cd1HelpContext
  CommonDialog1.HelpFile = App.HelpFile
  CommonDialog1.ShowHelp
End Sub
```

```
Private Sub txtEdit_Change()
  If grdView.Text <> txtEDIT.Text Then
    grdView.Text = txtEDIT.Text
    bChanged = True
  End If
End Sub
```

```
Attribute VB_Name = "frmSecAnalCase"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Option Explicit
```

```
Private Sub cmdCancel_Click()
    modSTAR.tmpFormCloseType = vbCancel
    Unload Me
End Sub
```

```
Private Sub cmdCancel_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then cmdCancel.ShowWhatsThis
End Sub
```

```
Private Sub cmdOK_Click()
    Dim dtKANRI As rdoResultset
    Dim csName$, SQL$
    Dim Ret As Long
    Dim KNR As KanriLine
    Dim InsPos As Long
    Dim InsFlag As Boolean
    Dim IFERR As Integer
    Dim hKANRI As Integer

    On Error GoTo Err1

    If lstCase.ListIndex < 0 Then
        MsgBox "解析ケースが選択されていません。"
        Exit Sub
    End If

    Screen.MousePointer = 11

    'バックアップデータの作成
    Call CopyTable("管理データ-フォルム", "管理データ-フォルムBACKUP", IFERR)

    '選択された解析ケース名の取得
    csName$ = lstCase.List(lstCase.ListIndex)

    '管理データ-フォルムのidフィールドをRe-Numberingしておく
    Call KanriSequence(IFERR)
    If IFERR = 1 Then GoTo Err1

    'このケースがすでに登録されているかをチェック
    SQL$ = "SELECT * FROM 管理データ-フォルム WHERE (" & "
        & "(試験装置 = " & modSTAR.nmSochi$ & ") AND " & " -
        & "(試験体 = " & modSTAR.nmTai$ & ") AND " & " -
        & "(試験部分 = " & modSTAR.nmBubun$ & ") AND " & " -
        & " ORDER BY 解析ケース;"
    Debug.Print SQL$
    Set dtKANRI = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
    Set dtKANRI = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hKANRI)
    If hKANRI = -1 Then GoTo Err1

    KNR.Cid = dtKANRI![き裂id]
    KNR.Sid = dtKANRI![試験条件id]
    KNR.Oid = dtKANRI![温度id]
    KNR.Mid = dtKANRI![形状id]

    dtKANRI.MoveLast
    InsPos = dtKANRI![id] + 1 '一応、新規挿入にしておく
    InsFlag = True
    modSTAR.tmpMaxID = -1
    dtKANRI.MoveFirst
    dtKANRI.MoveNext

    Do Until dtKANRI.EOF
        Debug.Print dtKANRI![解析ケース], csName$
        If StrComp(dtKANRI![解析ケース], csName$, 1) = 0 Then '既存
            InsPos = dtKANRI![id]
            InsFlag = False
            modSTAR.tmpMaxID = dtKANRI![解析id]
            Exit Do
        ElseIf StrComp(dtKANRI![解析ケース], csName$, 1) > 0 Then '新規
            InsPos = dtKANRI![id]
            InsFlag = True
            modSTAR.tmpMaxID = -1
            Exit Do
        End If

        dtKANRI.MoveNext
    Loop

    'dtKANRI.Close
    xCloseResultset dtKANRI, hKANRI
```

```
Screen.MousePointer = 0
If InsFlag = False Then '既存の解析ケースの更新
  Ret = MsgBox(csName$ & " はすでに登録されています。更新しますか?", vbYesNo)
  If Ret = vbNo Then
    Call DeleteTable("管理テーブル\BACKUP", IFERR) 'バックアップを削除
    Exit Sub
  End If
End If
```

```
Attribute VB_Name = "frmSecAnalView"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Option Explicit
```

```
Private Sub Command2_Click()
    MsgBox 2
End Sub
```

```
Private Sub Command3_Click()
    MsgBox 3
End Sub
```

```
Private Sub cmdCancel_Click()
    modSTAR.tmpFormCloseType = vbCancel

    Unload Me
End Sub
```

```
Private Sub cmdOK_Click()
    modSTAR.tmpFormCloseType = vbOK

    Unload Me
End Sub
```

```
Private Sub Form_Load()
    Dim Border As Single
    Dim i As Long

    On Error GoTo Err1

    Screen.MousePointer = 11

    tabAnal.ShowFocusRect = False
    Border = tabAnal.TabHeight / 2

    For i = 1 To 2
        tabAnal.TabVisible(i) = False
    Next i
```

```
If modSTAR.NoOfLine >= 3 Then
    tabAnal.TabCaption(2) = "LINE-3"
    tabAnal.TabVisible(2) = True
    dbgAnal3.Left = Border
    dbgAnal3.Top = tabAnal.TabHeight + Border
```

```
    dtaSourceDB3.Connect = modSTAR.tmpDBConnect$
    'dtaSourceDB3.DatabaseName = modSTAR.tmpDBDatabaseName3$
    dtaSourceDB3.DatabaseName = modSTAR.tmpDBName$(2)
    'dtaSourceDB3.RecordSource = modSTAR.tmpDBRecordSource3$
    dtaSourceDB3.RecordSource = modSTAR.tmpDBRecSrc$(2)
End If
```

```
If modSTAR.NoOfLine >= 2 Then
    tabAnal.TabCaption(1) = "LINE-2"
    tabAnal.TabVisible(1) = True
    dbgAnal2.Left = Border
    dbgAnal2.Top = tabAnal.TabHeight + Border
```

```
    dtaSourceDB2.Connect = modSTAR.tmpDBConnect$
    'dtaSourceDB2.DatabaseName = modSTAR.tmpDBDatabaseName2$
    dtaSourceDB2.DatabaseName = modSTAR.tmpDBName$(1)
    'dtaSourceDB2.RecordSource = modSTAR.tmpDBRecordSource2$
    dtaSourceDB2.RecordSource = modSTAR.tmpDBRecSrc$(1)
End If
```

```
If modSTAR.NoOfLine >= 1 Then
    tabAnal.TabCaption(0) = "LINE-1"
    tabAnal.TabVisible(0) = True
    dbgAnal1.Left = Border
    dbgAnal1.Top = tabAnal.TabHeight + Border
```

```
    dtaSourceDB1.Connect = modSTAR.tmpDBConnect$
    'dtaSourceDB1.DatabaseName = modSTAR.tmpDBDatabaseName1$
    dtaSourceDB1.DatabaseName = modSTAR.tmpDBName$(0)
    'dtaSourceDB1.RecordSource = modSTAR.tmpDBRecordSource1$
    dtaSourceDB1.RecordSource = modSTAR.tmpDBRecSrc$(0)
End If
```

```
modSTAR.tmpFormCloseType = vbCancel
```

```
Screen.MousePointer = 0
Exit Sub
```

```

Err1:
  Screen.MousePointer = 0
  MsgBox "Error in frmSecAnalView/Load"
  Call ShowErrorMessage
  modSTAR.tmpFormCloseType = vbCancel
  Unload Me '強制終了

```

```
End Sub
```

```

Private Sub Form_Resize()
  Dim ViewHeight, Border As Single
  Dim i As Long

  ViewHeight = ScaleHeight - cmdOK.Height * 1.5
  Border = tabAnal.TabHeight / 2

  tabAnal.Width = ScaleWidth
  tabAnal.Height = ViewHeight

  cmdOK.Top = ScaleHeight - cmdOK.Height * 1
  cmdOK.Width = ScaleWidth / 2

  cmdCancel.Top = ScaleHeight - cmdCancel.Height * 1
  cmdCancel.Left = ScaleWidth / 2
  cmdCancel.Width = ScaleWidth / 2

```

```

If modSTAR.NoOfLine >= 3 Then
  dbgAna13.Width = ScaleWidth - Border * 2
  dbgAna13.Height = ViewHeight - tabAnal.TabHeight - Border * 2
End If
If modSTAR.NoOfLine >= 2 Then
  dbgAna12.Width = ScaleWidth - Border * 2
  dbgAna12.Height = ViewHeight - tabAnal.TabHeight - Border * 2
End If
If modSTAR.NoOfLine >= 1 Then
  dbgAna11.Width = ScaleWidth - Border * 2
  dbgAna11.Height = ViewHeight - tabAnal.TabHeight - Border * 2
End If

```

```
End Sub
```

```
Private Sub tabAnal_Click(PreviousTab As Integer)
```

```

  If PreviousTab = 2 Then
    dbgAna13.Enabled = False
    dbgAna13.Visible = False
  End If
  If PreviousTab = 1 Then
    dbgAna12.Enabled = False
    dbgAna12.Visible = False
  End If
  If PreviousTab = 0 Then
    dbgAna11.Enabled = False
    dbgAna11.Visible = False
  End If

```

```

  If tabAnal.Tab = 2 Then
    dbgAna13.Enabled = True
    dbgAna13.Visible = True
  End If
  If tabAnal.Tab = 1 Then
    dbgAna12.Enabled = True
    dbgAna12.Visible = True
  End If
  If tabAnal.Tab = 0 Then
    dbgAna11.Enabled = True
    dbgAna11.Visible = True
  End If

```

```
End Sub
```

```

Attribute VB_Name = "frmSecDmgCtrl"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Option Explicit

Private SQL$

Private TargetLevel As Long

Private SelFlag As Boolean

Private la(2) As Long
Private da(5) As Double

Private dh(2) As Double
Private dr1(27) As Double
Private dr2(27) As Double

Private daResult(23) As Double

Private Sub cmdCancel_Click()
    Unload Me
End Sub

Private Sub cmdCancel_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then cmdCancel.ShowWhatsThis
End Sub

Private Sub cmdOK_Click()
    Dim dtKANRI As rdoResultset
    Dim dtHYOKA As rdoResultset
    Dim dtCOND As rdoResultset
    Dim dtANAL1 As rdoResultset
    Dim dtANAL2 As rdoResultset
    Dim dtDMG As rdoResultset
    Dim SQL$, SQL1$, SQL2$
    Dim nmC$, nmH$
    Dim nmID As Long
    Dim nmAID As Long
    Dim i As Long
    Dim j As Long
    Dim LI As Long
    Dim lRet As Long
    Dim method_id As Long
    Dim KNR As KanriLine
    Dim IFERR As Integer
    Dim hKANRI As Integer
    Dim hHYOKA As Integer
    Dim hCOND As Integer
    Dim hANAL1 As Integer
    Dim hANAL2 As Integer
    Dim hDMG As Integer

    On Error GoTo Err1

    frmSecDmgCtrl.Height = 6555

    'Set dtKANRI = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
    Set dtKANRI = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hKANRI)
    If hKANRI = -1 Then GoTo Err1

    'If dtKANRI.EOF = False Then
    '    Grid1.Visible = False
    '    frmSecDmgCtrl.Height = 9200
    '    dtKANRI.MoveLast
    '    Grid1.Rows = dtKANRI.RowCount + 1
    '    Grid1.Cols = dtKANRI.rdoColumns.Count + 1
    '    i = 0
    '    dtKANRI.MoveFirst
    '    Do Until dtKANRI.EOF
    '        Grid1.Row = i + 1
    '        For j = 0 To dtKANRI.rdoColumns.Count - 1
    '            Grid1.Col = j + 1
    '            Grid1.Text = dtKANRI.rdoColumns(j).Value
    '        Next j
    '        i = i + 1
    '    dtKANRI.MoveNext
    '    Loop
    'End If
    dtKANRI.Close
Exit Sub

Err1:
SQL$ = "SELECT * FROM 評価法7-7# WHERE 計算実施 = 1;"
Set dtHYOKA = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
Set dtHYOKA = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hHYOKA)
If hHYOKA = -1 Then GoTo Err1

```

```

If dtKANRI.EOF = False And dtHYOKA.EOF = False Then
Screen.MousePointer = 11
lblMessage.Caption = "処理中です。しばらくお待ちください。"
lblMessage.Refresh
LI = -1

Do Until dtKANRI.EOF

'試験部分データの取得
nmID = dtKANRI![参照id]
nmAID = dtKANRI![解析id]
nmC$ = dtKANRI![解析ケース]
Debug.Print nmID; nmC$

'試験条件データの取得/設定
SQL$ = "SELECT * FROM 試験条件 WHERE (参照id = " & nmID & ");"
Set dtCOND = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
Set dtCOND = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hCOND)
If hCOND = -1 Then GoTo Err1

If dtCOND.EOF = False Then
la(0) = dtCOND![材料id]
la(1) = dtCOND![サイクル数]
da(0) = dtCOND![COLD温度]
da(1) = dtCOND![HOT温度]
da(2) = dtCOND![COLD保持時間]
da(3) = dtCOND![HOT保持時間]
da(4) = dtCOND![機械荷重]
Else
lstError.AddItem "参照id = " & nmID & " の試験条件データがありません"
lstError.Refresh
If TargetLevel = 3 Then
MsgBox "参照id = " & nmID & " の試験条件データがありません"
End If
xCloseResultset dtCOND, hCOND
GoTo NextKanri
End If

'dtCOND.Close
xCloseResultset dtCOND, hCOND
lRet = m_theDMG.SetTestCondition(la, da)

dtHYOKA.MoveFirst
Do Until dtHYOKA.EOF

'リストボックスの更新
LI = LI + 1
lstTarget.ListIndex = LI
lstTarget.Refresh
Debug.Print LI

'評価法名の取得/設定
nmH$ = dtHYOKA![評価法名]
method_id = dtHYOKA![id]
lRet = m_theDMG.SetMethod(method_id)
Debug.Print " "; nmH$; method_id

'試験条件データの設定
lRet = m_theDMG.SetTestCondition(la, da)

'古い損傷値データを削除しておく
SQL$ = "DELETE " & nmH$ & " WHERE " -
& "(解析id = " & nmaID & ");" -
'dbSTAR5.Execute SQL$
Call DeleteDataRC(nmH$, "解析id", nmaID, nmC$)
Call DeleteDataA(nmH$, nmaID)

'損傷値テーブルをオープン
SQL$ = "SELECT * FROM " & nmH$ & " WHERE " -
& "(解析id = " & nmaID & ");"
Set dtDMG = dbSTAR5.OpenResultset(SQL$, rdOpenDynamic, rdConcurLock)
Set dtDMG = xOpenResultset(SQL$, rdOpenDynamic, rdConcurLock, hDMG)
If hDMG = -1 Then GoTo Err1

'解析データの取得および設定
SQL1$ = "SELECT 解析.*, 時点1.* " -
& "FROM 解析 INNER JOIN 時点 AS 時点1 ON 解析.時点1id = 時点1.id " -
& "WHERE (解析.解析id = " & nmaID & ");" -
& "ORDER BY 解析.ライン番号, 解析.L;"
SQL2$ = "SELECT 解析.TEMPMAX, 時点2.* " -
& "FROM 解析 INNER JOIN 時点 AS 時点2 ON 解析.時点2id = 時点2.id " -
& "WHERE (解析.解析id = " & nmaID & ");" -
& "ORDER BY 解析.ライン番号, 解析.L;"
Set dtANAL = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
Set dtANAL1 = dbSTAR5.OpenResultset(SQL1$, rdOpenStatic, rdConcurReadOnly)
Set dtANAL1 = xOpenResultset(SQL1$, rdOpenStatic, rdConcurReadOnly, hANAL1)
If hANAL1 = -1 Then GoTo Err1
Set dtANAL2 = dbSTAR5.OpenResultset(SQL2$, rdOpenStatic, rdConcurReadOnly)
Set dtANAL2 = xOpenResultset(SQL2$, rdOpenStatic, rdConcurReadOnly, hANAL2)
If hANAL2 = -1 Then GoTo Err1

i = 0
Do Until (dtANAL1.EOF And dtANAL2.EOF)
i = i + 1
Debug.Print i
dh(0) = dtANAL1![TEMPMAX]

```

```

'dh(1) = dtANAL1![Q]
dh(0) = GetDBData(dtANAL1![TEMPMAX])
dh(1) = GetDBData(dtANAL1![Q])

'dr1(0) = dtANAL1![TEMP]
'dr1(1) = dtANAL1![LTEMP]
'dr1(2) = dtANAL1![MTEMP]
'dr1(3) = dtANAL1![σr]
'dr1(4) = dtANAL1![σz]
'dr1(5) = dtANAL1![σθ]
'dr1(6) = dtANAL1![τrz]
'dr1(7) = dtANAL1![Lσr]
'dr1(8) = dtANAL1![Lσz]
'dr1(9) = dtANAL1![Lσθ]
'dr1(10) = dtANAL1![Lτrz]
'dr1(11) = dtANAL1![Mσr]
'dr1(12) = dtANAL1![Mσz]
'dr1(13) = dtANAL1![Mσθ]
'dr1(14) = dtANAL1![Mτrz]
'dr1(15) = dtANAL1![εr]
'dr1(16) = dtANAL1![εz]
'dr1(17) = dtANAL1![εθ]
'dr1(18) = dtANAL1![γrz]
'dr1(19) = dtANAL1![εpr]
'dr1(20) = dtANAL1![εpz]
'dr1(21) = dtANAL1![εpθ]
'dr1(22) = dtANAL1![γprz]
'dr1(23) = dtANAL1![εcr]
'dr1(24) = dtANAL1![εcz]
'dr1(25) = dtANAL1![εcθ]
'dr1(26) = dtANAL1![γcrz]
dr1(0) = GetDBData(dtANAL1![TEMP])
dr1(1) = GetDBData(dtANAL1![LTEMP])
dr1(2) = GetDBData(dtANAL1![MTEMP])
dr1(3) = GetDBData(dtANAL1![σr])
dr1(4) = GetDBData(dtANAL1![σz])
dr1(5) = GetDBData(dtANAL1![σθ])
dr1(6) = GetDBData(dtANAL1![τrz])
dr1(7) = GetDBData(dtANAL1![Lσr])
dr1(8) = GetDBData(dtANAL1![Lσz])
dr1(9) = GetDBData(dtANAL1![Lσθ])
dr1(10) = GetDBData(dtANAL1![Lτrz])
dr1(11) = GetDBData(dtANAL1![Mσr])
dr1(12) = GetDBData(dtANAL1![Mσz])
dr1(13) = GetDBData(dtANAL1![Mσθ])
dr1(14) = GetDBData(dtANAL1![Mτrz])
dr1(15) = GetDBData(dtANAL1![εr])
dr1(16) = GetDBData(dtANAL1![εz])
dr1(17) = GetDBData(dtANAL1![εθ])
dr1(18) = GetDBData(dtANAL1![γrz])
dr1(19) = GetDBData(dtANAL1![εpr])
dr1(20) = GetDBData(dtANAL1![εpz])
dr1(21) = GetDBData(dtANAL1![εpθ])
dr1(22) = GetDBData(dtANAL1![γprz])
dr1(23) = GetDBData(dtANAL1![εcr])
dr1(24) = GetDBData(dtANAL1![εcz])
dr1(25) = GetDBData(dtANAL1![εcθ])
dr1(26) = GetDBData(dtANAL1![γcrz])

'dr2(0) = dtANAL2![TEMP]
'dr2(1) = dtANAL2![LTEMP]
'dr2(2) = dtANAL2![MTEMP]
'dr2(3) = dtANAL2![σr]
'dr2(4) = dtANAL2![σz]
'dr2(5) = dtANAL2![σθ]
'dr2(6) = dtANAL2![τrz]
'dr2(7) = dtANAL2![Lσr]
'dr2(8) = dtANAL2![Lσz]
'dr2(9) = dtANAL2![Lσθ]
'dr2(10) = dtANAL2![Lτrz]
'dr2(11) = dtANAL2![Mσr]
'dr2(12) = dtANAL2![Mσz]
'dr2(13) = dtANAL2![Mσθ]
'dr2(14) = dtANAL2![Mτrz]
'dr2(15) = dtANAL2![εr]
'dr2(16) = dtANAL2![εz]
'dr2(17) = dtANAL2![εθ]
'dr2(18) = dtANAL2![γrz]
'dr2(19) = dtANAL2![εpr]
'dr2(20) = dtANAL2![εpz]
'dr2(21) = dtANAL2![εpθ]
'dr2(22) = dtANAL2![γprz]
'dr2(23) = dtANAL2![εcr]
'dr2(24) = dtANAL2![εcz]
'dr2(25) = dtANAL2![εcθ]
'dr2(26) = dtANAL2![γcrz]
dr2(0) = GetDBData(dtANAL2![TEMP])
dr2(1) = GetDBData(dtANAL2![LTEMP])
dr2(2) = GetDBData(dtANAL2![MTEMP])
dr2(3) = GetDBData(dtANAL2![σr])
dr2(4) = GetDBData(dtANAL2![σz])
dr2(5) = GetDBData(dtANAL2![σθ])
dr2(6) = GetDBData(dtANAL2![τrz])
dr2(7) = GetDBData(dtANAL2![Lσr])
dr2(8) = GetDBData(dtANAL2![Lσz])
dr2(9) = GetDBData(dtANAL2![Lσθ])
dr2(10) = GetDBData(dtANAL2![Lτrz])
dr2(11) = GetDBData(dtANAL2![Mσr])

```



```

dr2(12) = GetDBData(dtANAL2![Mσz])
dr2(13) = GetDBData(dtANAL2![Mσθ])
dr2(14) = GetDBData(dtANAL2![Mτrz])
dr2(15) = GetDBData(dtANAL2![εr])
dr2(16) = GetDBData(dtANAL2![εz])
dr2(17) = GetDBData(dtANAL2![εθ])
dr2(18) = GetDBData(dtANAL2![γrz])
dr2(19) = GetDBData(dtANAL2![εpr])
dr2(20) = GetDBData(dtANAL2![εpz])
dr2(21) = GetDBData(dtANAL2![εpθ])
dr2(22) = GetDBData(dtANAL2![γprz])
dr2(23) = GetDBData(dtANAL2![εcr])
dr2(24) = GetDBData(dtANAL2![εcz])
dr2(25) = GetDBData(dtANAL2![εcθ])
dr2(26) = GetDBData(dtANAL2![γcrz])

' 損傷値計算
!Ret = m_theDMG.SetRangeAnalysis(dh, dri, dr2)
!Ret = m_theDMG.CalcDamage
!Ret = m_theDMG.GetDamageResult(daResult)

' 計算結果の蓄込み
dtDMG.AddNew
dtDMG![id] = dtANAL1![id]
dtDMG![参照id] = nmID
dtDMG![解析id] = nmAID
dtDMG![試験条件id] = nmID
dtDMG![日付] = Date
dtDMG![解析ケース] = dtANAL1![解析ケース]
dtDMG![ライン番号] = dtANAL1![ライン番号]
dtDMG![L] = dtANAL1![L]
dtDMG![起点r] = dtANAL1![起点r1]
dtDMG![起点z] = dtANAL1![起点z1]
dtDMG![起点θ] = dtANAL1![起点θ1]
' If nmH$ = "TSDS" Then Call SetDmgTSDS(dtDMG)
' If nmH$ = "BDS" Then Call SetDmgBDS(dtDMG)
' If nmH$ = "BDSMD" Then Call SetDmgBDSMD(dtDMG)
' If nmH$ = "DDS" Then Call SetDmgDDS(dtDMG)
' If nmH$ = "DDSWELD" Then Call SetDmgDDSWELD(dtDMG)
' If nmH$ = "EQEF" Then Call SetDmgEQEF(dtDMG)
' If nmH$ = "EQEFWELD" Then Call SetDmgEQEFWELD(dtDMG)
' If nmH$ = "EQEF15" Then Call SetDmgEQEF15(dtDMG)
' If nmH$ = "EQEF167" Then Call SetDmgEQEF167(dtDMG)
' If nmH$ = "INELAST" Then Call SetDmgINELAST(dtDMG)
Call SetDmgTable(dtHYOKA, dtDMG, IFERR)
dtDMG.Update

dtANAL1.MoveNext ' 次の解析行へ
dtANAL2.MoveNext ' 次の解析行へ
Loop

'dtANAL1.Close
'dtANAL2.Close
'dtDMG.Close
xCloseResultset dtANAL1, hANAL1
xCloseResultset dtANAL2, hANAL2
xCloseResultset dtDMG, hDMG

' 管理テーブルの更新
KNR.id = dtKANRI![id]
KNR.nmS = dtKANRI![試験装置]
KNR.nmT = dtKANRI![試験体]
KNR.nmB = dtKANRI![試験部分]
KNR.NOL = dtKANRI![評価ライン数]
KNR.RID = dtKANRI![参照id]
KNR.nmC = dtKANRI![解析ケース]
KNR.Hid = method_id
KNR.Cid = dtKANRI![き裂id]
KNR.Sid = dtKANRI![試験条件id]
KNR.Oid = dtKANRI![温度id]
KNR.Mid = dtKANRI![形状id]
KNR.AID = dtKANRI![解析id]
KNR.Tid = dtKANRI![表計算id]
KNR.Did = dtKANRI![損傷値id]
KNR.CSid = dtKANRI![き裂Sid]
KNR.ASid = dtKANRI![解析Sid]
KNR.TSid = dtKANRI![表計算Sid]
KNR.DSid = dtKANRI![損傷値Sid]

' その評価法が既存か新規かをチェック
Call SecDmgHykChk(KNR, IFERR)
If IFERR = 1 Then GoTo Err1

' Call SetKanriXID("損傷値id", nmID, nmC$, method_id)
' Call SetKanriXID("損傷値id", nmAID, nmID, nmC$, method_id)
Call SetKanriXID6("損傷値id", nmAID, nmID, nmC$, method_id)

dtHYOKA.MoveNext ' 次の評価法へ
Loop

NextKanri:
dtKANRI.MoveNext ' 次の解析ケース/試験部分へ
Loop

End If

'dtKANRI.Close
'dtHYOKA.Close

```

```
xCloseResultset dtKANRI, hKANRI
xCloseResultset dtHYOKA, hHYOKA
```

```
Screen.MousePointer = 0
```

```
Unload Me
```

```
Exit Sub
```

```
Err1:
```

```
Screen.MousePointer = 0
MsgBox "Error in frmSecDmgCtrl/cmdOK"
Call ShowErrorMessage
On Error Resume Next
'dtKANRI.Close
'dtHYOKA.Close
'dtCOND.Close
'dtANAL1.Close
'dtANAL2.Close
'dtDMG.Close
xCloseResultset dtKANRI, hKANRI
xCloseResultset dtHYOKA, hHYOKA
xCloseResultset dtCOND, hCOND
xCloseResultset dtANAL1, hANAL1
xCloseResultset dtANAL2, hANAL2
xCloseResultset dtDMG, hDMG
```

```
End Sub
```

```
Private Sub cmdOK_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
```

```
    If Button = 2 Then cmdOK.ShowWhatsThis
```

```
End Sub
```

```
Private Sub Form_Load()
```

```
Dim dtKANRI As rdoResultset
Dim dtHYOKA As rdoResultset
Dim TargetLevel As Long
Dim SQL$, SQL1$, SQL2$, Buf$, Buf$
Dim hKANRI As Integer
Dim hHYOKA As Integer
```

```
On Error GoTo Err1
```

```
frmSecDmgCtrl.Height = 5300
```

```
Screen.MousePointer = 11
```

```
SelfFlag = False
```

```
TargetLevel = modSTAR.tmpSecDamageSel '対象レベル
```

```
SQL1$ = "SELECT * "
      & "FROM 管理データベース WHERE ("
If TargetLevel = 1 Then '1:全データベース
    SQL2$ = "(解析ケース LIKE 'ケース%') AND "
ElseIf TargetLevel = 2 Then '2:現試験体
    SQL2$ = "(試験装置 = " & modSTAR.nmSochi$ & "'') AND " -
      & "(試験体 = " & modSTAR.nmTai$ & "'') AND " -
      & "(解析ケース LIKE 'ケース%') AND "
ElseIf TargetLevel = 3 Then '3:現解析ケース
    SQL2$ = "(試験装置 = " & modSTAR.nmSochi$ & "'') AND " -
      & "(試験体 = " & modSTAR.nmTai$ & "'') AND " -
      & "(試験部分 = " & modSTAR.nmBibun$ & "'') AND " -
      & "(解析ケース = " & modSTAR.CurrentCase$ & "'') AND "
```

```
End If
SQLX$ = SQL1$ & SQL2$ & "(評価法id = 0)) ORDER BY id;"
```

```
'Set dtKANRI = dbSTAR5.OpenResultset(SQLX$, rdOpenStatic, rdConcurReadOnly)
Set dtKANRI = xOpenResultset(SQLX$, rdOpenStatic, rdConcurReadOnly, hKANRI)
If hKANRI = -1 Then GoTo Err1
```

```
SQL$ = "SELECT 評価法名 FROM 評価法データベース WHERE 計算実施 = 1;"
'Set dtHYOKA = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
Set dtHYOKA = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hHYOKA)
If hHYOKA = -1 Then GoTo Err1
```

```
'リストボックスへ表示
```

```
If dtKANRI.EOF = False And dtHYOKA.EOF = False Then
```

```
Do Until dtKANRI.EOF
    Buf$ = dtKANRI![試験装置] & "/" & dtKANRI![試験体] & "/"
    Buf$ = Buf$ & dtKANRI![試験部分] & "/" & dtKANRI![解析ケース]
    dtHYOKA.MoveFirst
    Do Until dtHYOKA.EOF
        1stTarget.AddItem Buf$ & "/" & dtHYOKA![評価法名]
        dtHYOKA.MoveNext
    Loop
    dtKANRI.MoveNext
Loop
cmdOK.Enabled = True
```

```

Else
    lstTarget.AddItem "処理するデータがありません。"
    cmdOK.Enabled = False
End If

'dtHYOKA.Close
'dtKANRI.Close
xCloseResultset dtHYOKA, hHYOKA
xCloseResultset dtKANRI, hKANRI

Screen.MousePointer = 0

Exit Sub

Err1:
Screen.MousePointer = 0
MsgBox "Error in frmSecDmgCtrl"
Call ShowErrorMessage
On Error Resume Next
'dtHYOKA.Close
'dtKANRI.Close
xCloseResultset dtHYOKA, hHYOKA
xCloseResultset dtKANRI, hKANRI
Unload Me '強制エラー

End Sub

Private Sub lblTarget_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblTarget.ShowWhatsThis
End Sub

Private Sub lstTarget_Click()
    If SelFlag = False Then lstTarget.ListIndex = -1
End Sub

Private Sub lstTarget_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lstTarget.ShowWhatsThis
End Sub

Public Sub SetDmgTTSDS(dtDMG As rdoResultset)
    '計算結果の書き込み
    dtDMG.rdoColumns("V M S").Value = daResult(1) '相当応力範囲 VMS
    dtDMG.rdoColumns("E E P C E Q").Value = daResult(2) '歪範囲 EEPCEQ
    dtDMG.rdoColumns("K e").Value = daResult(5) '歪集中係数 Ke
    dtDMG.rdoColumns("E t").Value = daResult(10) '全歪範囲 Et
    dtDMG.rdoColumns("D f").Value = daResult(19) '疲労損傷値 Df
    dtDMG.rdoColumns("D c").Value = daResult(20) 'ｸﾘｰﾌ損傷値 Dc
    dtDMG.rdoColumns("D").Value = daResult(21) 'ｸﾘｰﾌ疲労損傷値 D

End Sub

Public Sub SetDmgBDS(dtDMG As rdoResultset)
    '計算結果の書き込み
    dtDMG.rdoColumns("S p").Value = daResult(0) '表面応力 Sp
    dtDMG.rdoColumns("S n").Value = daResult(1) '等価線形応力 Sn
    dtDMG.rdoColumns("K").Value = daResult(3) '応力集中係数 K
    dtDMG.rdoColumns("Q").Value = daResult(11) '弾性追従係数 Q
    dtDMG.rdoColumns("K e d").Value = daResult(4) 'Keの係数 Ked
    dtDMG.rdoColumns("Neuber").Value = daResult(18) 'NEUBER則 Neuber
    dtDMG.rdoColumns("K e").Value = daResult(5) '歪集中係数 Ke
    dtDMG.rdoColumns("E f").Value = daResult(9) '熱びくろ歪 Ef
    dtDMG.rdoColumns("E t").Value = daResult(10) '全歪範囲 Et
    dtDMG.rdoColumns("D f").Value = daResult(19) '疲労損傷値 Df
    dtDMG.rdoColumns("D c").Value = daResult(20) 'ｸﾘｰﾌ損傷値 Dc
    dtDMG.rdoColumns("D").Value = daResult(21) 'ｸﾘｰﾌ疲労損傷値 D
    dtDMG.rdoColumns("D c r").Value = daResult(22) '許容値 Dcr

End Sub

Public Sub SetDmgBDSMD(dtDMG As rdoResultset)
    '計算結果の書き込み
    dtDMG.rdoColumns("S p").Value = daResult(0) '表面応力 Sp
    dtDMG.rdoColumns("S n").Value = daResult(1) '等価線形応力 Sn
    dtDMG.rdoColumns("K").Value = daResult(3) '応力集中係数 K
    dtDMG.rdoColumns("Q").Value = daResult(11) '弾性追従係数 Q

```

```

dtDMG.rdoColumns("K e d").Value = daResult(4) 'Keの係数 Ked
dtDMG.rdoColumns("Neuber").Value = daResult(18) 'NEUBER則 Neuber
dtDMG.rdoColumns("K e").Value = daResult(5) '歪集中係数 Ke
dtDMG.rdoColumns("E f").Value = daResult(9) '熱比-歪 Ef
dtDMG.rdoColumns("E t").Value = daResult(10) '全歪範囲 Et
dtDMG.rdoColumns("S i").Value = daResult(2) '初期応力 Si
dtDMG.rdoColumns("Q f").Value = daResult(16) '弾性追従係数 Qf
dtDMG.rdoColumns("D f").Value = daResult(19) '疲労損傷値 Df
dtDMG.rdoColumns("D c").Value = daResult(20) 'ｸﾘｰﾌ損傷値 Dc
dtDMG.rdoColumns("D").Value = daResult(21) 'ｸﾘｰﾌ疲労損傷値 D
dtDMG.rdoColumns("D c r").Value = daResult(22) '許容値 Dcr

```

End Sub

Public Sub SetDmgDDS(dtDMG As rdoResultset)

```

'計算結果の書き込み
dtDMG.rdoColumns("S p").Value = daResult(0) '表面応力 Sp
dtDMG.rdoColumns("S n").Value = daResult(1) '等価線形応力 Sn
dtDMG.rdoColumns("K").Value = daResult(3) '応力集中係数 K
dtDMG.rdoColumns("Q").Value = daResult(11) '弾性追従係数 Q
dtDMG.rdoColumns("K e d").Value = daResult(4) 'Keの係数 Ked
dtDMG.rdoColumns("Neuber").Value = daResult(18) 'NEUBER則 Neuber
dtDMG.rdoColumns("K e").Value = daResult(5) '歪集中係数 Ke
dtDMG.rdoColumns("E f").Value = daResult(9) '熱比-歪 Ef
dtDMG.rdoColumns("E t").Value = daResult(10) '全歪範囲 Et
dtDMG.rdoColumns("S i").Value = daResult(2) '初期応力 Si
dtDMG.rdoColumns("Q f").Value = daResult(16) '弾性追従係数 Qf
dtDMG.rdoColumns("D f").Value = daResult(19) '疲労損傷値 Df
dtDMG.rdoColumns("D c").Value = daResult(20) 'ｸﾘｰﾌ損傷値 Dc
dtDMG.rdoColumns("D").Value = daResult(21) 'ｸﾘｰﾌ疲労損傷値 D

```

End Sub

Public Sub SetDmgDDSWELD(dtDMG As rdoResultset)

```

'計算結果の書き込み
dtDMG.rdoColumns("S p").Value = daResult(0) '表面応力 Sp
dtDMG.rdoColumns("S n").Value = daResult(1) '等価線形応力 Sn
dtDMG.rdoColumns("K").Value = daResult(3) '応力集中係数 K
dtDMG.rdoColumns("K e d").Value = daResult(4) 'Keの係数 Ked
dtDMG.rdoColumns("Neuber").Value = daResult(18) 'NEUBER則 Neuber
dtDMG.rdoColumns("K e").Value = daResult(5) '歪集中係数 Ke
dtDMG.rdoColumns("E f").Value = daResult(9) '熱比-歪 Ef
dtDMG.rdoColumns("E t").Value = daResult(10) '全歪範囲 Et
dtDMG.rdoColumns("S i").Value = daResult(2) '初期応力 Si
dtDMG.rdoColumns("Q b m").Value = daResult(17) '母材に対する弾性追従係数 Qbm
dtDMG.rdoColumns("Q w").Value = daResult(12) '溶接金属に対する弾性追従係数 Qw
dtDMG.rdoColumns("Q e f f").Value = daResult(15) '弾性追従係数 Qeff
dtDMG.rdoColumns("D f").Value = daResult(19) '疲労損傷値 Df
dtDMG.rdoColumns("D c").Value = daResult(20) 'ｸﾘｰﾌ損傷値 Dc
dtDMG.rdoColumns("D").Value = daResult(21) 'ｸﾘｰﾌ疲労損傷値 D

```

End Sub

Public Sub SetDmgEQEF(dtDMG As rdoResultset)

```

'計算結果の書き込み
dtDMG.rdoColumns("S p").Value = daResult(0) '表面応力 Sp
dtDMG.rdoColumns("S n").Value = daResult(1) '等価線形応力 Sn
dtDMG.rdoColumns("K").Value = daResult(3) '応力集中係数 K
dtDMG.rdoColumns("K e d").Value = daResult(4) 'Keの係数 Ked
dtDMG.rdoColumns("K e").Value = daResult(5) '歪集中係数 Ke
dtDMG.rdoColumns("K e d g").Value = daResult(6) '相対的歪集中係数 Kedg
dtDMG.rdoColumns("K e d l").Value = daResult(7) '局所的歪集中係数 Kedl
dtDMG.rdoColumns("E f").Value = daResult(9) '熱比-歪 Ef
dtDMG.rdoColumns("E t").Value = daResult(10) '全歪範囲 Et
dtDMG.rdoColumns("S i").Value = daResult(2) '初期応力 Si
dtDMG.rdoColumns("Q n").Value = daResult(11) '相対的弾性追従係数 Qn
dtDMG.rdoColumns("Q l").Value = daResult(13) '局所的弾性追従係数 Ql
dtDMG.rdoColumns("Q n u").Value = daResult(14) '多軸効果を表す弾性追従係数 Qnu
dtDMG.rdoColumns("Q e f f").Value = daResult(15) '弾性追従係数 Qeff
dtDMG.rdoColumns("D f").Value = daResult(19) '疲労損傷値 Df
dtDMG.rdoColumns("D c").Value = daResult(20) 'ｸﾘｰﾌ損傷値 Dc
dtDMG.rdoColumns("D").Value = daResult(21) 'ｸﾘｰﾌ疲労損傷値 D
dtDMG.rdoColumns("D c r").Value = daResult(22) '許容値 Dcr

```

End Sub

Public Sub SetDmgEQEFWELD(dtDMG As rdoResultset)

```

'計算結果の書き込み
dtDMG.rdoColumns("S p").Value = daResult(0) '表面応力 Sp
dtDMG.rdoColumns("S n").Value = daResult(1) '等価線形応力 Sn
dtDMG.rdoColumns("K").Value = daResult(3) '応力集中係数 K
dtDMG.rdoColumns("K e").Value = daResult(5) '歪集中係数 Ke
dtDMG.rdoColumns("K e d g").Value = daResult(6) '相対的歪集中係数 Kedg
dtDMG.rdoColumns("K e d l").Value = daResult(7) '局所的歪集中係数 Kedl
dtDMG.rdoColumns("E f").Value = daResult(9) '熱比-歪 Ef

```

```

dtDMG.rdoColumns("E t").Value = daResult(10) ' 全歪範囲 Et
dtDMG.rdoColumns("S i").Value = daResult(2) ' 初期応力 Si
dtDMG.rdoColumns("Q n").Value = daResult(11) ' 相対的弾性追従係数 Qn
dtDMG.rdoColumns("Q l").Value = daResult(13) ' 局所的弾性追従係数 Ql
dtDMG.rdoColumns("Q n u").Value = daResult(14) ' 多軸効果を表す弾性追従係数 Qnu
dtDMG.rdoColumns("Q w").Value = daResult(12) ' 溶接金属に対する弾性追従係数 Qw
dtDMG.rdoColumns("Q e f f").Value = daResult(15) ' 弾性追従係数 Qeff
dtDMG.rdoColumns("D f").Value = daResult(19) ' 疲労損傷値 Df
dtDMG.rdoColumns("D c").Value = daResult(20) ' 列-7*損傷値 Dc
dtDMG.rdoColumns("D").Value = daResult(21) ' 列-7*疲労損傷値 D
dtDMG.rdoColumns("D c r").Value = daResult(22) ' 許容値 Dcr

```

End Sub

Public Sub SetDmgEQEF15(dtDMG As rdoResultset)

```

' 計算結果の書込み
dtDMG.rdoColumns("S p").Value = daResult(0) ' 表面応力 Sp
dtDMG.rdoColumns("S n").Value = daResult(1) ' 等価線形応力 Sn
dtDMG.rdoColumns("K").Value = daResult(3) ' 応力集中係数 K
dtDMG.rdoColumns("K e d").Value = daResult(4) ' Keの係数 Ked
dtDMG.rdoColumns("K e").Value = daResult(5) ' 歪集中係数 Ke
dtDMG.rdoColumns("K e d g").Value = daResult(6) ' 相対的歪集中係数 Kedg
dtDMG.rdoColumns("K e d l").Value = daResult(7) ' 局所的歪集中係数 Kedi
dtDMG.rdoColumns("E f").Value = daResult(9) ' 熱-ε歪 Ef
dtDMG.rdoColumns("E t").Value = daResult(10) ' 全歪範囲 Et
dtDMG.rdoColumns("S i").Value = daResult(2) ' 初期応力 Si
dtDMG.rdoColumns("Q n").Value = daResult(11) ' 相対的弾性追従係数 Qn
dtDMG.rdoColumns("Q l").Value = daResult(13) ' 局所的弾性追従係数 Ql
dtDMG.rdoColumns("Q n u").Value = daResult(14) ' 多軸効果を表す弾性追従係数 Qnu
dtDMG.rdoColumns("Q e f f").Value = daResult(15) ' 弾性追従係数 Qeff
dtDMG.rdoColumns("D f").Value = daResult(19) ' 疲労損傷値 Df
dtDMG.rdoColumns("D c").Value = daResult(20) ' 列-7*損傷値 Dc
dtDMG.rdoColumns("D").Value = daResult(21) ' 列-7*疲労損傷値 D
dtDMG.rdoColumns("D c r").Value = daResult(22) ' 許容値 Dcr

```

End Sub

Public Sub SetDmgEQEF167(dtDMG As rdoResultset)

```

' 計算結果の書込み
dtDMG.rdoColumns("S p").Value = daResult(0) ' 表面応力 Sp
dtDMG.rdoColumns("S n").Value = daResult(1) ' 等価線形応力 Sn
dtDMG.rdoColumns("K").Value = daResult(3) ' 応力集中係数 K
dtDMG.rdoColumns("K e d").Value = daResult(4) ' Keの係数 Ked
dtDMG.rdoColumns("K e").Value = daResult(5) ' 歪集中係数 Ke
dtDMG.rdoColumns("K e d g").Value = daResult(6) ' 相対的歪集中係数 Kedg
dtDMG.rdoColumns("K e d l").Value = daResult(7) ' 局所的歪集中係数 Kedi
dtDMG.rdoColumns("E f").Value = daResult(9) ' 熱-ε歪 Ef
dtDMG.rdoColumns("E t").Value = daResult(10) ' 全歪範囲 Et
dtDMG.rdoColumns("S i").Value = daResult(2) ' 初期応力 Si
dtDMG.rdoColumns("Q n").Value = daResult(11) ' 相対的弾性追従係数 Qn
dtDMG.rdoColumns("Q l").Value = daResult(13) ' 局所的弾性追従係数 Ql
dtDMG.rdoColumns("Q n u").Value = daResult(14) ' 多軸効果を表す弾性追従係数 Qnu
dtDMG.rdoColumns("Q e f f").Value = daResult(15) ' 弾性追従係数 Qeff
dtDMG.rdoColumns("D f").Value = daResult(19) ' 疲労損傷値 Df
dtDMG.rdoColumns("D c").Value = daResult(20) ' 列-7*損傷値 Dc
dtDMG.rdoColumns("D").Value = daResult(21) ' 列-7*疲労損傷値 D
dtDMG.rdoColumns("D c r").Value = daResult(22) ' 許容値 Dcr

```

End Sub

Public Sub SetDmgINELAST(dtDMG As rdoResultset)

```

' 計算結果の書込み
dtDMG.rdoColumns("V M S").Value = daResult(1) ' 相当応力範囲 VMS
dtDMG.rdoColumns("E E P C E Q").Value = daResult(2) ' 歪範囲 EEPCEQ
dtDMG.rdoColumns("K e").Value = daResult(5) ' 歪集中係数 Ke
dtDMG.rdoColumns("E t").Value = daResult(10) ' 全歪範囲 Et
dtDMG.rdoColumns("D f").Value = daResult(19) ' 疲労損傷値 Df
dtDMG.rdoColumns("D c").Value = daResult(20) ' 列-7*損傷値 Dc
dtDMG.rdoColumns("D").Value = daResult(21) ' 列-7*疲労損傷値 D

```

End Sub

Public Sub SetDmgTable(dthYOKA As rdoResultset, dtDMG As rdoResultset, IFERR As Integer)

```

Dim i As Integer
Dim ifld As Long
Dim FName$

IFERR = 0

For i = 4 To dthYOKA.rdoColumns.Count - 1
    ifld = dtDMG.rdoColumns(i).Value
    Debug.Print dthYOKA.rdoColumns(i).Name & ", " & ifld
    If ifld >= 0 Then
        FName$ = dthYOKA.rdoColumns(i).Name
        dtDMG.rdoColumns(FName$).Value = daResult(ifld)
    End If
Next i

```

JNC TN9520 99-001

(N_DMG_CT.FRM)

End Sub

```
Attribute VB_Name = "frmSecDmgSel"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Option Explicit
```

```
Private nHyokaT As Long
Private cHyokaT As Long
```

```
Private Sub cmdCancel_Click()
Unload Me
End Sub
```

```
Private Sub cmdCancel_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
If Button = 2 Then cmdCancel.ShowWhatsThis
End Sub
```

```
Private Sub cmdOK_Click()
Dim dtHYOKA As rdoResultset
Dim i As Long
Dim j As Long
Dim k As Long
Dim Item$, SQL$
Dim FF As Boolean
Dim hHYOKA As Integer

On Error GoTo Err1

Screen.MousePointer = 11

'元のデータを削除しておく
SQL$ = "DELETE 評価法テーブル;"
dbSTAR5.Execute SQL$

'配列の評価法テーブルを再登録
SQL$ = "SELECT 評価法名, 計算実施 FROM 評価法テーブル;"
SQL$ = "SELECT * FROM 評価法テーブル;"
Set dtHYOKA = dbSTAR5.OpenResultset(SQL$, rdOpenDynamic, rdConcurLock)
Set dtHYOKA = xOpenResultset(SQL$, rdOpenDynamic, rdConcurLock, hHYOKA)
If hHYOKA = -1 Then GoTo Err1

Do Until dtHYOKA.EOF
Item$ = dtHYOKA![評価法名]
FF = False
For i = 0 To lstSelected.ListCount - 1
If lstSelected.List(i) = Item$ Then
FF = True
Exit For
End If
Next i
dtHYOKA.Edit
dtHYOKA![計算実施] = FF
dtHYOKA.Update
dtHYOKA.Bookmark = dtHYOKA.LastModified
dtHYOKA.MoveNext
Loop

For i = 0 To nHyokaT - 1
'計算実施の更新
Item$ = HyokaT(i, 1) '評価法名
FF = False
For k = 0 To lstSelected.ListCount - 1
If lstSelected.List(k) = Item$ Then
FF = True
Exit For
End If
Next k
HyokaT(i, 2) = FF '計算実施の更新

'テーブルの再登録
dtHYOKA.AddNew
For j = 0 To cHyokaT - 1
dtHYOKA.rdoColumns(j).Value = HyokaT(i, j)
Next j
dtHYOKA.Update

Next i

dtHYOKA.Close
xCloseResultset dtHYOKA, hHYOKA

Screen.MousePointer = 0
Unload Me

Exit Sub
```

```

Err1:
  Screen.MousePointer = 0
  MsgBox "Error in frmSecDmgSel/cmdOK"
  Call ShowErrorMessage
  On Error Resume Next
  'dtHYOKA.Close
  xCloseResultset dtHYOKA, hHYOKA

End Sub

Private Sub cmdOK_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
  If Button = 2 Then cmdOK.ShowWhatsThis
End Sub

Private Sub cmdSel_Click()
  Dim LIndex As Long
  Dim Item$

  LIndex = lstUnselected.ListIndex
  If LIndex >= 0 Then
    Item$ = lstUnselected.List(LIndex)
    lstUnselected.RemoveItem LIndex
    lstSelected.AddItem Item$
  End If
End Sub

Private Sub cmdSel_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
  If Button = 2 Then cmdSel.ShowWhatsThis
End Sub

Private Sub cmdUnsel_Click()
  Dim LIndex As Long
  Dim Item$

  LIndex = lstSelected.ListIndex
  If LIndex >= 0 Then
    Item$ = lstSelected.List(LIndex)
    lstSelected.RemoveItem LIndex
    lstUnselected.AddItem Item$
  End If
End Sub

Private Sub cmdUnsel_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
  If Button = 2 Then cmdUnsel.ShowWhatsThis
End Sub

Private Sub Form_Load()
  Dim dtHYOKA As rdoResultset
  Dim SQL$
  Dim hHYOKA As Integer
  Dim j As Long

  On Error GoTo Err1

  Screen.MousePointer = 11

  'SQL$ = "SELECT 評価法名, 計算実施 FROM 評価法マスタ;"
  SQL$ = "SELECT * FROM 評価法マスタ;"
  'Set dtHYOKA = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
  Set dtHYOKA = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hHYOKA)
  If hHYOKA = -1 Then GoTo Err1

  cHyokaT = dtHYOKA.rdoColumns.Count

  nHyokaT = 0
  Do Until dtHYOKA.EOF

    For j = 0 To cHyokaT - 1
      HyokaT(nHyokaT, j) = dtHYOKA.rdoColumns(j).Value
    Next j

    If dtHYOKA![計算実施] Then
      lstSelected.AddItem dtHYOKA![評価法名]
    Else
      lstUnselected.AddItem dtHYOKA![評価法名]
    End If

    nHyokaT = nHyokaT + 1
    dtHYOKA.MoveNext
  End Sub

```



```
Loop
.'dthYOKA.Close
xCloseResultset dthYOKA, hHYOKA

Screen.MousePointer = 0

Exit Sub

Err1:
Screen.MousePointer = 0
MsgBox "Error in frmSecDmgSel/Load"
Call ShowErrorMessage
On Error Resume Next
.'dthYOKA.Close
xCloseResultset dthYOKA, hHYOKA      '強制エラー
Unload Me

End Sub

Private Sub lblSelected_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblSelected.ShowWhatsThis
End Sub

Private Sub lblUnselected_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblUnselected.ShowWhatsThis
End Sub

Private Sub lstSelected_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lstSelected.ShowWhatsThis
End Sub

Private Sub lstUnselected_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lstUnselected.ShowWhatsThis
End Sub
```

```
Attribute VB_Name = "frmDmgViewSel"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Option Explicit
```

```
Private Sub cmdCancel_Click()
    modSTAR.tmpFormCloseType = vbCancel
    Unload Me
End Sub
```

```
Private Sub cmdCancel_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then cmdCancel.ShowWhatsThis
End Sub
```

```
Private Sub cmdOK_Click()
    On Error GoTo Err1

    If lstHyoka.ListIndex < 0 Then
        MsgBox "評価法が選択されていません。"
        Exit Sub
    End If

    modSTAR.tmpViewDamageName$ = lstHyoka.List(lstHyoka.ListIndex)

    modSTAR.tmpFormCloseType = vbOK
    Unload Me

    Exit Sub
```

```
Err1:
    MsgBox "Error in frmDmgViewSel/cmdOK"
    Call ShowErrorMessage
    modSTAR.tmpFormCloseType = vbCancel
End Sub
```

```
Private Sub cmdOK_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then cmdOK.ShowWhatsThis
End Sub
```

```
Private Sub Form_Load()
    Dim dtTMP As rdoResultset
    Dim SQL$
    Dim hTMP As Integer

    On Error GoTo Err1

    Screen.MousePointer = 11

    SQL$ = "SELECT 評価法id FROM 管理テーブル WHERE (" _
        & "(参照id=" & modSTAR.nmID & ") AND " _
        & "(解析キー=" & modSTAR.CurrentCase$ & ") AND " _
        & "(評価法id > 0));"

    ' Set dtTMP = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
    Set dtTMP = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hTMP)
    If hTMP = -1 Then GoTo Err1

    Do Until dtTMP.EOF
        lstHyoka.AddItem GetHyokaName(dtTMP![評価法id])
        dtTMP.MoveNext
    Loop

    ' dtTMP.Close
    xCloseResultset dtTMP, hTMP

    modSTAR.tmpFormCloseType = vbCancel

    Screen.MousePointer = 0

    Exit Sub
```

```
Err1:
    Screen.MousePointer = 0
    MsgBox "Error in frmDmgViewSel/Load"
    Call ShowErrorMessage
    xCloseResultset dtTMP, hTMP
```

```
modSTAR.tmpFormCloseType = vbCancel  
Unload Me ' 強制エラー  
End Sub
```

```
Private Sub lblHyoka_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)  
    If Button = 2 Then lblHyoka.ShowWhatsThis  
End Sub
```

```
Private Sub lstHyoka_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)  
    If Button = 2 Then lstHyoka.ShowWhatsThis  
End Sub
```

```

Attribute VB_Name = "frmSecFinasView"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Option Explicit

#If Win32 Then
    Private Declare Function ShellExecute Lib _
        "shell32.dll" Alias "ShellExecuteA" _
        (ByVal hwnd As Long,
        ByVal lpOperation As String, _
        ByVal lpFile As String, _
        ByVal lpParameters As String, _
        ByVal lpDirectory As String, _
        ByVal nShowCmd As Long) As Long
#Else
    Private Declare Function ShellExecute Lib _
        "shell.dll" Alias "ShellExecuteA" _
        (ByVal hwnd As Integer,
        ByVal lpOperation As String, _
        ByVal lpFile As String, _
        ByVal lpParameters As String, _
        ByVal lpDirectory As String, _
        ByVal nShowCmd As Integer) As Integer
#End If

Private Const SW_SHOWNORMAL = 1

Private SRow As Long
Private SCol As Long
Private OrgStr$

Private Sub Command1_Click()
    Unload Me
End Sub

Private Sub cmdCancel_GotFocus()
    txtEDIT.Visible = False
End Sub

Private Sub cmdCancel_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then cmdCancel.ShowWhatsThis
End Sub

Private Sub cmdClose_Click()
    Unload Me
End Sub

Private Sub cmdCancel_Click()
    Unload Me
End Sub

Private Sub cmdClose_GotFocus()
    txtEDIT.Visible = False
End Sub

Private Sub cmdClose_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then cmdClose.ShowWhatsThis
End Sub

Private Sub cmdOK_Click()
    Dim dtFINAS As rdoResultset
    Dim SQL$
    Dim FRows As Long
    Dim FCols As Long
    Dim i As Long
    Dim j As Long
    Dim hFINAS As Integer
    Dim IFERR As Integer

    On Error GoTo Err1

    Screen.MousePointer = 11

```

```

'SQL$ = "SELECT 試験装置,試験体,試験部分,解析ケース,評価ライン数,形状DXF," _
& "FEマシ,FINAS入力,Range1,Range2,Range3,FINAS出力,FINAS説明" _
& "FROM 管理データベース WHERE 評価法id = 0;"
'Set dtFINAS = dbSTAR5.OpenResultset(SQL$, rdOpenDynamic, rdConcurLock)
'Set dtFINAS = xOpenResultset(SQL$, rdOpenDynamic, rdConcurLock, hFINAS)
'If hFINAS = -1 Then GoTo Err1
,
'If dtFINAS.EOF = False Then
,
' 'テーブルのサイズ'を取得
' FCols = dtFINAS.rdoColumns.Count
' dtFINAS.MoveLast
' FRows = dtFINAS.RowCount
,
' 'データの取得
' dtFINAS.MoveFirst
' For i = 1 To FRows
'   grdFinas.Row = i
'   dtFINAS.Edit
'   For j = 6 To FCols
'     grdFinas.Col = j
'     dtFINAS.rdoColumns(j - 1).Value = grdFinas.Text
'   Next j
'   dtFINAS.Update
'   dtFINAS.Bookmark = dtFINAS.LastModified
'   dtFINAS.MoveNext
' Next i
'End If
,
'dtFINAS.Close
'xCloseResultset dtFINAS, hFINAS

grdFinas.Row = 0
For i = 0 To nKanriT - 1
  If KanriT(i, 7) = 0 Then '評価法id>0はスキップ
    grdFinas.Row = grdFinas.Row + 1
    Debug.Print grdFinas.Row
    For j = 6 To 13
      grdFinas.Col = j
      KanriT(i, j + 13) = grdFinas.Text
    Next j
  End If
Next i

'配列の書込み
Call KanriWrite(IFERR)
If IFERR = 1 Then GoTo Err1

'管理データベースの再読み込み
Call KanriRead(IFERR)

Screen.MousePointer = 0

Unload Me

Exit Sub

Err1:
Screen.MousePointer = 0
MsgBox "Error in frmSecFinasView/cmdOK"
Call ShowErrorMessage
On Error Resume Next
'dtFINAS.Close
xCloseResultset dtFINAS, hFINAS

End Sub

Private Sub cmdOK_GotFocus()
txtEDIT.Visible = False
End Sub

Private Sub cmdOK_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
If Button = 2 Then cmdOK.ShowWhatsThis
End Sub

Private Sub Form_Load()
Dim dtFINAS As rdoResultset
Dim FRows As Long
Dim FCols As Long
Dim i As Long
Dim j As Long
Dim SQL$
Dim hFINAS As Integer

On Error GoTo Err1

Screen.MousePointer = 11

```

```

If modSTAR.tmpFormOpenType = vbOKCancel Then '解析情報入力
    cmdClose.Enabled = False
    cmdClose.Visible = False
Else '解析情報確認
    cmdOK.Enabled = False
    cmdOK.Visible = False
    cmdCancel.Enabled = False
    cmdCancel.Visible = False
End If

'SQL$ = "SELECT * FROM FINAS連携テ-プ#;"
SQL$ = "SELECT 試験装置,試験体,試験部分,解析ケ-ス,評価ライン数,形状DXF," _
    & "FEマップ,FINAS入力,Range1,Range2,Range3,FINAS出力,FINAS説明" _
    & "FROM 管理テ-プ# WHERE (評価法id = 0)" _
    & " ORDER BY id;"
Set dtFINAS = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
Set dtFINAS = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hFINAS)
If hFINAS = -1 Then GoTo Err1

If dtFINAS.EOF = False Then

    'テ-プ#のサイズを取得
    FCols = dtFINAS.rdoColumns.Count
    dtFINAS.MoveLast
    FRows = dtFINAS.RowCount

    'GRIDの大きさを設定
    grdFinas.Cols = FCols + 1
    grdFinas.Rows = FRows + 1

    'フイ-ト名の設定
    grdFinas.Row = 0
    For j = 1 To FCols
        grdFinas.ColWidth(j) = 1000
        grdFinas.Col = j
        grdFinas.Text = dtFINAS.rdoColumns(j - 1).Name
    Next j

    'テ-タの取得
    dtFINAS.MoveFirst
    For i = 1 To FRows
        grdFinas.Row = i
        For j = 1 To FCols
            grdFinas.Col = j
            If dtFINAS.rdoColumns(j - 1).Value <> "" Then
                grdFinas.Text = dtFINAS.rdoColumns(j - 1).Value
            End If
        Next j
        dtFINAS.MoveNext
    Next i

End If

'dtFINAS.Close
xCloseResultset dtFINAS, hFINAS

Screen.MousePointer = 0

Exit Sub

Err1:
Screen.MousePointer = 0
MsgBox "Error in frmSecFinasView/Load"
Call ShowErrorMessage
On Error Resume Next
'dtFINAS.Close
xCloseResultset dtFINAS, hFINAS
Unload Me '強制エラー

End Sub

Private Sub Form_Resize()
grdFinas.Width = ScaleWidth
grdFinas.Height = ScaleHeight - cmdCancel.Height * 1.5

If modSTAR.tmpFormOpenType = vbOKCancel Then
    cmdOK.Top = ScaleHeight - cmdOK.Height * 1
    cmdOK.Left = 0
    cmdOK.Width = ScaleWidth / 2
    cmdCancel.Top = ScaleHeight - cmdCancel.Height * 1
    cmdCancel.Left = ScaleWidth / 2
    cmdCancel.Width = ScaleWidth / 2
Else
    cmdClose.Top = ScaleHeight - cmdClose.Height * 1
    cmdClose.Left = 0
    cmdClose.Width = ScaleWidth
End If

End Sub

```

```

Private Sub grdFinas_Click()
    'Dim SRow, SCol As Long
    Dim RowPos As Long
    Dim ColPos As Long
    Dim RowHeight As Long
    Dim ColWidth As Long
    'Dim OrgStr$

    If modSTAR.tmpFormOpenType = vbOKOnly Then Exit Sub
    If grdFinas.Col <= 5 Then
        txtEDIT.Visible = False
        Exit Sub
    End If

    OrgStr = grdFinas.Text
    SRow = grdFinas.Row
    SCol = grdFinas.Col
    RowPos = grdFinas.RowPos(SRow)
    ColPos = grdFinas.ColPos(SCol)
    RowHeight = grdFinas.RowHeight(SRow)
    ColWidth = grdFinas.ColWidth(SCol)
    txtEDIT.Top = RowPos
    txtEDIT.Left = ColPos
    txtEDIT.Height = RowHeight
    txtEDIT.Width = ColWidth
    txtEDIT.Text = OrgStr$
    txtEDIT.Visible = True
    'Debug.Print "SelChange"; grdFinas.Row; grdFinas.Col

End Sub

Private Sub grdFinas_DbIClick()
    Dim iret As Long
    Dim FNSName$

    If modSTAR.tmpFormOpenType = vbOKOnly Then
        FNSName$ = grdFinas.Text

        If IsNull(FNSName$) = True Then Exit Sub
        If FNSName$ = "" Then Exit Sub
        If grdFinas.Col <= 5 Then Exit Sub

        If Right(FNSName$, 1) = "*" Then
            Shell "EXPLORER /e,/root,," & FNSName$, vbNormalFocus
        Else
            Shell "EXPLORER /e,/select," & FNSName$, vbNormalFocus
        End If

        'iret = ShellExecute(Me.hwnd, _
        '    vbNullString, _
        '    FNSName$, _
        '    vbNullString, _
        '    vbNullString, _
        '    SW_SHOWNORMAL)

    End If

End Sub

Private Sub grdFinas_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    Dim iret As Long
    Dim FNSName$

    If modSTAR.tmpFormOpenType = vbOKOnly Then
        FNSName$ = grdFinas.Text

        If IsNull(FNSName$) = True Then Exit Sub
        If FNSName$ = "" Then Exit Sub
        If grdFinas.Col <= 5 Then Exit Sub

        If (Shift And 4) <> 4 Then Exit Sub 'Alt+の状態
        If Right(FNSName$, 1) = "*" Then Exit Sub

        iret = ShellExecute(Me.hwnd, _
            vbNullString, _
            FNSName$, _
            vbNullString, _
            vbNullString, _
            SW_SHOWNORMAL)

    End If

End Sub

Private Sub txtEdit_Change()
    grdFinas.Text = txtEDIT.Text
End Sub

Private Sub txtEDIT_LostFocus()
    Debug.Print txtEDIT.Text; grdFinas.Row; grdFinas.Col
End Sub

```



```
Attribute VB_Name = "frmSecTblCtrl"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Option Explicit
```

```
Private SelFlag As Boolean
Private SQL$
```

```
Private Sub cmdCancel_Click()
    Unload Me
End Sub
```

```
Private Sub cmdCancel_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then cmdCancel.ShowWhatsThis
End Sub
```

```
Private Sub cmdOK_Click()
    Dim dtKANRI As rdoResultset
    Dim nmCase$
    Dim i As Long
    Dim nmID As Long
    Dim nmAID As Long
    Dim IFERR As Integer
    Dim hKANRI As Integer

    On Error GoTo Err1

    Screen.MousePointer = 11

    Debug.Print SQL$
    ' Set dtKANRI = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
    Set dtKANRI = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hKANRI)
    If hKANRI = -1 Then GoTo Err1
```

```
lblMessage.Caption = "処理中です。しばらくお待ちください。"
lblMessage.Refresh
i = -i
```

```
Do Until dtKANRI.EOF
    i = i + 1
    lstTarget.ListIndex = i
    lstTarget.Refresh
```

```
nmID = dtKANRI![参照id]
nmAID = dtKANRI![解析id]
nmCase$ = dtKANRI![解析ケース]
Debug.Print nmAID, nmCase$
```

```
' 表計算を実行
' Call SecTableCalc5(nmID, nmCase$, IFERR) 'TCALCSVを使用
Call SecTableCalc5(nmID, nmAID, nmCase$, IFERR) 'TCALCSVを使用
If IFERR = i Then GoTo Err1
```

```
' Call SetKanriXID("表計算id", nmID, nmCase$, 0)
' Call SetKanriXID("表計算id", nmAID, nmID, nmCase$, 0)
Call SetKanriXID6("表計算id", nmAID, nmID, nmCase$, 0)
```

```
dtKANRI.MoveNext
Loop
```

```
Debug.Print ""
```

```
' dtKANRI.Close
xCloseResultset dtKANRI, hKANRI
```

```
Screen.MousePointer = 0
Unload Me
```

```
Exit Sub
```

```
Err1:
    Screen.MousePointer = 0
    MsgBox "Error in frmSecTblCtrl/cmdOK"
    Call ShowErrorMessage
    On Error Resume Next
    ' dtKANRI.Close
    xCloseResultset dtKANRI, hKANRI
```

```
End Sub
```

```
Private Sub cmdOK_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then cmdOK.ShowWhatsThis
```

End Sub

```

Private Sub Form_Load()
    Dim dtKANRI As rdarResultset
    Dim TargetLevel As Long
    'Dim SQL$
    Dim SQL1$, SQL2$, Buf$
    Dim hKANRI As Integer

    On Error GoTo Err1

    Screen.MousePointer = 11

    SelfFlag = False

    TargetLevel = modSTAR.tmpSecTableSel          '対象レベル

    SQL1$ = "SELECT 試験装置,試験体,試験部分,解析ケース,参照id,解析id " _
        & "FROM 管理データベース WHERE ("
    If TargetLevel = 1 Then                        '1:全テーブル
        SQL2$ = "(解析ケース LIKE 'ケース%') AND "
    ElseIf TargetLevel = 2 Then                   '2:現試験体
        SQL2$ = "(試験装置 = " & modSTAR.nmSochi$ & "'') AND " _
            & "(試験体 = " & modSTAR.nmTai$ & "'') AND " _
            & "(解析ケース LIKE 'ケース%') AND "
    ElseIf TargetLevel = 3 Then                   '3:現解析ケース
        SQL2$ = "(試験装置 = " & modSTAR.nmSochi$ & "'') AND " _
            & "(試験体 = " & modSTAR.nmTai$ & "'') AND " _
            & "(試験部分 = " & modSTAR.nmBubun$ & "'') AND " _
            & "(解析ケース = " & modSTAR.CurrentCase$ & "'') AND "
    End If
    SQL$ = SQL1$ & SQL2$ & "(評価法id = 0)) ORDER BY id;"

    'リストボックスへ表示
    'Set dtKANRI = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
    Set dtKANRI = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hKANRI)
    If hKANRI = -1 Then GoTo Err1

    If dtKANRI.EOF = False Then

        Do Until dtKANRI.EOF
            Buf$ = dtKANRI![試験装置] & "/" & dtKANRI![試験体] & "/"
            Buf$ = Buf$ & dtKANRI![試験部分] & "/" & dtKANRI![解析ケース]
            lstTarget.AddItem Buf$
            dtKANRI.MoveNext
        Loop
        dtKANRI.Close
        xCloseResultset dtKANRI, hKANRI
        cmdOK.Enabled = True

    Else

        lstTarget.AddItem "処理するデータがありません。"
        cmdOK.Enabled = False

    End If

    Screen.MousePointer = 0

    Exit Sub

Err1:
    Screen.MousePointer = 0
    MsgBox "Error in frmSecTblCtrl/Load"
    Call ShowErrorMessage
    On Error Resume Next
    dtKANRI.Close
    xCloseResultset dtKANRI, hKANRI
    Unload Me          '強制エラー

End Sub

Private Sub lblTarget_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblTarget.ShowWhatsThis
End Sub

Private Sub lstTarget_Click()
    If SelfFlag = False Then lstTarget.ListIndex = -1
End Sub

Private Sub lstTarget_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lstTarget.ShowWhatsThis

```

End Sub

```
Attribute VB_Name = "frmSelBubun2"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Option Explicit
```

```
Private Sub cmdCancel_Click()
    Unload Me
End Sub
```

```
Private Sub cmdCancel_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then cmdCancel.ShowWhatsThis
End Sub
```

```
Private Sub cmdOK_Click()
    Dim dtTMP As rdoResultset
    Dim SQL$
    Dim i As Long
    Dim LIndex As Long
    Dim SV As Boolean
    Dim hTMP As Integer

    On Error GoTo Err1 'Err

    Screen.MousePointer = 11

    '試験部分名, 試験体名, 試験装置名の取得
    LIndex = outBubun.ListIndex
    modSTAR.nmBubun$ = outBubun.List(LIndex)

    For i = LIndex - 1 To 0 Step -1
        If outBubun.Indent(i) = 2 Then
            modSTAR.nmTai$ = outBubun.List(i)
            Exit For
        End If
    Next i

    For i = LIndex - 1 To 0 Step -1
        If outBubun.Indent(i) = 1 Then
            modSTAR.nmSochi$ = outBubun.List(i)
            Exit For
        End If
    Next i

    '現解析ケース名の取得
    modSTAR.CurrentCase$ = ""
    modSTAR.KanriLine2 = -1

    SQL$ = "SELECT * FROM 管理テーブル WHERE ("
        & "(試験装置 = '" & modSTAR.nmSochi$ & "') AND "
        & "(試験体 = '" & modSTAR.nmTai$ & "') AND "
        & "(試験部分 = '" & modSTAR.nmBubun$ & "');"
    Set dtTMP = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
    Set dtTMP = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hTMP)
    If hTMP = -1 Then GoTo Err1

    'Do Until dtTMP.EOF
    ' Debug.Print dtTMP![試験装置], dtTMP![試験体], dtTMP![試験部分]
    ' dtTMP.MoveNext
    'Loop
    modSTAR.nmID = dtTMP![参照id]
    modSTAR.NoOfLine = dtTMP![評価ライン数]
    modSTAR.nmAID = 0 '解析idは地獄
    Debug.Print dtTMP![参照id], dtTMP![評価ライン数]

    'dtTMP.Close
    xCloseResultset dtTMP, hTMP

    SV = modSTAR.SuperVisor

    '各種コメントを実行可に設定
    modSTAR.frmMain.mnuSelCase.Enabled = True
    modSTAR.frmMain.mnuSelShape.Enabled = False

    modSTAR.frmMain.mnuPriCrackReg.Enabled = SV
    modSTAR.frmMain.mnuPriCrackShow.Enabled = True
    modSTAR.frmMain.mnuPriCrackInt.Enabled = False
    modSTAR.frmMain.mnuPriCondReg.Enabled = SV
    modSTAR.frmMain.mnuPriCondShow.Enabled = True
    modSTAR.frmMain.mnuPriTempReg.Enabled = SV
    modSTAR.frmMain.mnuPriTempShow.Enabled = True
    modSTAR.frmMain.mnuPriShapeReg.Enabled = SV
    modSTAR.frmMain.mnuPriShapeShow.Enabled = True

    modSTAR.frmMain.mnuSecAnalReg.Enabled = SV
```

```

modSTAR.frmMain.mnuSecAnalShow.Enabled = False
modSTAR.frmMain.mnuSecTableTai.Enabled = SV
modSTAR.frmMain.mnuSecTableCase.Enabled = False
modSTAR.frmMain.mnuSecTableShow.Enabled = False
modSTAR.frmMain.mnuSecDamageTai.Enabled = SV
modSTAR.frmMain.mnuSecDamageCase.Enabled = False
modSTAR.frmMain.mnuSecDamageShow.Enabled = False
modSTAR.frmMain.mnuSecQCalc.Enabled = SV

modSTAR.frmMain.mnuBunExec.Enabled = False

modSTAR.frmMain.mnuTokCrack.Enabled = SV
modSTAR.frmMain.mnuTokInterTai.Enabled = SV
modSTAR.frmMain.mnuTokInterCase.Enabled = False
modSTAR.frmMain.mnuTokDataView.Enabled = False

'frmStarMain.Caption = "PNC STAR (" & modSTAR.nmBubun$ & ")"

'データの初期化
modSTAR.CurBunLineNo = -1
modSTAR.CurBunQueryNo = -1

'Normalフォームの更新
modSTAR.frmMain.Caption = modSTAR.nmBubun$
modSTAR.frmMain.lblSochi.Caption = modSTAR.nmSochi$
modSTAR.frmMain.lblTai.Caption = modSTAR.nmTai$
modSTAR.frmMain.lblBubun.Caption = modSTAR.nmBubun$
modSTAR.frmMain.lblCase.Caption = ""

Screen.MousePointer = 0
Unload Me

Exit Sub                                'Err

Err1:                                    'Err
Screen.MousePointer = 0                  'Err
MsgBox "Error in frmSelBubun2/cmdOK"     'Err
Call ShowErrorMessage                   'Err
On Error Resume Next                    'Err
'dtTMP.Close                             'Err
xCloseResultset dtTMP, hTMP             'Err

End Sub

Private Sub cmdOK_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then cmdOK.ShowWhatsThis
End Sub

Private Sub Form_Load()
    Dim dtKANRI As rdoResultset
    Dim nameSochi$, nameTai$, nameBubun$, SQL$
    Dim i As Long
    Dim Item As Long
    Dim hKANRI As Integer

    On Error GoTo Err1                    'Err

    Screen.MousePointer = 11

    SQL$ = "SELECT * FROM 管理データテーブル ORDER BY id;"
    'Set dtKANRI = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
    Set dtKANRI = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hKANRI)
    If hKANRI = -1 Then GoTo Err1

    nameSochi$ = ""
    nameTai$ = ""
    nameBubun$ = ""
    i = 0
    Item = 0

    Do Until dtKANRI.EOF

        If dtKANRI![試験装置] <> nameSochi$ Then
            nameSochi$ = dtKANRI![試験装置]
            nameTai$ = ""
            nameBubun$ = ""
            outBubun.AddItem nameSochi$
            outBubun.Indent(i) = 1
            i = i + 1
        End If

        If dtKANRI![試験体] <> nameTai$ Then
            nameTai$ = dtKANRI![試験体]
            nameBubun$ = ""
            outBubun.AddItem nameTai$
            outBubun.Indent(i) = 2
            i = i + 1
        End If
    
```

```

If dtKANRI![試験部分] <> nameBubun$ Then
    nameBubun$ = dtKANRI![試験部分]
    outBubun.AddItem nameBubun$
    outBubun.Indent(i) = 3
    outBubun.PictureType(i) = outLeaf
    outBubun.ItemData(i) = Item
    i = i + 1
End If

dtKANRI.MoveNext
Item = Item + 1

Loop

'すべての試験部分名を拡張表示
If outBubun.ListCount > 0 Then
    For i = 0 To outBubun.ListCount - 1
        If outBubun.HasSubItems(i) Then
            outBubun.Expand(i) = True
        End If
    Next i
    outBubun.ListIndex = 0
End If

'dtKANRI.Close
xCloseResultset dtKANRI, hKANRI

cmdOK.Enabled = False
Screen.MousePointer = 0

Exit Sub                                     'Err

Err1:                                         'Err
Screen.MousePointer = 0                       'Err
MsgBox "Error in frmSelBubun2/Load"           'Err
Call ShowErrorMessage                         'Err
On Error Resume Next                         'Err
'dtKANRI.Close                               'Err
xCloseResultset dtKANRI, hKANRI              'Err
Unload Me                                     '強制エラー発生

End Sub

Private Sub lblBubun_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblBubun.ShowWhatsThis
End Sub

Private Sub outBubun_Click()
    If outBubun.Indent(outBubun.ListIndex) = 3 Then
        cmdOK.Enabled = True
    Else
        cmdOK.Enabled = False
    End If
End Sub

Private Sub outBubun_Collapse(ListIndex As Integer)
    outBubun.ListIndex = ListIndex
End Sub

Private Sub outBubun_DblClick()
    Dim LIndex As Long
    LIndex = outBubun.ListIndex
    'MsgBox outBubun.ListIndex

    If outBubun.HasSubItems(LIndex) Then
        If outBubun.Expand(LIndex) = False Then
            outBubun.Expand(LIndex) = True
            outBubun.PictureType(LIndex) = outOpen
        Else
            outBubun.Expand(LIndex) = False
            outBubun.PictureType(LIndex) = outClosed
        End If
    End If

    If outBubun.Indent(LIndex) = 3 Then
        cmdOK.Enabled = True
    Else
        cmdOK.Enabled = False
    End If
End Sub

```

```
Private Sub outBubun_Expand(ListIndex As Integer)
    outBubun.ListIndex = ListIndex
End Sub
```

```
Private Sub outBubun_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then outBubun.ShowWhatsThis
End Sub
```

```

Attribute VB_Name = "frmSelCase2"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Option Explicit

Private Sub cmdCancel_Click()
    Unload Me
End Sub

Private Sub cmdCancel_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then cmdCancel.ShowWhatsThis
End Sub

Private Sub cmdOK_Click()
    Dim SV As Boolean
    Dim LI As Long

    On Error GoTo Err1 'Err

    LI = lstCase.ListIndex

    If LI < 0 Then
        MsgBox "解析ケースが選択されていません。"
        Exit Sub
    End If

    '解析ケース名の取得
    modSTAR.CurrentCase$ = lstCase.List(LI)
    modSTAR.nmAID = lstCase.ItemData(LI)
    Debug.Print lstCase.List(LI); lstCase.ItemData(LI)

    SV = modSTAR.SuperVisor

    '各種コマンドを実行可に設定
    modSTAR.frmMain.mnuSelShape.Enabled = True
    modSTAR.frmMain.mnuPriCrackInt.Enabled = SV
    modSTAR.frmMain.mnuSecAnalShow.Enabled = True
    modSTAR.frmMain.mnuSecTableCase.Enabled = SV
    modSTAR.frmMain.mnuSecTableShow.Enabled = True
    modSTAR.frmMain.mnuSecDamageCase.Enabled = SV
    modSTAR.frmMain.mnuSecDamageShow.Enabled = True
    modSTAR.frmMain.mnuBunExec.Enabled = True
    modSTAR.frmMain.mnuTokInterCase.Enabled = SV
    modSTAR.frmMain.mnuTokDataView.Enabled = True

    'Normalフォームの更新
    modSTAR.frmMain.tblCase.Caption = modSTAR.CurrentCase$

    Unload Me

    Exit Sub 'Err

Err1:
    MsgBox "Error in frmSelCase2/cmdOK" 'Err
    Call ShowErrorMessage 'Err
End Sub

Private Sub cmdOK_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then cmdOK.ShowWhatsThis
End Sub

Private Sub Form_Load()
    Dim dtTMP As rdoResultset
    Dim SQL$
    Dim i As Long
    Dim IData As Long
    Dim hTMP As Integer

    On Error GoTo Err1 'Err

    Screen.MousePointer = 11

    SQL$ = "SELECT 解析ケース,解析id FROM 管理テーブル WHERE (" & _
        & "(試験装置 = '" & modSTAR.nmSochi$ & "') AND " & _
        & "(試験体 = '" & modSTAR.nmTai$ & "') AND " & _
        & "(試験部分 = '" & modSTAR.nmBubun$ & "') AND " & _
        & "(解析ケース LIKE 'ケース*) AND " & _
        & "(評価法id = 0)) ORDER BY 解析ケース;"

```



```

' Set dtTMP = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
Set dtTMP = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hTMP)
If hTMP = -1 Then GoTo Err1

j = 0
Do Until dtTMP.EOF
  lstCase.AddItem dtTMP![解析ヶ-入]
  IData = dtTMP![解析id]
  lstCase.ItemData(i) = IData
  j = j + 1
  dtTMP.MoveNext
Loop

' dtTMP.Close
xCloseResultset dtTMP, hTMP

cmdOK.Enabled = False

Screen.MousePointer = 0

Exit Sub                                     'Err

Err1:
Screen.MousePointer = 0                       'Err
MsgBox "Error in frmSelCase2/Load"           'Err
Call ShowErrorMessage                         'Err
On Error Resume Next                         'Err
' dtTMP.Close                                'Err
xCloseResultset dtTMP, hTMP                  'Err
Unload Me                                     '強制終了

End Sub

Private Sub lblCase_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
  If Button = 2 Then lblCase.ShowWhatsThis
End Sub

Private Sub lstCase_Click()
  If lstCase.ListIndex >= 0 Then
    cmdOK.Enabled = True
  End If
End Sub

Private Sub lstCase_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
  If Button = 2 Then lstCase.ShowWhatsThis
End Sub

```

```

Attribute VB_Name = "frmSelView"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Option Explicit

```

```

Private Sub cmdOK_Click()
    Unload Me
End Sub

```

```

Private Sub cmdOK_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then cmdOK.ShowWhatsThis
End Sub

```

```

Private Sub Form_Load()
    On Error GoTo Err1 'Err

    txtSochi = modSTAR.nmSochi$
    txtTai = modSTAR.nmTai$
    txtBubun = modSTAR.nmBubun$
    txtCase = modSTAR.CurrentCase$

    Exit Sub 'Err

Err1:
    MsgBox "Error in frmSelView/Load" 'Err
    Call ShowErrorMessage 'Err
    Unload Me 'Err 強制エラー発生
End Sub

```

```

Private Sub lblBubun_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblBubun.ShowWhatsThis
End Sub

```

```

Private Sub lblCase_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblCase.ShowWhatsThis
End Sub

```

```

Private Sub lblSochi_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblSochi.ShowWhatsThis
End Sub

```

```

Private Sub lblTai_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblTai.ShowWhatsThis
End Sub

```

```

Private Sub txtBubun_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then txtBubun.ShowWhatsThis
End Sub

```

```

Private Sub txtCase_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then txtCase.ShowWhatsThis
End Sub

```

```

Private Sub txtSochi_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then txtSochi.ShowWhatsThis
End Sub

```

```
Private Sub txtTai_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
  If Button = 2 Then txtTai.ShowWhatsThis
End Sub
```

```

Attribute VB_Name = "frmStarMain"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Option Explicit

Private Sub MDIForm_Load()
    Dim i As Long
    Dim SV As Boolean
    Public MaxCase As Long           ' 解析ケース番号の最大値

    ' カレントディレクトリの取得
    modSTAR.InpDir$ = CurDir

    ' ヘルプファイルの設定
    App.HelpFile = App.Path & "%STAR.HLP"

    ' レジストリの取得
    modSTAR.DSN$ = GetSetting("STAR7", "STAR", "DataSourceName")
    modSTAR.DBN$ = GetSetting("STAR7", "STAR", "DatabaseName")
    modSTAR.QDB$ = GetSetting("STAR7", "STAR", "QueryDB")
    modSTAR.BTView = Val(GetSetting("STAR7", "STAR", "BTView"))
    modSTAR.MaxRet = Val(GetSetting("STAR7", "STAR", "MaxRetrieve"))
    modSTAR.WorkDir$ = GetSetting("STAR7", "STAR", "WorkDir")
    modSTAR.WSketchDir$ = GetSetting("STAR7", "STAR", "WSketchDir")

    ' レジストリのチェック
    If modSTAR.DSN$ = "" Or _
        modSTAR.DBN$ = "" Or _
        modSTAR.QDB$ = "" Or _
        modSTAR.WorkDir$ = "" Then
        i = MsgBox("STARの環境設定がすんでいません。環境設定を行いますか?", vbYesNo)
        If i = vbYes Then
            frmUtilSetup.Show 1
        Else
            End
        End If
    End If

    If Dir(modSTAR.QDB$) = "" Then
        i = MsgBox(modSTAR.QDB$ & "が見つかりません。環境設定を行いますか?", vbYesNo)
        If i = vbYes Then
            frmUtilSetup.Show 1
        Else
            End
        End If
    End If

    If Dir(modSTAR.WorkDir$, vbDirectory) = "" Then
        i = MsgBox(modSTAR.WorkDir$ & "が見つかりません。環境設定を行いますか?", vbYesNo)
        If i = vbYes Then
            frmUtilSetup.Show 1
        Else
            End
        End If
    End If

    If modSTAR.WSketchDir$ <> "" And Dir(modSTAR.WSketchDir$ & "%SKETCH.EXE") = "" Then
        i = MsgBox(modSTAR.WSketchDir$ & "%SKETCH.EXEが見つかりません。環境設定を行いますか?", vbYesNo)
        If i = vbYes Then
            frmUtilSetup.Show 1
        Else
            End
        End If
    End If

    If modSTAR.BTView <> 0 And modSTAR.BTView <> 1 Then modSTAR.BTView = 0
    If modSTAR.MaxRet = 0 Then modSTAR.MaxRet = 300

    ' システムファイル名の設定
    modSTAR.StarDB$ = modSTAR.DBDir$ & "STAR.MDB"           ' STAR管理データベース名
    modSTAR.SysQDB$ = modSTAR.DBDir$ & "STAR.MDB"         ' システムのクエリデータベース
    modSTAR.QueryDB$ = modSTAR.SysQDB$                     ' システムのクエリデータベース
    modSTAR.QueryDB$ = modSTAR.QDB$

    ' ログイン処理
    frmUserCheck2.Show 1
    SV = modSTAR.SuperVisor
    Debug.Print "StarVisor", SV

    ' キャプションの変更
    If SV = True Then
        frmStarMain.Caption = "PNC STAR - (" & UID$ & ") - Super)"
    Else
        frmStarMain.Caption = "PNC STAR - (" & UID$ & ") - General)"
    End If

    ' 各種初期値の設定
    modSTAR.posSochi = -1           ' 管理TBLの現試験装置の位置
    modSTAR.posTai = -1            ' 管理TBLの現試験体の位置
    modSTAR.posBubun = -1          ' 管理TBLの現試験部分の位置
    modSTAR.nmSochi$ = ""          ' 現試験装置名

```

```

modSTAR.nmTai$ = "" ' 現試験体名
modSTAR.nmBubun$ = "" ' 現試験部分名
modSTAR.nmID = 0 ' 参照id
modSTAR.nmAID = 0 ' 解析の参照ID

modSTAR.KanriLine1 = -1 ' 管理TBLの現在の行位置(試験部分へ)
modSTAR.KanriLine2 = -1 ' 管理TBLの現在の行位置(解析ケースへ)
modSTAR.CurrentCase$ = "" ' 現在の解析ケース名

modSTAR.tmpViewNo = 0

modSTAR.tmpBunNo = 0
modSTAR.CurBunLineNo = -1
modSTAR.CurBunQueryNo = -1
modSTAR.CurBunQueryName$ = ""

modSTAR.tmpTokNo = 0
modSTAR.CurTokFromName$ = ""
modSTAR.CurTokQueryNo = -1
modSTAR.CurTokQueryName$ = ""

' VBのフィールドタイプをSQLの文字列に変換
modSTAR.CnvFldType$(0) = ""
modSTAR.CnvFldType$(1) = ""
modSTAR.CnvFldType$(2) = " BYTE "
modSTAR.CnvFldType$(3) = " SHORT "
modSTAR.CnvFldType$(4) = " LONG "
modSTAR.CnvFldType$(5) = " CURRENCY "
modSTAR.CnvFldType$(6) = " SINGLE "
modSTAR.CnvFldType$(7) = " DOUBLE "
modSTAR.CnvFldType$(8) = " DATETIME "
modSTAR.CnvFldType$(9) = ""
modSTAR.CnvFldType$(10) = " TEXT "
modSTAR.CnvFldType$(11) = " BIT "
modSTAR.CnvFldType$(12) = ""
modSTAR.CnvFldType$(13) = ""
modSTAR.CnvFldType$(14) = ""

' 初期フォームを表示
modSTAR.frmMain.Show
modSTAR.frmMain.Caption = "未定義"
modSTAR.frmMain.lblSochi.Caption = ""
modSTAR.frmMain.lblTai.Caption = ""
modSTAR.frmMain.lblBubun.Caption = ""
modSTAR.frmMain.lblCase.Caption = ""

' クラックを削除
Call KillFile(modSTAR.WorkDir & "$*.*)"

modSTAR.AFldMax = 86

Call SetAnalField(0, 0, 0, " " "id" " " "id")
Call SetAnalField(1, 0, 1, " " "参照id" " " "参照id")
Call SetAnalField(2, 0, 2, " " "解析id" " " "解析id")
Call SetAnalField(3, 0, 3, " " "日付" " " "日付")
Call SetAnalField(4, 0, 4, " " "解析ケース" " " "解析ケース")
Call SetAnalField(5, 0, 5, " " "ライン番号" " " "ライン番号")
Call SetAnalField(6, 0, 6, " " "L" " " "L")
Call SetAnalField(7, 0, 7, " " "Q" " " "Q")
Call SetAnalField(8, 0, 8, "R1" " " "起点r1" " " "起点r1")
Call SetAnalField(9, 0, 9, "Z1" " " "起点z1" " " "起点z1")
Call SetAnalField(10, 0, 10, "H1" " " "起点θ1" " " "起点θ1")
Call SetAnalField(11, 0, 11, "R2" " " "終点r2" " " "終点r2")
Call SetAnalField(12, 0, 12, "Z2" " " "終点z2" " " "終点z2")
Call SetAnalField(13, 0, 13, "H2" " " "終点θ2" " " "終点θ2")
Call SetAnalField(14, 0, 14, "TEMPMAX" " " "TEMPMAX" " " "最高金属温度", "TEMPMAX")
Call SetAnalField(15, 0, 15, " " "時点id" " " "時点id")
Call SetAnalField(16, 0, 16, " " "時点2id" " " "時点2id")
Call SetAnalField(17, 0, 17, " " "試験条件id" " " "試験条件id")

Call SetAnalField(18, 1, 0, " " "id" " " "id")
Call SetAnalField(19, 1, 1, " " "参照id" " " "参照id")
Call SetAnalField(20, 1, 1, " " "解析id" " " "解析id")
Call SetAnalField(21, 1, 1, " " "解析ケース" " " "解析ケース")
Call SetAnalField(22, 1, 1, "TIME1" " " "TIME1" " " "評価時点1", "評価時点")
Call SetAnalField(23, 1, 2, "TEMP1" " " "TEMP1" " " "表面温度1", "TEMP")
Call SetAnalField(24, 1, 3, "LTEMP1" " " "LTEMP1" " " "等価線形温度1", "LTEMP")
Call SetAnalField(25, 1, 4, "MTEMP1" " " "MTEMP1" " " "平均温度1", "MTEMP")
Call SetAnalField(26, 1, 5, "STRS11" " " "σr1" " " "表面応力σr1", "σr")
Call SetAnalField(27, 1, 6, "STRS21" " " "σz1" " " "表面応力σz1", "σz")
Call SetAnalField(28, 1, 7, "STRS31" " " "σθ1" " " "表面応力σθ1", "σθ")
Call SetAnalField(29, 1, 8, "STRS41" " " "τrz1" " " "表面応力τrz1", "τrz")
Call SetAnalField(30, 1, 9, "LSTRS11" " " "Lσr1" " " "等価線形応力σr1", "Lσr")
Call SetAnalField(31, 1, 10, "LSTRS21" " " "Lσz1" " " "等価線形応力σz1", "Lσz")
Call SetAnalField(32, 1, 11, "LSTRS31" " " "Lσθ1" " " "等価線形応力σθ1", "Lσθ")
Call SetAnalField(33, 1, 12, "LSTRS41" " " "Lτrz1" " " "等価線形応力τrz1", "Lτrz")
Call SetAnalField(34, 1, 13, "MSTRS11" " " "Mσr1" " " "膜応力σr1", "Mσr")
Call SetAnalField(35, 1, 14, "MSTRS21" " " "Mσz1" " " "膜応力σz1", "Mσz")
Call SetAnalField(36, 1, 15, "MSTRS31" " " "Mσθ1" " " "膜応力σθ1", "Mσθ")
Call SetAnalField(37, 1, 16, "MSTRS41" " " "Mτrz1" " " "膜応力τrz1", "Mτrz")
Call SetAnalField(38, 1, 17, "VMS1" " " "VMS1" " " "VMS")
Call SetAnalField(39, 1, 18, "TRES1" " " "TRES1" " " "TRES")
Call SetAnalField(40, 1, 19, "EE1" " " "εr1" " " "弾性歪εr1", "εr")
Call SetAnalField(41, 1, 20, "EE2" " " "εz1" " " "弾性歪εz1", "εz")
Call SetAnalField(42, 1, 21, "EE3" " " "εθ1" " " "弾性歪εθ1", "εθ")
Call SetAnalField(43, 1, 22, "EE4" " " "γrz1" " " "弾性歪γrz1", "γrz")
Call SetAnalField(44, 1, 23, "EP1" " " "εpr1" " " "塑性歪εpr1", "εpr")

```

```

Call SetAnalField(45, 1, 24, "EP21", "εpz1", "塑性歪εz1", "εpz")
Call SetAnalField(46, 1, 25, "EP31", "εpθ1", "塑性歪εθ1", "εpθ")
Call SetAnalField(47, 1, 26, "EP41", "γprz1", "塑性歪γrz1", "γprz")
Call SetAnalField(48, 1, 27, "EC11", "εcr1", "クリープ歪εr1", "εcr")
Call SetAnalField(49, 1, 28, "EC21", "εcz1", "クリープ歪εz1", "εcz")
Call SetAnalField(50, 1, 29, "EC31", "εcθ1", "クリープ歪εθ1", "εcθ")
Call SetAnalField(51, 1, 30, "EC41", "γcrz1", "クリープ歪γrz1", "γcrz")

Call SetAnalField(52, 2, 0, "id", "id", "id")
Call SetAnalField(53, 2, 1, "参照id", "参照id", "参照id")
Call SetAnalField(54, 2, 1, "解析id", "解析id", "解析id")
Call SetAnalField(55, 2, 1, "解析γ-2", "解析γ-2", "解析γ-2")
Call SetAnalField(56, 2, 1, "TIME2", "評価時点2", "評価時点")
Call SetAnalField(57, 2, 2, "TEMP2", "表面温度2", "TEMP")
Call SetAnalField(58, 2, 3, "LTEMP2", "LTEMP2", "等価線形温度2", "LTEMP")
Call SetAnalField(59, 2, 4, "MTEMP2", "MTEMP2", "平均温度2", "MTEMP")
Call SetAnalField(60, 2, 5, "STRS12", "σr2", "表面応力σr2", "σr")
Call SetAnalField(61, 2, 6, "STRS22", "σz2", "表面応力σz2", "σz")
Call SetAnalField(62, 2, 7, "STRS32", "σθ2", "表面応力σθ2", "σθ")
Call SetAnalField(63, 2, 8, "STRS42", "τrz2", "表面応力τrz2", "τrz")
Call SetAnalField(64, 2, 9, "LSTRS12", "Lσr2", "等価線形応力σr2", "Lσr")
Call SetAnalField(65, 2, 10, "LSTRS22", "Lσz2", "等価線形応力σz2", "Lσz")
Call SetAnalField(66, 2, 11, "LSTRS32", "Lσθ2", "等価線形応力σθ2", "Lσθ")
Call SetAnalField(67, 2, 12, "LSTRS42", "Lτrz2", "等価線形応力τrz2", "Lτrz")
Call SetAnalField(68, 2, 13, "MSTRS12", "Mσr2", "膜応力σr2", "Mσr")
Call SetAnalField(69, 2, 14, "MSTRS22", "Mσz2", "膜応力σz2", "Mσz")
Call SetAnalField(70, 2, 15, "MSTRS32", "Mσθ2", "膜応力σθ2", "Mσθ")
Call SetAnalField(71, 2, 16, "MSTRS42", "Mτrz2", "膜応力τrz2", "Mτrz")
Call SetAnalField(72, 2, 17, "VMS2", "VMS2", "VMS")
Call SetAnalField(73, 2, 18, "TRES2", "TRES2", "TRES")
Call SetAnalField(74, 2, 19, "EE12", "εr2", "弾性歪εr2", "εr")
Call SetAnalField(75, 2, 20, "EE22", "εz2", "弾性歪εz2", "εz")
Call SetAnalField(76, 2, 21, "EE32", "εθ2", "弾性歪εθ2", "εθ")
Call SetAnalField(77, 2, 22, "EE42", "γrz2", "弾性歪γrz2", "γrz")
Call SetAnalField(78, 2, 23, "EP12", "εpr2", "塑性歪εr2", "εpr")
Call SetAnalField(79, 2, 24, "EP22", "εpz2", "塑性歪εz2", "εpz")
Call SetAnalField(80, 2, 25, "EP32", "εpθ2", "塑性歪εθ2", "εpθ")
Call SetAnalField(81, 2, 26, "EP42", "γprz2", "塑性歪γrz2", "γprz")
Call SetAnalField(82, 2, 27, "EC12", "εcr2", "クリープ歪εr2", "εcr")
Call SetAnalField(83, 2, 28, "EC22", "εcz2", "クリープ歪εz2", "εcz")
Call SetAnalField(84, 2, 29, "EC32", "εcθ2", "クリープ歪εθ2", "εcθ")
Call SetAnalField(85, 2, 30, "EC42", "γcrz2", "クリープ歪γrz2", "γcrz")

```

End Sub

```

Private Sub MDIForm_QueryUnload(Cancel As Integer, UnloadMode As Integer)
    dbSTAR.Close '親ウインドウの閉じるが有効にする
    Close #9 '親ウインドウの閉じるが有効にする
End Sub
End Sub

```

```

Private Sub mnuSelExit_Click()
    dbSTAR.Close
End Sub
End Sub

```

```
Attribute VB_Name = "frmTokDataSel"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
```

```
Private Sub cmdCancel_Click()
    modSTAR.tmpFormCloseType = vbCancel
    Unload Me
End Sub
```

```
Private Sub cmdCancel_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then cmdCancel.ShowWhatsThis
End Sub
```

```
Private Sub cmdOK_Click()
    Dim Target$$
    Dim SQL$, SQLW$
    Dim LI As Long

    On Error GoTo Err1

    If frmTokDataSel.optCrk.Value = True Then '1:き裂
        Target$$ = "き裂" & "_S"
    ElseIf frmTokDataSel.optAnl.Value = True Then '2:解析
        Target$$ = "解析" & "_S"
    ElseIf frmTokDataSel.optTbl.Value = True Then '3:表計算
        Target$$ = "表計算" & "_S"
    ElseIf frmTokDataSel.optDmg.Value = True Then '4:損傷値
        LI = frmTokDataSel.cmbDmgSel.ListIndex
        If LI < 0 Then
            MsgBox "評価法が選択されていません。"
            Exit Sub
        End If
        Target$$ = frmTokDataSel.cmbDmgSel.List(LI) & "_S"
    End If

    If Target$$ = "き裂_S" Then
        SQLW$ = " WHERE (参照id = " & modSTAR.nmID & ");"
        SQL$ = "SELECT * FROM " & Target$$ & SQLW$
    ElseIf Target$$ = "解析_S" Then
        SQL$ = "SELECT 解析_S.*, 時点1_S.*, 時点2_S.*"
        & "FROM (解析_S INNER JOIN 時点_S AS 時点1_S ON 解析_S.時点1id = 時点1_S.id) " _
        & "INNER JOIN 時点_S AS 時点2_S ON 解析_S.時点2id = 時点2_S.id " _
        & "WHERE (解析_S.解析id = " & modSTAR.nmAID & ");"
    Else
        SQLW$ = " WHERE (解析id = " & modSTAR.nmAID & ");"
        SQL$ = "SELECT * FROM " & Target$$ & SQLW$
    End If
    Debug.Print SQL$

    modSTAR.tmpFormName$ = Target$$
    modSTAR.tmpViewSQL$ = SQL$
    modSTAR.tmpDeleteSQL$ = "DELETE " & Target$$ & SQLW$

    modSTAR.tmpFormCloseType = vbOK

    Unload Me
Exit Sub
```

```
Err1:
    MsgBox "Error in frmTokDataSel/cmdOK"
    Call ShowErrorMessage
End Sub
```

```
Private Sub cmdOK_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then cmdOK.ShowWhatsThis
End Sub
```

```
Private Sub Form_Load()
    Dim dtKANRI As rdoResultset
    Dim SQL$
    Dim hKANRI As Integer

    On Error GoTo Err1

    Screen.MousePointer = 11
    cmbDmgSel.Text = ""
```

```

cmbDmgSel.Enabled = False

SQL$ = "SELECT 評価法id FROM 管理データファイル WHERE (" _
      & "(参照id = " & modSTAR.nmID & ") AND " _
      & "(解析ケース = " & modSTAR.CurrentCase$ & "-" ) AND " _
      & "(評価法id > 0))" _
      & " ORDER BY id;" _
Debug.Print SQL$
'Set dtKANRI = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
Set dtKANRI = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hKANRI)
If hKANRI = -1 Then GoTo Err1

Do Until dtKANRI.EOF
  Debug.Print "評価法id "; dtKANRI![評価法id]
  cmbDmgSel.AddItem GetHyokaName(dtKANRI![評価法id])
  dtKANRI.MoveNext
Loop

'dtKANRI.Close
xCloseResultset dtKANRI, hKANRI

cmbDmgSel.ListIndex = -1

modSTAR.tmpFormCloseType = vbCancel

Screen.MousePointer = 0

Exit Sub

Err1:
Screen.MousePointer = 0
MsgBox "Error in frmTokDataSel/Load"
Call ShowErrorMessage
modSTAR.tmpFormCloseType = vbCancel
On Error Resume Next
'dtKANRI.Close
xCloseResultset dtKANRI, hKANRI
Unload Me '強制エラー

End Sub

Private Sub fraTarget_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
  If Button = 2 Then fraTarget.ShowWhatsThis
End Sub

Private Sub optAn1_Click()
  cmbDmgSel.Enabled = False
End Sub

Private Sub optAn1_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
  If Button = 2 Then optAn1.ShowWhatsThis
End Sub

Private Sub optCrk_Click()
  cmbDmgSel.Enabled = False
End Sub

Private Sub optCrk_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
  If Button = 2 Then optCrk.ShowWhatsThis
End Sub

Private Sub optDmg_Click()
  cmbDmgSel.Enabled = True
End Sub

Private Sub optDmg_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
  If Button = 2 Then optDmg.ShowWhatsThis
End Sub

Private Sub optTbl_Click()
  cmbDmgSel.Enabled = False

```


End Sub

Private Sub optTbl_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)

 If Button = 2 Then optTbl.ShowWhatsThis

End Sub

```

Attribute VB Name = "frmTokIntCtrl"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Option Explicit

Private SQL$
Private TergetLevel As Long
Private Terget$
Private Terget$$
Private nmFLD$

Private CrkS(600, 3) As Variant      '統計用のき裂データ
Private Tgt(3, 300, 100) As Variant '計算データ(解析,表計算,損傷値)
Private TgtS(600, 100) As Variant   '統計用の計算データ

Private Sub cmdCancel_Click()
    Unload Me
End Sub

Private Sub cmdCancel_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then cmdCancel.ShowWhatsThis
End Sub

Private Sub cmdOK_Click()
    Dim dtKANRI As rdoResultset
    Dim dtCRKS As rdoResultset
    Dim dtTRG As rdoResultset
    Dim dtTRGS As rdoResultset
    Dim dtANLS As rdoResultset
    Dim dtJTIS As rdoResultset
    Dim dtJT2S As rdoResultset
    Dim nmCase$, SQL$
    Dim nmID As Long
    Dim nmAID As Long
    Dim nmHID As Long
    Dim MaxID As Long
    Dim i As Long
    Dim j As Long
    Dim k As Long
    Dim m As Long
    Dim CSRows As Long
    Dim TRows As Long
    Dim TCols As Long
    Dim LineNo As Long
    Dim CrkS(), Tgt(), TgtS() As Variant
    Dim obj As Object
    Dim hKANRI As Integer
    Dim hCRKS As Integer
    Dim hTRG As Integer
    Dim hTRGS As Integer
    Dim hANLS As Integer
    Dim hJTIS As Integer
    Dim hJT2S As Integer

    On Error GoTo Err1

    frmTokIntCtrl.Height = 6375

    Screen.MousePointer = 11

    Set obj = CreateObject("INTPSV.INTP")

    'Set dtKANRI = dbSTAR5.OpenResultset(SQLX$, rdOpenStatic, rdConcurReadOnly)
    Set dtKANRI = xOpenResultset(SQLX$, rdOpenStatic, rdConcurReadOnly, hKANRI)
    If hKANRI = -1 Then GoTo Err1

    '-----
    'If dtKANRI.EOF = False Then
    '    Grid1.Visible = True
    '    frmTokIntCtrl.Height = 8300
    '    dtKANRI.MoveLast
    '    Grid1.Rows = dtKANRI.RowCount + 1
    '    Grid1.Cols = dtKANRI.rdoColumns.Count + 1
    '    dtKANRI.MoveFirst
    '    i = 1
    '    Do Until dtKANRI.EOF
    '        Grid1.Row = i
    '        For j = 0 To dtKANRI.rdoColumns.Count - 1
    '            Grid1.Col = j + 1
    '            Grid1.Text = dtKANRI.rdoColumns(j).Value
    '        Next j
    '        i = i + 1
    '        dtKANRI.MoveNext
    '    Loop
    'End If
    dtKANRI.Close

```

```

' Set obj = Nothing
' Screen.MousePointer = 0
' Exit Sub
'-----
' dtKANRI.MoveFirst

Do Until dtKANRI.EOF

    nmID = dtKANRI![参照id]
    nmAID = dtKANRI![解析id]
    nmHID = dtKANRI![評価法id]
    nmCase$ = dtKANRI![解析ケース]

    ' 計算データ名の取得(損傷値の場合)
    If nmHID > 0 Then Target$ = GetHyokaName(nmHID)
    Target$$ = Target$ & "_S"

    Debug.Print nmID; nmCase$; Target$

    ' き裂データのオープン
    SQL$ = "SELECT id, ライン番号, L FROM き裂_S WHERE " _
        & "(参照id = " & nmID & ")" _
        & " ORDER BY ライン番号, L;" _
    Debug.Print "き裂S"; SQL$
    ' Set dtCRKS = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
    Set dtCRKS = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hCRKS)
    If hCRKS = -1 Then GoTo Err1

    If dtCRKS.EOF = True Then
        lstError.AddItem "参照id = " & nmID & " のき裂_Sデータがありません"
        lstError.Refresh
        If TargetLevel = 3 Then
            MsgBox "参照id=" & nmID & " のき裂_Sデータがありません"
        End If
        xCloseResultset dtCRKS, hCRKS
        GoTo NextKanri
    End If

    ' 計算データのオープン
    If Target$ = "解析" Then
        SQL$ = "SELECT 解析.*, 時点1.*, 時点2.*" _
            & " FROM (解析 INNER JOIN 時点 AS 時点1 ON 解析.時点id = 時点1.id) " _
            & " INNER JOIN 時点 AS 時点2 ON 解析.時点2id = 時点2.id " _
            & " WHERE (解析.解析id = " & nmAID & ")" _
            & " ORDER BY 解析.ライン番号, 解析.L;"
    Else
        SQL$ = "SELECT * FROM " & Target$ & " WHERE " _
            & "(解析id = " & nmAID & ")" _
            & " ORDER BY ライン番号, L;"
    End If
    Debug.Print "対象 "; SQL$
    ' Set dtTRG = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
    Set dtTRG = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hTRG)
    If hTRG = -1 Then GoTo Err1

    If dtCRKS.EOF = False And dtTRG.EOF = False Then

        ' き裂データの読み込み
        dtCRKS.MoveLast
        CSRows = dtCRKS.RowCount
        Debug.Print CSRows
        obj.CrackSRows = CSRows
        ' ReDim CrkS(CSRows, 2), TgtS(3, CSRows, 100)
        i = 0
        dtCRKS.MoveFirst
        Do Until dtCRKS.EOF
            CrkS(i, 0) = dtCRKS![ライン番号]
            CrkS(i, 1) = dtCRKS![L]
            CrkS(i, 2) = dtCRKS![id]
            Debug.Print i; CrkS(i, 0); CrkS(i, 1); CrkS(i, 2)
            i = i + 1
        dtCRKS.MoveNext
    Loop

        ' 計算データの読み込み
        dtTRG.MoveLast
        TRows = dtTRG.RowCount
        TCols = dtTRG.rdoColumns.Count
        Debug.Print "TRows,TCols:"; TRows; TCols
        obj.TargetRows = TRows
        obj.TargetCols = TCols + 1
        ' ReDim Tgt(3, TRows, 100)
        m = -1
        i = 0
        dtTRG.MoveFirst
        Do Until dtTRG.EOF
            If (dtTRG![ライン番号] - 1) <> m Then
                m = m + 1
                i = 0
            End If
            Tgt(m, i, 0) = dtTRG![L]
            Debug.Print m; i; 0; Tgt(m, i, 0)
            For j = 0 To dtTRG.rdoColumns.Count - 1
                If IsNull(dtTRG.rdoColumns(j).Value) = False Then
                    Tgt(m, i, j + 1) = dtTRG.rdoColumns(j).Value
                End If
            Next j
        Loop
    End If

```

```

    i = i + 1
    dtTRG.MoveNext
Loop
' 計算の実行
Debug.Print "Exec Start"
obj.Exec CrkS, Tgt, TgtS
Debug.Print "Exec End"

''チェックプリント
'Grid1.Rows = CSRows + 1
'Grid1.Cols = TCols + 1
'For i = 0 To CSRows - 1
'    Grid1.Row = i + 1
'    For j = 0 To TCols - 1
'        Grid1.Col = j + 1
'        Grid1.Text = TgtS(i, j + 1)
'    Next j
'Next i

'古い統計用計算テーブルを削除しておく
If TargetS = "解析" Then
    Call DeleteAnalRC("解析_S", "時点_S", nmAID, nmCaseS)
    Call DeleteAnalA("解析_S", "時点_S", nmAID)
Else
    Call DeleteDataRC(TargetS$, "解析id", nmAID, nmCaseS)
    Call DeleteDataA(TargetS$, nmAID)
End If

'補間結果の書き込み
If TargetS = "解析" Then ' 解析テーブルの書き込み

    MaxID = GetMaxID("時点_S", "id")
    SQL$ = "SELECT * FROM 解析_S WHERE " _
        & "(解析id = " & nmAID & ");"
    'Set dtANLS = dbSTAR5.OpenResultset(SQL$, rdOpenDynamic, rdConcurLock)
    Set dtANLS = xOpenResultset(SQL$, rdOpenDynamic, rdConcurLock, hANLS)
    If hANLS = -1 Then GoTo Err1

    SQL$ = "SELECT * FROM 時点_S WHERE (id = 0);"
    'Set dtJT1S = dbSTAR5.OpenResultset(SQL$, rdOpenDynamic, rdConcurLock)
    Set dtJT1S = xOpenResultset(SQL$, rdOpenDynamic, rdConcurLock, hJT1S)
    If hJT1S = -1 Then GoTo Err1
    'Set dtJT2S = dbSTAR5.OpenResultset(SQL$, rdOpenDynamic, rdConcurLock)
    Set dtJT2S = xOpenResultset(SQL$, rdOpenDynamic, rdConcurLock, hJT2S)
    If hJT2S = -1 Then GoTo Err1
    For i = 0 To CSRows - 1
        dtANLS.AddNew
        dtJT1S.AddNew
        dtJT2S.AddNew
        m = 0
        For j = 0 To dtANLS.rdoColumns.Count - 1 ' 解析_S
            m = m + 1
            dtANLS.rdoColumns(j).Value = TgtS(i, m)
        Next j
        For j = 0 To dtJT1S.rdoColumns.Count - 1 ' 時点1_S
            m = m + 1
            dtJT1S.rdoColumns(j).Value = TgtS(i, m)
        Next j
        For j = 0 To dtJT2S.rdoColumns.Count - 1 ' 時点2_S
            m = m + 1
            dtJT2S.rdoColumns(j).Value = TgtS(i, m)
        Next j
        dtANLS![時点1id] = MaxID + 1
        dtJT1S![id] = MaxID + 1
        dtANLS![時点2id] = MaxID + 2
        dtJT2S![id] = MaxID + 2
        MaxID = MaxID + 2
        dtANLS.Update
        dtJT1S.Update
        dtJT2S.Update
    Next i
    dtJT2S.Close
    xCloseResultset dtJT2S, hJT2S
    dtJT1S.Close
    xCloseResultset dtJT1S, hJT1S
    dtANLS.Close
    xCloseResultset dtANLS, hANLS

Else ' 表計算と損傷値

    SQL$ = "SELECT * FROM " & TargetS$ & " WHERE " _
        & "(解析id = " & nmAID & ");"
    'Set dtTRGS = dbSTAR5.OpenResultset(SQL$, rdOpenDynamic, rdConcurLock)
    Set dtTRGS = xOpenResultset(SQL$, rdOpenDynamic, rdConcurLock, hTRGS)
    If hTRGS = -1 Then GoTo Err1

    For i = 0 To CSRows - 1
        dtTRGS.AddNew
        For j = 0 To TCols - 1
            dtTRGS.rdoColumns(j).Value = TgtS(i, j + 1)
        Next j
        dtTRGS.Update
    Next i

    dtTRGS.Close
    xCloseResultset dtTRGS, hTRGS

```

```

End If

End If

'dtCRKS.Close          'き裂_シ-フルをク-ス
xCloseResultset dtCRKS, hCRKS 'き裂_シ-フルをク-ス
'dtTRG.Close          '計算_シ-フルをク-ス
xCloseResultset dtTRG, hTRG '計算_シ-フルをク-ス

'Call SetKanriXID(nmFLD$, nmAID, nmID, nmCase$, nmHID)
Call SetKanriXID6(nmFLD$, nmAID, nmID, nmCase$, nmHID)

NextKanri:
  dtKANRI.MoveNext      '管理_シ-フルの次の行へ
Loop

'dtKANRI.Close          '管理_シ-フルをク-ス
xCloseResultset dtKANRI, hKANRI '管理_シ-フルをク-ス

Debug.Print " "

Set obj = Nothing

Screen.MousePointer = 0

Unload Me

Exit Sub

Err1:
Screen.MousePointer = 0
MsgBox "Error in frmTokIntCtrl/cmdOK"
Call ShowErrorMessage
On Error Resume Next
'dtKANRI.Close
'dtCRKS.Close
'dtTRG.Close
'dtTRGS.Close
'dtANLS.Close
'dtJTIS.Close
'dtJT2S.Close
xCloseResultset dtKANRI, hKANRI
xCloseResultset dtCRKS, hCRKS
xCloseResultset dtTRG, hTRG
xCloseResultset dtTRGS, hTRGS
xCloseResultset dtANLS, hANLS
xCloseResultset dtJTIS, hJTIS
xCloseResultset dtJT2S, hJT2S
Set obj = Nothing

End Sub

Private Sub cmdOK_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
  If Button = 2 Then cmdOK.ShowWhatsThis
End Sub

Private Sub Form_Load()
  Dim dtKANRI As rdoResultset
  Dim TergetType As Long
  Dim TergetSubType As Long
  Dim SQL$, SQL0$, SQL1$, SQL2$, Buf$
  Dim Terget$
  Dim hKANRI As Integer

  On Error GoTo Err1

  frmTokIntCtrl.Height = 5000

  Screen.MousePointer = 11

  TergetLevel = modSTAR.tmpTokInterSel          '対象レベル

  If frmTokInterSel.optAnal.Value = True Then   '1:解析
    TergetType = 1
    Terget$ = "解析"
    nmFLD$ = "解析Sid"
  ElseIf frmTokInterSel.optTable.Value = True Then '2:表計算
    TergetType = 2
    Terget$ = "表計算"
    nmFLD$ = "表計算Sid"
  ElseIf frmTokInterSel.optDamage.Value = True Then '3:損傷値
    TergetType = 3
    TergetSubType = frmTokInterSel.cmbDmgSel.ListIndex
    Terget$ = frmTokInterSel.cmbDmgSel.List(TergetSubType)
    nmFLD$ = "損傷値Sid"
  End If

  '対象レベルによる管理_シ-フルの検索SQLの作成
  SQL0$ = "SELECT * FROM 管理_シ-フル WHERE ("

```

```

If TargetLevel = 1 Then                                '1:全テーム
  SQL1$ = ""
Elseif TargetLevel = 2 Then                            '2:現試験体
  SQL1$ = "(試験装置 = " & modSTAR.nmSochi$ & ") AND " -
  & "(試験体 = " & modSTAR.nmTai$ & ") AND " -
Elseif TargetLevel = 3 Then                            '3:現解析ケース
  SQL1$ = "(試験装置 = " & modSTAR.nmSochi$ & ") AND " -
  & "(試験体 = " & modSTAR.nmTai$ & ") AND " -
  & "(試験部分 = " & modSTAR.nmBibun$ & ") AND " -
  & "(解析ケース = " & modSTAR.CurrentCase$ & ") AND " -
End If

'対象テームによる管理テームの検索SQLの作成
If TargetType <= 2 Then                                '1,2:解析,表計算
  SQL2$ = "(解析ケース LIKE 'ケース%') AND (評価法id = 0)"
Elseif TargetType = 3 And TargetSubType = 0 Then      '3:損傷値(全評価法)
  SQL2$ = "(評価法id > 0)"
Elseif TargetType = 3 And TargetSubType > 0 Then     '3:損傷値(指定評価法)
  SQL2$ = "(評価法id = " & GetHyokaID(Target$) & ")"
End If
SQL$ = SQL0$ & SQL1$ & SQL2$ & ") ORDER BY id;"

'リストボックスへ表示
Set dtKANRI = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
Set dtKANRI = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hKANRI)
If hKANRI = -1 Then GoTo Err1

If dtKANRI.EOF = False Then

  Do Until dtKANRI.EOF
    Buf$ = dtKANRI![試験装置] & "/" & dtKANRI![試験体] & "/"
    Buf$ = Buf$ & dtKANRI![試験部分] & "/" & dtKANRI![解析ケース]
    If TargetType = 3 And TargetSubType = 0 Then
      Buf$ = Buf$ & "/" & GetHyokaName(dtKANRI![評価法id])
    Else
      Buf$ = Buf$ & "/" & Target$
    End If
    lstTarget.AddItem Buf$
    dtKANRI.MoveNext
  Loop
  cmdOK.Enabled = True

Else

  lstTarget.AddItem "処理するデータがありません。"
  cmdOK.Enabled = False

End If

'dtKANRI.Close
xCloseResultset dtKANRI, hKANRI

Screen.MousePointer = 0

Exit Sub

Err1:
  Screen.MousePointer = 0
  MsgBox "Error in frmTokIntCtrl/Load"
  Call ShowErrorMessage
  On Error Resume Next
  'dtKANRI.Close
  xCloseResultset dtKANRI, hKANRI
  Unload Me                                '強制エラー

End Sub

Private Sub lblTarget_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
  If Button = 2 Then lblTarget.ShowWhatsThis
End Sub

Private Sub lstTarget_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
  If Button = 2 Then lstTarget.ShowWhatsThis
End Sub

```

```
Attribute VB_Name = "frmTokInterSel"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Option Explicit
```

```
Private Sub cmdCancel_Click()
    Unload Me
End Sub
```

```
Private Sub cmdCancel_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then cmdCancel.ShowWhatsThis
End Sub
```

```
Private Sub cmdOK_Click()
    frmTokIntCtrl.Show 1
    Unload Me
End Sub
```

```
Private Sub cmdOK_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then cmdOK.ShowWhatsThis
End Sub
```

```
Private Sub Form_Load()
    Dim dtHYOKA As rdoResultset
    Dim SQL$
    Dim hHYOKA As Integer

    On Error GoTo Err1

    Screen.MousePointer = 11
    cmbDmgSel.Enabled = False

    SQL$ = "SELECT 評価法名 FROM 評価法マスタ;"
    ' Set dtHYOKA = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
    Set dtHYOKA = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hHYOKA)
    If hHYOKA = -1 Then GoTo Err1

    cmbDmgSel.AddItem "全評価法"

    Do Until dtHYOKA.EOF
        cmbDmgSel.AddItem dtHYOKA![評価法名]
        dtHYOKA.MoveNext
    Loop

    cmbDmgSel.ListIndex = 0

    ' dtHYOKA.Close
    xCloseResultset dtHYOKA, hHYOKA

    Screen.MousePointer = 0

    Exit Sub
```

```
Err1:
    Screen.MousePointer = 0
    MsgBox "Error in frmTokInterSel/Load"
    Call ShowErrorMessage
    On Error Resume Next
    ' dtHYOKA.Close
    xCloseResultset dtHYOKA, hHYOKA
    Unload Me '強制終了
End Sub
```

```
Private Sub fraTarget_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then fraTarget.ShowWhatsThis
End Sub
```

```
Private Sub optAnal_Click()
    cmbDmgSel.Enabled = False
```

End Sub

Private Sub optAnal_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)

 If Button = 2 Then optAnal.ShowWhatsThis

End Sub

Private Sub optDamage_Click()

 cmbDmgSel.Enabled = True

End Sub

Private Sub optDamage_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)

 If Button = 2 Then optDamage.ShowWhatsThis

End Sub

Private Sub optTable_Click()

 cmbDmgSel.Enabled = False

End Sub

Private Sub optTable_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)

 If Button = 2 Then optTable.ShowWhatsThis

End Sub


```

Attribute VB_Name = "frmTokSelect"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Option Explicit

```

```
Private Sub cmbQuery_Click()
```

```
    If cmbQuery.ListIndex >= 0 Then
        cmdOK.Enabled = True
    End If

```

```
End Sub
```

```
Private Sub cmdCancel_Click()
```

```
    modSTAR.tmpFormCloseType = vbCancel
```

```
    Unload Me
```

```
End Sub
```

```
Private Sub cmdCancel_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
```

```
    If Button = 2 Then cmdCancel.ShowWhatsThis
```

```
End Sub
```

```
Private Sub cmdOK_Click()
```

```
    On Error GoTo Err1
```

```
    If cmbQuery.ListIndex < 0 Then
        MsgBox "検索方法が選択されていません。"
        Exit Sub
    End If

```

```
    modSTAR.CurTokQueryNo = cmbQuery.ListIndex
    modSTAR.CurTokQueryName$ = cmbQuery.List(cmbQuery.ListIndex)

```

```
    modSTAR.tmpFormCloseType = vbOK
```

```
    Unload Me
```

```
    Exit Sub

```

```
Err1:
```

```
    MsgBox "Error in frmTokSelect/cmdOK"
    Call ShowErrorMessage
    modSTAR.tmpFormCloseType = vbCancel

```

```
End Sub
```

```
Private Sub cmdOK_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
```

```
    If Button = 2 Then cmdOK.ShowWhatsThis
```

```
End Sub
```

```
Private Sub cmdQDBChange_Click()
```

```
    Dim dbQuery As Database
    Dim i As Long
    Dim LCount As Long
    Dim Buf$

```

```
    CommonDialog1.Filter = "Access (*.MDB)*.mdb"
    CommonDialog1.Flags = cd1OFNReadOnly

```

```
    On Error Resume Next
    CommonDialog1.ShowOpen
    On Error GoTo 0

```

```
    '復帰ボタンのチェック
```

```
    If Err.Number <> 0 Then
        If Err.Number = 32755 Then '1-サがキャンセルを選択
            Exit Sub
        Else

```

```
            MsgBox "[ファイルを開く]タイルボックスで予期しないエラーが発生しました。"
            Exit Sub
        End If
    End If

```

```
End Sub
```

```

If CommonDialog1.FileName <> "" Then
    '入力ファイル名の取得
    txtQueryDB.Text = CommonDialog1.FileName
    modSTAR.QueryDB$ = CommonDialog1.FileName

    'cmbQueryリストボックスの更新
    LCount = cmbQuery.ListCount
    For i = LCount - 1 To 0 Step -1
        cmbQuery.RemoveItem i
    Next i

    Set dbQuery = Workspaces(0).OpenDatabase(modSTAR.QueryDB$)
    For i = 0 To dbQuery.QueryDefs.Count - 1
        Buf$ = dbQuery.QueryDefs(i).Name
        If Left$(Buf$, 2) = "統計" Then
            cmbQuery.AddItem dbQuery.QueryDefs(i).Name
        End If
    Next i
    dbQuery.Close

End If
End Sub

Private Sub cmdQDBChange_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then cmdQDBChange.ShowWhatsThis
End Sub

Private Sub Form_Load()
    Dim dbQuery As Database
    Dim i As Long
    Dim QName$, Buf$

    On Error GoTo Err1

    QName$ = modSTAR.QueryDB$
    txtQueryDB.Text = QName$

    If QName$ <> modSTAR.StarDB$ Then
        Set dbQuery = Workspaces(0).OpenDatabase(QName$)
    Else
        Set dbQuery = dbSTAR
    End If

    For i = 0 To dbQuery.QueryDefs.Count - 1
        Buf$ = dbQuery.QueryDefs(i).Name
        If Left$(Buf$, 2) = "統計" Then
            cmbQuery.AddItem dbQuery.QueryDefs(i).Name
        End If
    Next i

    If QName$ <> modSTAR.StarDB$ Then dbQuery.Close
    cmbQuery.ListIndex = modSTAR.CurTokQueryNo

    If cmbQuery.ListIndex < 0 Then
        cmdOK.Enabled = False
    End If

    modSTAR.tmpFormCloseType = vbCancel

Exit Sub

Err1:
    MsgBox "Error in frmTokSelect/Load"
    Call ShowErrorMessage
    modSTAR.tmpFormCloseType = vbCancel
    Unload Me '強制終了
End Sub

Private Sub fraFrom_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then fraFrom.ShowWhatsThis
End Sub

Private Sub fraQuery_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then fraQuery.ShowWhatsThis
End Sub

```

```
Private Sub fraQueryDB_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then fraQueryDB.ShowWhatsThis
End Sub
```

```
Private Sub txtQueryDB_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then txtQueryDB.ShowWhatsThis
End Sub
```

```

Attribute VB_Name = "frmUtilCaseAdd"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Option Explicit

Private Sub cmdCancel_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then cmdCancel.ShowWhatsThis
End Sub

Private Sub cmdOK_Click()
    Dim dtCASE As rdoResultset
    Dim SQL$, Buf$
    Dim IFERR As Integer
    Dim hCASE As Integer

    On Error GoTo Err1

    '解析ケースが重複していないかをチェックする必要がある！
    '-----
    Screen.MousePointer = 11

    'バックアップテーブルの作成
    Call CopyTable("解析ケーステーブル", "解析ケーステーブルBACKUP", IFERR)
    If IFERR = 1 Then GoTo Err1

    On Error GoTo Err2

    SQL$ = "SELECT * FROM 解析ケーステーブル;"
    ' Set dtCASE = dbSTAR5.OpenResultset(SQL$, rdOpenDynamic, rdConcurLock)
    Set dtCASE = xOpenResultset(SQL$, rdOpenDynamic, rdConcurLock, hCASE)
    If hCASE = -1 Then GoTo Err2

    dtCASE.AddNew

    modSTAR.MaxCase = modSTAR.MaxCase + 1
    dtCASE![id] = modSTAR.MaxCase
    dtCASE![解析ケース] = "ケース" & modSTAR.MaxCase
    If optTempJkn.Value = True Then
        dtCASE![流体温度] = optTempJkn.Caption
    ElseIf optTempDgn.Value = True Then
        dtCASE![流体温度] = optTempDgn.Caption
    End If
    If optTrnsJkn.Value = True Then
        dtCASE![熱伝達係数] = optTrnsJkn.Caption
    ElseIf optTrnsDgn.Value = True Then
        dtCASE![熱伝達係数] = optTrnsDgn.Caption
    End If
    If optMeshDtl.Value = True Then
        dtCASE![メッシュテーブル] = optMeshDtl.Caption
    ElseIf optMeshDgn.Value = True Then
        dtCASE![メッシュテーブル] = optMeshDgn.Caption
    End If
    dtCASE![解析法] = txtMethod.Text

    'リストボックスの更新
    Buf$ = dtCASE![解析ケース] & "; " & dtCASE![流体温度]
    Buf$ = Buf$ & " " & dtCASE![熱伝達係数]
    Buf$ = Buf$ & " " & dtCASE![メッシュテーブル]
    Buf$ = Buf$ & " " & dtCASE![解析法]
    frmUtilCaseReg.lstCase.AddItem Buf$

    dtCASE.Update

    'dtCASE.Close
    xCloseResultset dtCASE, hCASE

    'バックアップしておいたテーブルを削除(処理が成功したので)
    Call DeleteTable("解析ケーステーブルBACKUP", IFERR)

    Screen.MousePointer = 0

    Unload Me

    Exit Sub

Err1:
    Screen.MousePointer = 0
    MsgBox "Error in frmUtilCaseAdd/cmdOK 1"
    Call ShowErrorMessage
    Exit Sub

Err2:
    Screen.MousePointer = 0
    MsgBox "Error in frmUtilCaseAdd/cmdOK 2"

```

```

Call ShowErrorMessage
'dtCASE.Close
xCloseResultset dtCASE, hCASE
Call CopyTable("解析ケース*BACKUP", "解析ケース*", IFERR)
Call DeleteTable("解析ケース*BACKUP", IFERR)
frmUtilCaseReg.Form_Load 'リストボックスの更新のため

End Sub

Private Sub cmdCancel_Click()
Unload Me
End Sub

Private Sub cmdOK_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
If Button = 2 Then cmdOK.ShowWhatsThis
End Sub

Private Sub Form_Load()
Dim dtCASE As rdoResultset
Dim SQL$, Buf$
Dim hCASE As Integer

On Error GoTo Err1

Screen.MousePointer = 11

SQL$ = "SELECT * FROM 解析ケース*";
Set dtCASE = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
Set dtCASE = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hCASE)
If hCASE = -1 Then GoTo Err1

Do Until dtCASE.EOF
Buf$ = dtCASE![解析法]
lstMethod.AddItem Buf$
dtCASE.MoveNext
Loop

'dtCASE.Close
xCloseResultset dtCASE, hCASE

lblMessage.Caption = "ケース" & (modSTAR.MaxCase + 1) & " を追加します。"
optTempJkn = True
optTrnsJkn = True
optMeshDtl = True
txtMethod.Text = ""

Screen.MousePointer = 0

Exit Sub

Err1:
Screen.MousePointer = 0
MsgBox "Error in frmUtilCaseAdd/Load"
Call ShowErrorMessage
On Error Resume Next
'dtCASE.Close
xCloseResultset dtCASE, hCASE
Unload Me '強制エラー

End Sub

Private Sub fraMesh_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
If Button = 2 Then fraMesh.ShowWhatsThis
End Sub

Private Sub fraTemp_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
If Button = 2 Then fraTemp.ShowWhatsThis
End Sub

Private Sub fraTrns_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
If Button = 2 Then fraTrns.ShowWhatsThis
End Sub

Private Sub lblMessage_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)

```

```
    If Button = 2 Then lblMessage.ShowWhatsThis
End Sub

Private Sub lblMethod_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblMethod.ShowWhatsThis
End Sub

Private Sub lstMethod_Click()
    txtMethod.Text = lstMethod.List(lstMethod.ListIndex)
End Sub

Private Sub lstMethod_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lstMethod.ShowWhatsThis
End Sub

Private Sub optMeshDgn_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then optMeshDgn.ShowWhatsThis
End Sub

Private Sub optMeshDtl_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then optMeshDtl.ShowWhatsThis
End Sub

Private Sub optTempDgn_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then optTempDgn.ShowWhatsThis
End Sub

Private Sub optTempJkn_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then optTempJkn.ShowWhatsThis
End Sub

Private Sub optTrnsDgn_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then optTrnsDgn.ShowWhatsThis
End Sub

Private Sub optTrnsJkn_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then optTrnsJkn.ShowWhatsThis
End Sub

Private Sub txtMethod_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then txtMethod.ShowWhatsThis
End Sub
```

```
Attribute VB_Name = "frmUtilCaseReg"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Option Explicit
```

```
Private Sub cmdAdd_Click()
    frmUtilCaseAdd.Show 1
End Sub
```

```
Private Sub cmdAdd_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then cmdAdd.ShowWhatsThis
End Sub
```

```
Private Sub cmdDelete_Click()
    Dim dtCASE As rdoResultset
    Dim dtHYOKA As rdoResultset
    Dim Ret As Long
    Dim CName$, LName$, SQL$, SQL1$, SQL2$
    Dim P$
    Dim IFERR As Integer
    Dim hCASE As Integer
    Dim hHYOKA As Integer

    On Error GoTo Err1

    If lstCase.ListIndex < 0 Then
        MsgBox "解析ケースが選択されていません。"
        Exit Sub
    End If

    LName$ = lstCase.List(lstCase.ListIndex)
    Ret = MsgBox(LName$ & " を削除しますか?", vbYesNo)

    CName$ = Left$(LName$, InStr(1, LName$, ":")) - 1)

    If Ret <> 6 Then Exit Sub
```

```
Screen.MousePointer = 11
```

```
'バックアップテーブルの作成
Call CopyTable("解析ケーステーブル", "解析ケーステーブルBACKUP", IFERR)
If IFERR = 1 Then GoTo Err1
```

```
On Error GoTo Err2
```

```
'解析ケーステーブルの更新
SQL$ = "DELETE 解析ケーステーブル WHERE 解析ケース = '" & CName$ & "';"
Call DeleteDataSQL(SQL$)
dbSTAR5.Execute SQL$
```

```
'最大の解析ケース番号の取得
SQL$ = "SELECT 解析ケース FROM 解析ケーステーブル ORDER BY 解析ケース;"
Set dtCASE = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
Set dtCASE = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hCASE)
If hCASE = -1 Then GoTo Err2
```

```
dtCASE.MoveLast
modSTAR.MaxCase = Val(Mid(dtCASE![解析ケース], 4))
```

```
'dtCASE.Close
xCloseResultset dtCASE, hCASE
```

```
'管理テーブルからその解析ケースのレコードを削除
SQL$ = "DELETE 管理テーブル WHERE 解析ケース = '" & CName$ & "';"
Call DeleteKanriSQL(SQL$)
dbSTAR5.Execute SQL$
Call KanriRead(IFERR)
If IFERR = 1 Then GoTo Err2
Call KanriSequence(IFERR)
If IFERR = 1 Then GoTo Err2
```

```
'各種テーブルからその解析ケースのレコードを削除
SQL1$ = "DELETE
SQL2$ = " WHERE 解析ケース = '" & CName$ & "';"
Call DeleteDataSQL(SQL1$ & "解析" & SQL2$)
Call DeleteDataSQL(SQL1$ & "解析_S" & SQL2$)
Call DeleteDataSQL(SQL1$ & "表計算" & SQL2$)
Call DeleteDataSQL(SQL1$ & "表計算_S" & SQL2$)
Call DeleteAnaIC("解析", "時点", CName$)
Call DeleteAnaIC("解析_S", "時点_S", CName$)
Call DeleteDataC("表計算", CName$)
Call DeleteDataC("表計算_S", CName$)
```

```

'P$ = Chr$(34) & "("
SQL$ = "SELECT 評価法名 FROM 評価法テーブル;"
'Set dtHYOKA = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
Set dtHYOKA = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hHYOKA)
If hHYOKA = -1 Then GoTo Err2

Do Until dtHYOKA.EOF
'SQL$ = SQL1$ & P$ & dtHYOKA![評価法名] & P$ & SQL2$
'Call DeleteDataSQL(SQL$)
'SQL$ = SQL1$ & P$ & dtHYOKA![評価法名] & "_S" & P$ & SQL2$
'Call DeleteDataSQL(SQL$)
Call DeleteDataC(dtHYOKA![評価法名], CName$)
Call DeleteDataC(dtHYOKA![評価法名] & "_S", CName$)
dtHYOKA.MoveNext
Loop

'dtHYOKA.Close
xCloseResultset dtHYOKA, hHYOKA

'リストボックスの更新
lstCase.RemoveItem lstCase.ListIndex
cmdDelete.Enabled = False

'バックアップしておいたテーブルを削除(処理が成功したので)
Call DeleteTable("解析テーブルBACKUP", IFERR)

Screen.MousePointer = 0

Exit Sub

Err1:
Screen.MousePointer = 0
MsgBox "Error in frmUtilCaseReg/cmdDelete 1"
Call ShowErrorMessage
Exit Sub

Err2:
Screen.MousePointer = 0
MsgBox "Error in frmUtilCaseReg/cmdDelete 2"
Call ShowErrorMessage
'dtCASE.Close
xCloseResultset dtCASE, hCASE
'dtHYOKA.Close
xCloseResultset dtHYOKA, hHYOKA
Call CopyTable("解析テーブルBACKUP", "解析テーブル", IFERR)
Call DeleteTable("解析テーブルBACKUP", IFERR)
Form_Load 'リストボックスの更新のため

End Sub

Private Sub cmdDelete_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
If Button = 2 Then cmdDelete.ShowWhatsThis
End Sub

Private Sub cmdExit_Click()
Unload Me
End Sub

Private Sub cmdExit_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
If Button = 2 Then cmdExit.ShowWhatsThis
End Sub

Public Sub Form_Load()
Dim dtCASE As rdoResultset
Dim SQL$, Buf$
Dim IFERR As Integer
Dim hCASE As Integer

On Error GoTo Err1

Screen.MousePointer = 11

'解析ケース名をリストボックスに表示
modSTAR.MaxCase = 0
SQL$ = "SELECT * FROM 解析テーブル ORDER BY 解析ケース;"
'Set dtCASE = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
Set dtCASE = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hCASE)
If hCASE = -1 Then GoTo Err1

lstCase.Clear
Do Until dtCASE.EOF
If Val(Mid(dtCASE![解析ケース], 4)) > modSTAR.MaxCase Then
modSTAR.MaxCase = Val(Mid(dtCASE![解析ケース], 4))

```



```

End If
Buf$ = dtCASE![解析ケース] & ". " & dtCASE![流体温度]
Buf$ = Buf$ & " " & dtCASE![熱伝達係数]
Buf$ = Buf$ & " " & dtCASE![メッシュ]
Buf$ = Buf$ & " " & dtCASE![解析法]
lstCase.AddItem Buf$
dtCASE.MoveNext
Loop

'dtCASE.Close
xCloseResultset dtCASE, hCASE

cmdDelete.Enabled = False

Screen.MousePointer = 0

Exit Sub

Err1:
Screen.MousePointer = 0
MsgBox "Error in frmUtilCaseReg/Load"
Call ShowErrorMessage
On Error Resume Next
'dtCASE.Close
xCloseResultset dtCASE, hCASE
Unload Me '強制終了-

End Sub

Private Sub lblCase_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblCase.ShowWhatsThis
End Sub

Private Sub lstCase_Click()
    cmdDelete.Enabled = True
End Sub

Private Sub lstCase_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lstCase.ShowWhatsThis
End Sub

```

```
Attribute VB_Name = "frmUtilHyokaAdd"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Option Explicit
```

```
Private Sub cmdBrowse_Click()
    Dim OpenFileDialogType As Long
```

```
    OpenFileDialogType = 2                '1:コンタ*イロク*
                                           '2:X_SELDB1*イロク*
```

```
    If OpenFileDialogType = 1 Then
        CommonDialog1.Filter = "Excel 5.0 (*.XLS)|*.xls"
        CommonDialog1.Flags = cd1OFNReadOnly
```

```
        On Error Resume Next
        CommonDialog1.ShowOpen
        On Error GoTo 0
```

```
        '復帰*タンのシキク
```

```
        If Err.Number <> 0 Then
```

```
            If Err.Number = 32755 Then 'ユ*ガ*キャンセルを選択
```

```
                Exit Sub
```

```
            Else
```

```
                MsgBox "[ファイルを開く]が*イロク*が*ツクスで予期しないエラーが発生しました。"
```

```
                Exit Sub
```

```
            End If
```

```
        End If
```

```
        '入力ファイル名の取得
```

```
        txtTemplate.Text = CommonDialog1.FileName
```

```
        txtRange.Text = "Database"
```

```
    ElseIf OpenFileDialogType = 2 Then
        modSTAR.tmpInpFilter = "Excel 8.0 (*.XLS)|CSVファイル (*.CSV)|"
        frmSelDB1.Show 1
```

```
        If modSTAR.tmpFormCloseType = vbCancel Then Exit Sub
```

```
        '入力ファイル名の取得
```

```
        txtTemplate.Text = modSTAR.tmpInpFName$(0)
```

```
        txtRange.Text = modSTAR.tmpInpTName$(0)
```

```
    End If
```

```
End Sub
```

```
Private Sub cmdBrowse_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
```

```
    If Button = 2 Then cmdBrowse.ShowWhatsThis
```

```
End Sub
```

```
Private Sub cmdCancel_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
```

```
    If Button = 2 Then cmdCancel.ShowWhatsThis
```

```
End Sub
```

```
Private Sub cmdOK_Click()
```

```
    Dim dtHYOKA As rdoResultset
```

```
    Dim dbEX_HYOKA As Database
```

```
    Dim dtEX_HYOKA As Recordset
```

```
    Dim Buf$, SQL$, PS
```

```
    Dim nmHyoka$, nmHyokaS$, nmTemplate$, nmRange$, nmConnect$
```

```
    Dim i As Long
```

```
    Dim j As Long
```

```
    Dim fld As Long
```

```
    Dim fld$
```

```
    Dim fldName$, Fields$, FName$
```

```
    Dim IsEOF As Boolean
```

```
    Dim IFERR As Integer
```

```
    Dim hHYOKA As Integer
```

```
    On Error GoTo Err1
```

```
    If txtName.Text = "" Then
```

```
        MsgBox "評価法の名称が設定されていません。"
```

```
        Exit Sub
```

```
    End If
```

```
    If txtBiko.Text = "" Then
```

```
        MsgBox "評価法の備考欄が入力されていません。"
```

```
        Exit Sub
```

```
    End If
```

```
    If txtID.Text = "" Then
```

```
        MsgBox "評価法のidが設定されていません。"
```

```

Exit Sub
End If
If txtTemplate.Text = "" Then
MsgBox "評価法のテンプレートのシートが設定されていません。"
Exit Sub
End If

' 評価法名の重複がないかをチェックする必要がある！
Screen.MousePointer = 11
SQL$ = "SELECT 評価法名 FROM 評価法テーブル"
      & "WHERE ((評価法名 = " & txtName.Text & ") OR "
      & " (id = " & Val(txtID.Text) & "));"
' Set dtHYOKA = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
Set dtHYOKA = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hHYOKA)
If hHYOKA = -1 Then GoTo Err1

IsEOF = dtHYOKA.EOF

' dtHYOKA.Close
xCloseResultset dtHYOKA, hHYOKA

Screen.MousePointer = 0
If IsEOF = False Then
MsgBox "その評価法またはidはすでに登録されています。"
Exit Sub
End If

-----

Screen.MousePointer = 11

' バックアップテーブルの作成
Call CopyTable("評価法テーブル", "評価法テーブルBACKUP", IFERR)
If IFERR = 1 Then GoTo Err1

On Error GoTo Err2

nmHyoka$ = txtName.Text
nmHyoka$$ = nmHyoka$ & "_S"
Buf$ = txtTemplate.Text
If StrComp(Right$(Buf$, 3), "XLS", 1) = 0 Then ' Excelシート
nmConnect$ = "Excel 8.0;"
nmTemplate$ = txtTemplate.Text
nmRange$ = txtRange.Text
ElseIf StrComp(Right$(Buf$, 3), "CSV", 1) = 0 Then ' CSVファイル
i = InStrRev(Buf$, "#") ' #の位置
j = Len(Buf$) - i - 4 ' 拡張子を除くファイル名の長さ
nmConnect$ = "Text;"
nmTemplate$ = Left$(Buf$, i)
nmRange$ = Mid$(Buf$, i + 1, j) & "#CSV"
End If

' テンプレートシートをオフン
Set dbEX_HYOKA = Workspaces(0).OpenDatabase(nmTemplate$, False, True, nmConnect$)
Set dtEX_HYOKA = dbEX_HYOKA.OpenRecordset(nmRange$, dbOpenSnapshot)

' 入力ファイルのファイル名チェック
Call InputFieldCheck(90, dtEX_HYOKA, IFERR)
If IFERR > 0 Then GoTo Err2

' 評価法テーブルの更新
SQL$ = "SELECT * FROM 評価法テーブル;"
' Set dtHYOKA = dbSTAR5.OpenResultset(SQL$, rdOpenDynamic, rdConcurLock)
Set dtHYOKA = xOpenResultset(SQL$, rdOpenDynamic, rdConcurLock, hHYOKA)
If hHYOKA = -1 Then GoTo Err2

dtHYOKA.AddNew
dtHYOKA![id] = txtID.Text
dtHYOKA![評価法名] = nmHyoka$
dtHYOKA![計算実施] = True
dtHYOKA![備考] = txtBiko.Text
For i = 4 To dtHYOKA.rdcColumns.Count - 1
dtHYOKA.rdcColumns(i).Value = -1 ' クリア
Next i
For i = 0 To dtEX_HYOKA.Fields.Count - 1
ifld = dtEX_HYOKA.Fields(i).Value
fld$ = dtEX_HYOKA.Fields(i).Name
If ifld >= 0 Then
dtHYOKA.rdcColumns(fld$).Value = ifld
End If
Next i
dtHYOKA.Update

' dtHYOKA.Close
xCloseResultset dtHYOKA, hHYOKA

' リストボックスを更新
frmUtilHyokaReg.lstHyoka.AddItem txtID.Text & " " & nmHyoka$
frmUtilHyokaReg.Form_Load

' テンプレートシートの登録
-----

' STAR5-から既存のテンプレートを削除しておく
Call DeleteTable(nmHyoka$, IFERR)
Call DeleteTable(nmHyoka$$, IFERR)

```

```

'テンプレートのフィールド名を定義
'PS = Chr$(34) & "("
'FName$ = modSTAR.WorkDir & "FNAME.TXT"
'Open FName$ For Output Access Write As #1
Fields$ = "(id INT,参照id INT,解析id INT,試験条件id INT" _
& ",解析ケース VARCHAR(50),ライン番号 INT,L FLOAT" _
& ",起点r FLOAT,起点z FLOAT,起点θ FLOAT"
'Print #1, Fields$;
For i = 0 To dtEX_HYOKA.Fields.Count - 1
fldName$ = BH2US(dtEX_HYOKA.Fields(i).Name) 'アランク/イアツはアツク-スコアに
If fldName$ <> "id" And _
fldName$ <> "参照id" And _
fldName$ <> "解析id" And _
fldName$ <> "試験条件id" And _
fldName$ <> "解析ケース" And _
fldName$ <> "ライン番号" And _
fldName$ <> "L" And _
fldName$ <> "起点r" And _
fldName$ <> "起点z" And _
fldName$ <> "起点θ" Then
Fields$ = Fields$ & "," & fldName$ & " FLOAT"
'Print #1, "," & fldName$ & " FLOAT";
End If
Next i
Fields$ = Fields$ & ")"
'Print #1, ")"
'Close #1

'Open FName$ For Input Access Read As #1
'Line Input #1, Fields$
'Close #1

'STARデータベースの中にテンプレートのテーブルを作成
Call CreateTable(nmHyoka$, Fields$)
Call CreateTable(nmHyoka$$, Fields$)

'テンプレートシートをクローズ
dtEX_HYOKA.Close
dbEX_HYOKA.Close

'バックアップテーブルを削除しておく(処理が成功したので)
Call DeleteTable("評価法テーブルBACKUP", IFERR)

Screen.MousePointer = 0

Unload Me

Exit Sub

Err1:
Screen.MousePointer = 0
MsgBox "Error in frmUtilHyokaAdd/cmdOK 1"
Call ShowErrorMessage
xCloseResultset dtHYOKA, hHYOKA
Exit Sub

Err2:
Screen.MousePointer = 0
MsgBox "Error in frmUtilHyokaAdd/cmdOK 2"
Call ShowErrorMessage
dtHYOKA.Close
xCloseResultset dtHYOKA, hHYOKA
Call CopyTable("評価法テーブルBACKUP", "評価法テーブル", IFERR)
Call DeleteTable("評価法テーブルBACKUP", IFERR)
Call DeleteTable(nmHyoka$, IFERR)
Call DeleteTable(nmHyoka$$, IFERR)
frmUtilHyokaReg.Form_Load 'リストボックスの更新のため
On Error Resume Next
dtEX_HYOKA.Close
dbEX_HYOKA.Close

End Sub

Private Sub cmdCancel_Click()
Unload Me
End Sub

Private Sub cmdOK_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
If Button = 2 Then cmdOK.ShowWhatsThis
End Sub

Private Sub Form_Load()
txtName.Text = ""
txtBiko.Text = ""
'txtID.Text = GetMaxID("評価法テーブル", "id") + 1
txtID.Text = ""
txtTemplate.Text = ""
txtRange.Text = ""

```

End Sub

```
Private Sub fraTemplate_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then fraTemplate.ShowWhatsThis
End Sub
```

```
Private Sub lblBiko_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblBiko.ShowWhatsThis
End Sub
```

```
Private Sub lblID_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblID.ShowWhatsThis
End Sub
```

```
Private Sub lblName_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblName.ShowWhatsThis
End Sub
```

```
Private Sub lblRange_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblRange.ShowWhatsThis
End Sub
```

```
Private Sub lblTemplate_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblTemplate.ShowWhatsThis
End Sub
```

```
Private Sub txtBiko_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then txtBiko.ShowWhatsThis
End Sub
```

```
Private Sub txtID_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then txtID.ShowWhatsThis
End Sub
```

```
Private Sub txtName_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then txtName.ShowWhatsThis
End Sub
```

```
Private Sub txtRange_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then txtRange.ShowWhatsThis
End Sub
```

```
Private Sub txtTemplate_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then txtTemplate.ShowWhatsThis
End Sub
```

```
Attribute VB_Name = "frmUtilHyokaReg"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Option Explicit
```

```
Private Sub cmdAdd_Click()
    frmUtilHyokaAdd.Show 1
End Sub
```

```
Private Sub cmdAdd_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then cmdAdd.ShowWhatsThis
End Sub
```

```
Private Sub cmdDelete_Click()
    Dim Ret As Long
    Dim nmHyoka$, SQL$, tmp$
    Dim nmHID As Long
    Dim SP As Long
    Dim IFERR As Integer

    On Error GoTo Err1

    If lstHyoka.ListIndex < 0 Then
        MsgBox "損傷評価法が選択されていません。"
        Exit Sub
    End If

    tmp$ = lstHyoka.List(lstHyoka.ListIndex)
    SP = InStr(1, tmp, " ")
    nmHyoka$ = Mid$(tmp$, SP + 2) 'スペースが2つ入っている
    'nmHyoka$ = lstHyoka.List(lstHyoka.ListIndex)
    'nmHID = GetHyokaID(nmHyoka$)
    nmHID = lstHyoka.ItemData(lstHyoka.ListIndex)
    Debug.Print "nmHID: " & nmHID
    Ret = MsgBox("(" & nmHyoka$ & ")を削除しますか?", vbYesNo)

```

```
If Ret <> vbYes Then Exit Sub
```

```
Screen.MousePointer = 11
```

```
'バックアップテーブルの作成
Call CopyTable("評価法テーブル", "評価法テーブルBACKUP", IFERR)
If IFERR = 1 Then GoTo Err1
Call CopyTable(nmHyoka$, nmHyoka$ & "BACKUP", IFERR)
If IFERR = 1 Then GoTo Err1
Call CopyTable(nmHyoka$ & "_S", nmHyoka$ & "_SBACKUP", IFERR)
If IFERR = 1 Then GoTo Err1
```

```
On Error GoTo Err2
```

```
'評価法テーブルから指定された名前を削除
SQL$ = "DELETE 評価法テーブル WHERE "
    & "(評価法名 = " & nmHyoka$ & ");"
'Call DeleteDataSQL(SQL$)
dbSTAR5.Execute SQL$
```

```
'管理テーブルからこの評価法の行を削除
SQL$ = "DELETE 管理テーブル WHERE (評価法id= " & nmHID & ");"
'Call DeleteKanriSQL(SQL$)
dbSTAR5.Execute SQL$
Call KanriRead(IFERR)
If IFERR = 1 Then GoTo Err2
Call KanriSequence(IFERR)
If IFERR = 1 Then GoTo Err2
```

```
'損傷評価法の登録フォームから今削除した名前を消す
lstHyoka.RemoveItem lstHyoka.ListIndex
cmdDelete.Enabled = False
```

```
'損傷値テーブルを削除
Call DeleteTable(nmHyoka$, IFERR)
Call DeleteTable(nmHyoka$ & "_S", IFERR)
```

```
'バックアップテーブルを削除(処理が成功したので)
Call DeleteTable("評価法テーブルBACKUP", IFERR)
Call DeleteTable(nmHyoka$ & "BACKUP", IFERR)
Call DeleteTable(nmHyoka$ & "_SBACKUP", IFERR)
```

```
Screen.MousePointer = 0
```

```
Exit Sub
```

```
Err1:
    Screen.MousePointer = 0
```

```

MsgBox "Error in frmUtilHyokaReg/cmdDelete 1"
Call ShowErrorMessage
Exit Sub

Err2:
Screen.MousePointer = 0
MsgBox "Error in frmUtilHyokaReg/cmdDelete 2"
Call ShowErrorMessage
Call CopyTable("評価法テーブルBACKUP", "評価法テーブル", IFERR)
Call CopyTable(nmHyoka$ & "BACKUP", nmHyoka$, IFERR)
Call CopyTable(nmHyoka$ & "SBACKUP", nmHyoka$ & "_S", IFERR)
Call DeleteTable("評価法テーブルBACKUP", IFERR)
Call DeleteTable(nmHyoka$ & "BACKUP", IFERR)
Call DeleteTable(nmHyoka$ & "SBACKUP", IFERR)
Form_Load 'リストボックスの更新のため

End Sub

Private Sub cmdDelete_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then cmdDelete.ShowWhatsThis
End Sub

Private Sub cmdExit_Click()
    Unload Me
End Sub

Private Sub cmdExit_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then cmdExit.ShowWhatsThis
End Sub

Public Sub Form_Load()
    Dim dtHYOKA As rdoResultset
    Dim SQL$
    Dim i As Long
    Dim IFERR As Integer
    Dim hHYOKA As Integer

    On Error GoTo Err1

    Screen.MousePointer = 11

    ' 評価法名をリストボックスに表示
    SQL$ = "SELECT id, 評価法名 FROM 評価法テーブル _
            & " ORDER BY id;"
    ' Set dtHYOKA = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
    Set dtHYOKA = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hHYOKA)
    If hHYOKA = -1 Then GoTo Err1

    i = 0
    lstHyoka.Clear
    Do Until dtHYOKA.EOF
        lstHyoka.AddItem dtHYOKA![評価法名]
        lstHyoka.AddItem dtHYOKA![id] & " " & dtHYOKA![評価法名]
        lstHyoka.ItemData(i) = dtHYOKA![id]
        dtHYOKA.MoveNext
        i = i + 1
    Loop

    ' dtHYOKA.Close
    xCloseResultset dtHYOKA, hHYOKA

    cmdDelete.Enabled = False

    Screen.MousePointer = 0

    Exit Sub

Err1:
    Screen.MousePointer = 0
    MsgBox "Error in frmUtilHyokaReg/Load"
    Call ShowErrorMessage
    On Error Resume Next
    ' dtHYOKA.Close
    xCloseResultset dtHYOKA, hHYOKA
    Unload Me '強制終了

End Sub

Private Sub lblHyoka_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblHyoka.ShowWhatsThis

```

```
End Sub
```

```
Private Sub lstHyoka_Click()  
    cmdDelete.Enabled = True  
End Sub
```

```
Private Sub lstHyoka_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)  
    If Button = 2 Then lstHyoka.ShowWhatsThis  
End Sub
```

```
Public Sub RestoreHyokaTable()  
    Dim i As Long  
    Dim SQL$  
    '評価法テーブルの削除  
    For i = 0 To dbSTAR5.rdoTables.Count - 1  
        If dbSTAR5.rdoTables(i).Name = "評価法テーブル" Then  
            SQL$ = "DROP TABLE 評価法テーブル;"  
            dbSTAR5.Execute SQL$  
        Exit For  
    End If  
Next i  
    'テーブルのリスト  
    SQL$ = "SELECT * INTO 評価法テーブル FROM 評価法テーブルBACKUP;"  
    dbSTAR5.Execute SQL$  
End Sub
```



```
Attribute VB_Name = "frmUtilKanriReg2"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Option Explicit
```

```
Private Sub cmdAdd_Click()
```

```
Dim LIndex As Long
Dim FullName$
Dim i1 As Long
Dim i2 As Long
```

```
On Error GoTo Err1
```

```
LIndex = outBubun.ListIndex
```

```
If LIndex >= 0 Or outBubun.ListCount = 0 Then
If outBubun.ListCount = 0 Then '全く登録されていまいとき
modSTAR.typeKanriAdd = 1
FullName$ = ""
Else
modSTAR.typeKanriAdd = outBubun.Indent(LIndex)
Debug.Print "FullPath-1", modSTAR.typeKanriAdd
FullName$ = outBubun.FullPath(LIndex)
Debug.Print "FullPath-2"
End If
```

```
If modSTAR.typeKanriAdd = 1 Then
modSTAR.tmpSochi$ = ""
modSTAR.tmpTai$ = ""
modSTAR.tmpBubun$ = ""
ElseIf modSTAR.typeKanriAdd = 2 Then
i1 = InStr(1, FullName$, "¥", 0)
modSTAR.tmpSochi$ = Left(FullName$, i1 - 1) '試験装置名の取得
modSTAR.tmpTai$ = ""
modSTAR.tmpBubun$ = ""
ElseIf modSTAR.typeKanriAdd = 3 Then
i1 = InStr(1, FullName$, "¥", 0)
i2 = InStr(i1 + 1, FullName$, "¥", 0)
modSTAR.tmpSochi$ = Left(FullName$, i1 - 1) '試験装置名の取得
modSTAR.tmpTai$ = Mid(FullName$, i1 + 1, i2 - i1 - 1) '試験体名の取得
modSTAR.tmpBubun$ = ""
End If
```

```
frmUtilKanriAdd2.Show 1
```

```
Else
MsgBox "試験装置・試験体・試験部分を選択してください。"
```

```
End If
```

```
Exit Sub
```

```
Err1:
```

```
MsgBox "Error in frmUtilKanriReg2/cmdAdd"
Call ShowErrorMessage
```

```
End Sub
```

```
Private Sub cmdAdd_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
```

```
If Button = 2 Then cmdAdd.ShowWhatsThis
```

```
End Sub
```

```
Private Sub cmdDelete_Click()
```

```
Dim dtKANRI As rdoResultset
Dim dtHYOKA As rdoResultset
Dim LIndex As Long
Dim DelLevel As Long
Dim SQL$, SQL1$, SQL2$, SQL$, SQLW$
Dim nmS$, nmI$, nmB$, FullName$, DelName$
Dim i As Long
Dim i1 As Long
Dim i2 As Long
Dim IFERR As Integer
Dim hKANRI As Integer
Dim hHYOKA As Integer
```

```
On Error GoTo Err1
```

```
LIndex = outBubun.ListIndex
If LIndex < 0 Then
MsgBox "削除する対象が選択されていません。"
Exit Sub
End If
```

```
DelLevel = outBubun.Indent(LIndex)
```

```

FullName$ = outBubun.FullPath(LIndex)

SQL1$ = "SELECT 参照id FROM 管理テ-テ-フM WHERE ("
If DelLevel = 1 Then
    nm$$ = FullName$
    nmT$ = ""
    nmB$ = ""
    DelName$ = "試験装置:" & nm$$
    SQLW$ = "(試験装置 = " & nm$$ & ")"
ElseIf DelLevel = 2 Then
    it = InStr(1, FullName$, "#", 0)
    nm$$ = Left(FullName$, it - 1)
    nmT$ = Mid(FullName$, it + 1)
    nmB$ = ""
    DelName$ = "試験体:" & nmT$
    SQLW$ = "(試験装置 = " & nm$$ & " AND " &
    & "(試験体 = " & nmT$ & ")"
ElseIf DelLevel = 3 Then
    i1 = InStr(1, FullName$, "#", 0)
    i2 = InStr(i1 + 1, FullName$, "#", 0)
    nm$$ = Left(FullName$, i1 - 1)
    nmT$ = Mid(FullName$, i1 + 1, i2 - i1 - 1)
    nmB$ = Mid(FullName$, i2 + 1)
    DelName$ = "試験部分:" & nmB$
    SQLW$ = "(試験装置 = " & nm$$ & " AND " &
    & "(試験体 = " & nmT$ & " AND " &
    & "(試験部分 = " & nmB$ & ")"
End If
SQLK$ = SQL1$ & SQLW$ & " AND (解析ケース = '-');"
Debug.Print SQLK$

If MsgBox(DelName$ & " を削除しますか?", vbYesNo) = vbNo Then
    Exit Sub
End If

'-----
Screen.MousePointer = 11

'バックアップテ-フMの作成
Call CopyTable("管理テ-テ-フM", "管理テ-テ-フMBACKUP", IFERR)
If IFERR = 1 Then GoTo Err1

On Error GoTo Err2

'各種テ-テ-フMからテ-テ-フMを削除
SQL$ = "SELECT 評価法名 FROM 評価法テ-フM;"
Set dtHYOKA = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
Set dtHYOKA = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hHYOKA)
If hHYOKA = -1 Then GoTo Err2

' Set dtKANRI = dbSTAR5.OpenResultset(SQLK$, rdOpenStatic, rdConcurReadOnly)
Set dtKANRI = xOpenResultset(SQLK$, rdOpenStatic, rdConcurReadOnly, hKANRI)
If hKANRI = -1 Then GoTo Err2

' SQL1$ = "DELETE "
Do Until dtKANRI.EOF
    ' SQL2$ = " WHERE (参照id = " & dtKANRI![参照id] & ");"
    ' Call DeleteDataSQL(SQL1$ & "き裂" & SQL2$)
    ' Call DeleteDataSQL(SQL1$ & "き裂_S" & SQL2$)
    ' Call DeleteDataSQL(SQL1$ & "試験条件" & SQL2$)
    ' Call DeleteDataSQL(SQL1$ & "温度" & SQL2$)
    ' Call DeleteDataSQL(SQL1$ & "形状" & SQL2$)
    ' Call DeleteDataSQL(SQL1$ & "解析" & SQL2$)
    ' Call DeleteDataSQL(SQL1$ & "解析_S" & SQL2$)
    ' Call DeleteDataSQL(SQL1$ & "表計算" & SQL2$)
    ' Call DeleteDataSQL(SQL1$ & "表計算_S" & SQL2$)
    i = dtKANRI![参照id]
    Call DeleteDataR("き裂", i)
    Call DeleteDataR("き裂_S", i)
    Call DeleteDataR("試験条件", i)
    Call DeleteDataR("温度", i)
    Call DeleteDataR("形状", i)
    Call DeleteAnaIR("解析", "時点", i)
    Call DeleteAnaIR("解析_S", "時点_S", i)
    Call DeleteDataR("表計算", i)
    Call DeleteDataR("表計算_S", i)

    dtHYOKA.MoveFirst
    Do Until dtHYOKA.EOF
        ' SQL$ = SQL1$ & dtHYOKA![評価法名] & SQL2$
        ' Call DeleteDataSQL(SQL$)
        ' SQL$ = SQL1$ & dtHYOKA![評価法名] & "_S" & SQL2$
        ' Call DeleteDataSQL(SQL$)
        Call DeleteDataR(dtHYOKA![評価法名], i)
        Call DeleteDataR(dtHYOKA![評価法名] & "_S", i)
        dtHYOKA.MoveNext
    Loop

    dtKANRI.MoveNext
Loop

dtKANRI.Close
xCloseResultset dtKANRI, hKANRI
dtHYOKA.Close

```

```

xCloseResultset dtHYOKA, hHYOKA

'管理テーブルの更新
SQL$ = "DELETE 管理テーブル WHERE (" & SQL$ & ");"
'Call DeleteKanriSQL(SQL$)
dbSTAR5.Execute SQL$
Call KanriRead(IFERR)
If IFERR = 1 Then GoTo Err2
Call KanriSequence(IFERR)
If IFERR = 1 Then GoTo Err2

'[管理データの登録]フォームの更新
Lindex = outBubun.ListCount
For i = Lindex - 1 To 0 Step -1
    outBubun.RemoveItem i
Next i
Form_Load          'リストボックスの更新のため

'バックアップしておいたテーブルを削除(処理が成功したため)
Call DeleteTable("管理テーブルBACKUP", IFERR)

Screen.MousePointer = 0

Exit Sub

Err1:
Screen.MousePointer = 0
MsgBox "Error in frmUtilKanriReg2/cmdDelete 1"
Call ShowErrorMessage
Exit Sub

Err2:
Screen.MousePointer = 0
MsgBox "Error in frmUtilKanriReg2/cmdDelete 2"
Call ShowErrorMessage
'dtKANRI.Close
xCloseResultset dtKANRI, hKANRI
'dtHYOKA.Close
xCloseResultset dtHYOKA, hHYOKA
Call CopyTable("管理テーブルBACKUP", "管理テーブル", IFERR)
Call DeleteTable("管理テーブルBACKUP", IFERR)
Form_Load          'リストボックスの更新のため

End Sub

Private Sub cmdDelete_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then cmdDelete.ShowWhatsThis
End Sub

Private Sub cmdExit_Click()
    Unload Me
End Sub

Private Sub cmdExit_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then cmdExit.ShowWhatsThis
End Sub

Public Sub Form_Load()
    Dim dtTMP As rdoResultset
    Dim nameSochi$, nameTai$, nameBubun$
    Dim i As Long
    Dim Item As Long
    Dim SQL$
    Dim IFERR As Integer
    Dim hTMP As Integer

    On Error GoTo Err1

    Screen.MousePointer = 11

    'バックアップテーブルの作成
    'Call CopyTable("管理テーブル", "管理テーブルBACKUP", IFERR)
    'If IFERR = 1 Then GoTo Err1

    '試験装置名,試験体名,試験部分名をリストボックスに表示

    SQL$ = "SELECT * FROM 管理テーブル;"
    'Set dtTMP = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
    Set dtTMP = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hTMP)
    If hTMP = -1 Then GoTo Err1

    'If dtKANRI.RecordCount > 0 Then
        nameSochi$ = ""

```

```

nameTai$ = ""
nameBubun$ = ""
i = 0
Item = 0
outBubun.Clear          'アクトラインホックス内をクリア

Do Until dtTMP.EOF

    If dtTMP![試験装置] <> nameSochi$ Then
        nameSochi$ = dtTMP![試験装置]
        nameTai$ = ""
        nameBubun$ = ""
        outBubun.AddItem nameSochi$
        outBubun.Indent(i) = 1
        i = i + 1
    End If

    If dtTMP![試験体] <> nameTai$ Then
        nameTai$ = dtTMP![試験体]
        nameBubun$ = ""
        outBubun.AddItem nameTai$
        outBubun.Indent(i) = 2
        i = i + 1
    End If

    If dtTMP![試験部分] <> nameBubun$ Then
        nameBubun$ = dtTMP![試験部分]
        outBubun.AddItem nameBubun$
        outBubun.Indent(i) = 3
        outBubun.PictureType(i) = outLeaf
        outBubun.ItemData(i) = Item
        i = i + 1
    End If

    dtTMP.MoveNext
    Item = Item + 1

Loop

'End If

'dtTMP.Close
xCloseResultset dtTMP, hTMP

cmdDelete.Enabled = False

Screen.MousePointer = 0

Exit Sub

Err1:
Screen.MousePointer = 0
MsgBox "Error in frmUtilKanriReg2"
Call ShowErrorMessage
On Error Resume Next
'dtTMP.Close
xCloseResultset dtTMP, hTMP
Unload Me          '強制エラー

End Sub

Private Sub lblBubun_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)

    If Button = 2 Then lblBubun.ShowWhatsThis

End Sub

Private Sub outBubun_Click()
    cmdDelete.Enabled = True
End Sub

Private Sub outBubun_Collapse(ListIndex As Integer)
    outBubun.ListIndex = ListIndex
End Sub

Private Sub outBubun_Db1Click()
    Dim LIndex As Long

    LIndex = outBubun.ListIndex
    'MsgBox outBubun.ListIndex

    If outBubun.HasSubItems(LIndex) Then
        If outBubun.Expand(LIndex) = False Then
            outBubun.Expand(LIndex) = True
            outBubun.PictureType(LIndex) = outOpen
        Else
            outBubun.Expand(LIndex) = False
            outBubun.PictureType(LIndex) = outClosed
        End If
    End If

```

End If

End Sub

```
Private Sub outBubun_Expand(ListIndex As Integer)
    outBubun.ListIndex = ListIndex
End Sub
```

```
Private Sub outBubun_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then outBubun.ShowWhatsThis
End Sub
```

```
Attribute VB_Name = "frmUtilKanriAdd2"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Option Explicit
```

```
Private Sub cmdCancel_Click()
    Unload Me
End Sub
```

```
Private Sub cmdCancel_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then cmdCancel.ShowWhatsThis
End Sub
```

```
Private Sub cmdReg_Click()
    Dim dtKANRI As rdoResultset
    Dim InsPos As Long
    Dim MaxRec As Long
    Dim LIndex As Long
    Dim i As Long
    Dim Ret As Long
    Dim SQL$, SQL1$, SQL2$
    Dim IFERR As Integer
    Dim hKANRI As Integer
```

```
On Error GoTo Err1
```

```
'-----入力データのチェック-----
```

```
'重複チェックも行わないと！
If txtSochi.Text = "" Then
    MsgBox "試験装置名が設定されていません。"
    Exit Sub
End If
```

```
If txtTai.Text = "" Then
    MsgBox "試験体名が設定されていません。"
    Exit Sub
End If
```

```
If txtBubun.Text = "" Then
    MsgBox "試験部分名が設定されていません。"
    Exit Sub
End If
```

```
'i = Val(txtLine.Text)
i = Val(cmbLine.Text)
If i <> 1 And i <> 2 And i <> 3 Then
    'ライン数は2か3！
    MsgBox "評価ライン数が設定されていません。"
    Exit Sub
End If
```

```
'-----管理テーブルの更新-----
```

```
Screen.MousePointer = 11
```

```
'バックアップテーブルの作成
Call CopyTable("管理テーブル", "管理テーブルBACKUP", IFERR)
If IFERR = 1 Then GoTo Err1
```

```
On Error GoTo Err2
```

```
'新しいレコードの挿入位置を調べる
SQL1$ = "SELECT id FROM 管理テーブル"
```

```
Select Case modSTAR.typeKanriAdd
Case 2
    SQL2$ = " WHERE (試験装置 = '" & modSTAR.tmpSochi$ & "'" )"
Case 3
    SQL2$ = " WHERE ((試験装置 = '" & modSTAR.tmpSochi$ & "'" ) _
        & " AND (試験体 = '" & modSTAR.tmpTai$ & "'" ))"
End Select
SQL$ = SQL1$ & SQL2$ & ";"
```

```
'Set dtKANRI = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
Set dtKANRI = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hKANRI)
If hKANRI = -1 Then GoTo Err2
```

```
If dtKANRI.EOF = False Then
    dtKANRI.MoveLast
    InsPos = dtKANRI![id] + 1
Else
    InsPos = 0
End If
```

```
'dtKANRI.Close
```

```

xCloseResultset dtKANRI, hKANRI

'新しいレコードの挿入および移動
'Call modSTAR.KanriInsert2(InsPos)
'Call modSTAR.KanriInsert5(InsPos, IFERR)
'Call modSTAR.KanriInsert6(InsPos, IFERR)
If IFERR = 1 Then GoTo Err2

'SQL$ = "SELECT * FROM 管理データテーブル WHERE (id = " & InsPos & ");"
'Set dtKANRI = dbSTAR5.OpenResultset(SQL$, rdOpenDynamic, rdConcurLock)
'Set dtKANRI = xOpenResultset(SQL$, rdOpenDynamic, rdConcurLock, hKANRI)
If hKANRI = -1 Then GoTo Err2

'dtKANRI.Edit
'dtKANRI![試験装置] = txtSochi.Text
'dtKANRI![試験体] = txtTai.Text
'dtKANRI![試験部分] = txtBubun.Text
'dtKANRI![評価件数] = cmbLine.Text
'dtKANRI![参照id] = txtID.Text
'dtKANRI![解析ケース] = "-"
'dtKANRI![評価法id] = 0
'dtKANRI![き裂id] = 0
'dtKANRI![試験条件id] = 0
'dtKANRI![温度id] = 0
'dtKANRI![形状id] = 0
'dtKANRI![解析id] = 0
'dtKANRI![表計算id] = 0
'dtKANRI![損傷値id] = 0
'dtKANRI![き裂Sid] = 0
'dtKANRI![解析Sid] = 0
'dtKANRI![表計算Sid] = 0
'dtKANRI![損傷値Sid] = 0
'dtKANRI.Update

''dtKANRI.Close
'xCloseResultset dtKANRI, hKANRI

KanriI(InsPos, 1) = txtSochi.Text '試験装置
KanriI(InsPos, 2) = txtTai.Text '試験体
KanriI(InsPos, 3) = txtBubun.Text '試験部分
KanriI(InsPos, 4) = cmbLine.Text '評価件数
KanriI(InsPos, 5) = txtID.Text '参照id
KanriI(InsPos, 6) = "-" '解析ケース
KanriI(InsPos, 7) = 0 '評価法id
KanriI(InsPos, 8) = 0 'き裂id
KanriI(InsPos, 9) = 0 '試験条件id
KanriI(InsPos, 10) = 0 '温度id
KanriI(InsPos, 11) = 0 '形状id
KanriI(InsPos, 12) = 0 '解析id
KanriI(InsPos, 13) = 0 '表計算id
KanriI(InsPos, 14) = 0 '損傷値id
KanriI(InsPos, 15) = 0 'き裂Sid
KanriI(InsPos, 16) = 0 '解析Sid
KanriI(InsPos, 17) = 0 '表計算Sid
KanriI(InsPos, 18) = 0 '損傷値Sid

Call KanriWrite(IFERR)
If IFERR = 1 Then GoTo Err2

'-----「管理データの登録」フォームの更新-----

'アウトラインボックスに新しいアイテムを追加
LIndex = frmUtilKanriReg2.outBubun.ListCount
For i = LIndex - 1 To 0 Step -1
    frmUtilKanriReg2.outBubun.RemoveItem i
Next i
frmUtilKanriReg2.Form_Load 'リストボックスの更新のため

'バックアップしておいたテーブルを削除(処理が成功したので)
Call DeleteTable("管理データテーブルBACKUP", IFERR)

Screen.MousePointer = 0

Unload Me

Exit Sub

Err1:
Screen.MousePointer = 0
MsgBox "Error in frmUtilKanriAdd2/cmdReg"
Call ShowErrorMessage
Exit Sub

Err2:
Screen.MousePointer = 0
MsgBox "Error in frmUtilKanriAdd2/cmdReg"
Call ShowErrorMessage
'dtKANRI.Close
xCloseResultset dtKANRI, hKANRI
Call CopyTable("管理データテーブルBACKUP", "管理データテーブル", IFERR)
Call DeleteTable("管理データテーブルBACKUP", IFERR)
frmUtilKanriReg2.Form_Load 'リストボックスの更新のため

End Sub

```

```

Private Sub cmdReg_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then cmdReg.ShowWhatsThis
End Sub

Private Sub Form_Load()
    On Error GoTo Err1

    Screen.MousePointer = 11

    cmbLine.AddItem "1"
    cmbLine.AddItem "2"
    cmbLine.AddItem "3"

    txtSochi.Text = ""
    txtTai.Text = ""
    txtBubun.Text = ""
    cmbLine.Text = ""
    txtID.Text = GetMaxID("管理マ-ターナル", "参照id") + 1 '最大IDの取得

    If modSTAR.typeKanriAdd = 1 Then
        Caption = "試験装置の追加"
        lblMessage.Caption = "試験装置を新規に追加します。"
    ElseIf modSTAR.typeKanriAdd = 2 Then
        Caption = "試験体の追加"
        lblMessage.Caption = modSTAR.tmpSochi$ & "に試験体を新規に追加します。"

        txtSochi.Text = modSTAR.tmpSochi$
        txtSochi.Enabled = False
    ElseIf modSTAR.typeKanriAdd = 3 Then
        Caption = "試験部分の追加"
        lblMessage.Caption = modSTAR.tmpTai$ & "に試験部分を新規に追加します。"

        txtSochi.Text = modSTAR.tmpSochi$
        txtTai.Text = modSTAR.tmpTai$
        txtSochi.Enabled = False
        txtTai.Enabled = False
    End If

    Screen.MousePointer = 0

Exit Sub

Err1:
    Screen.MousePointer = 0
    MsgBox "Error in frmUtilKanriAdd2/Load"
    Call ShowErrorMessage
    Unload Me '強制エラー
End Sub

Private Sub lblBubun_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblBubun.ShowWhatsThis
End Sub

Private Sub lblID_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblID.ShowWhatsThis
End Sub

Private Sub lblLine_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblLine.ShowWhatsThis
End Sub

Private Sub lblMessage_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblMessage.ShowWhatsThis
End Sub

Private Sub lblSochi_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblSochi.ShowWhatsThis
End Sub

```



```
Private Sub lblTai_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblTai.ShowWhatsThis
End Sub
```

```
Private Sub txtBubun_Change()
    txtRelation.Text = txtSochi.Text & "_" & txtTai.Text & "_" & txtBubun.Text
End Sub
```

```
Private Sub txtBubun_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then txtBubun.ShowWhatsThis
End Sub
```

```
Private Sub txtID_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then txtID.ShowWhatsThis
End Sub
```

```
Private Sub txtSochi_Change()
    txtRelation.Text = txtSochi.Text & "_" & txtTai.Text & "_" & txtBubun.Text
End Sub
```

```
Private Sub txtSochi_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then txtSochi.ShowWhatsThis
End Sub
```

```
Private Sub txtTai_Change()
    txtRelation.Text = txtSochi.Text & "_" & txtTai.Text & "_" & txtBubun.Text
End Sub
```

```
Private Sub txtTai_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then txtTai.ShowWhatsThis
End Sub
```

```
Attribute VB_Name = "frmSecAnalReg"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Option Explicit
```

```
Private Sub cmbFileType_Click()
    Dim tmp As String

    tmp = Mid(cmbFileType.Text, InStr(cmbFileType.Text, "(") + 1)
    fillLine1.Pattern = Left(tmp, Len(tmp) - 1)
    fillLine2.Pattern = Left(tmp, Len(tmp) - 1)
    fillLine3.Pattern = Left(tmp, Len(tmp) - 1)
End Sub
```

```
Private Sub cmdCancel_Click()
    modSTAR.tmpFormCloseType = vbCancel

    Unload Me
End Sub
```

```
Private Sub cmdOK_Click()
    Dim dtKANRI As Recordset
    Dim FName$(3)

    '入力チェック
    If modSTAR.NoOfLine >= 1 Then
        If txtLine1.Text = "" Then
            MsgBox "評価ライン1の入力ファイル名が指定されていません!"
            Exit Sub
        End If
    End If
    If modSTAR.NoOfLine >= 2 Then
        If txtLine2.Text = "" Then
            MsgBox "評価ライン2の入力ファイル名が指定されていません!"
            Exit Sub
        End If
    End If
    If modSTAR.NoOfLine >= 3 Then
        If txtLine3.Text = "" Then
            MsgBox "評価ライン3の入力ファイル名が指定されていません!"
            Exit Sub
        End If
    End If
```

```
If Right(dirDirectory.Path, 1) <> "*" Then
    FName$(0) = dirDirectory.Path & "*" & txtLine1.Text
    FName$(1) = dirDirectory.Path & "*" & txtLine2.Text
    FName$(2) = dirDirectory.Path & "*" & txtLine3.Text
Else
    FName$(0) = dirDirectory.Path & txtLine1.Text
    FName$(1) = dirDirectory.Path & txtLine2.Text
    FName$(2) = dirDirectory.Path & txtLine3.Text
End If
```

```
If modSTAR.NoOfLine < 3 Then
    FName$(2) = FName$(0) 'NULだと他のM-テンでエラーになる為
End If
```

```
modSTAR.tmpInpFName$(0) = FName$(0)
modSTAR.tmpInpFName$(1) = FName$(1)
modSTAR.tmpInpFName$(2) = FName$(2)
```

```
modSTAR.tmpFormCloseType = vbOK
```

```
Unload Me
End Sub
```

```
Private Sub dirDirectory_Change()
    fillLine1.Path = dirDirectory.Path
    fillLine2.Path = dirDirectory.Path
    fillLine3.Path = dirDirectory.Path
End Sub
```

```
Private Sub drvDrive_Change()
    Dim oldPath As String

    oldPath = dirDirectory.Path
    On Error GoTo trapDrv:
    dirDirectory.Path = drvDrive.Drive
    Exit Sub
```

```
trapDrv:
    Beep
    MsgBox "ドライブの準備ができていません!", 48, "エラー"
```

```
dirDirectory.Path = oldPath
drvDrive.Drive = Left(oldPath, 2)
End Sub
```

```
Private Sub filLine1_Click()
txtLine1.Text = filLine1.FileName
End Sub
```

```
Private Sub filLine2_Click()
txtLine2.Text = filLine2.FileName
End Sub
```

```
Private Sub filLine3_Click()
txtLine3.Text = filLine3.FileName
End Sub
```

```
Private Sub Form_Load()
txtLine1.Text = ""
txtLine2.Text = ""
txtLine3.Text = ""

cmbFileType.AddItem "標準テキスト (*.STF)"
cmbFileType.AddItem "Excel 8.0 (*.XLS)"
cmbFileType.AddItem "CSVファイル (*.CSV)"
cmbFileType.ListIndex = 0

modSTAR.tmpFormCloseType = vbCancel
End Sub
```

```

Attribute VB_Name = "frmPriViewOLE"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Option Explicit

```

```

Private Sub Form_Load()
    mnuViewSave.Enabled = modSTAR.SuperVisor
End Sub

```

```

Private Sub Form_Resize()
    frmStarMain.ActiveForm.ScaleMode = 2 'ポイント
    oleView.Move 25, 17, Me.ScaleWidth - 25, Me.ScaleHeight - 17
End Sub

```

```

Private Sub mnuViewClose_Click()
    Unload Me
End Sub

```

```

Private Sub mnuViewSave_Click()

```

```

    ' このボタンは使用されない。

```

```

    Dim dtTMP As Recordset
    Dim objOLE As Object
    Dim SQL$
    Dim i As Long
    Dim j As Long
    Dim Rows As Long
    Dim Cols As Long

```

```

    ' 行数・桁数の取得

```

```

    Rows = modSTAR.tmpViewTableRow
    Cols = modSTAR.tmpViewTableCol

```

```

    ' 古いデータを削除しておく

```

```

    SQL$ = "DELETE" & Mid$(modSTAR.tmpViewSQL$, 7) & ";"
    dbSTAR.Execute SQL$, dbDenyWrite

```

```

    Set objOLE = oleView.object

```

```

    Set dtTMP = dbSTAR.OpenRecordset(modSTAR.tmpViewTableName$, dbOpenDynaset)

```

```

    For i = 0 To modSTAR.tmpViewTableRow - 1

```

```

        Debug.Print i

```

```

        dtTMP.AddNew

```

```

        For j = 0 To modSTAR.tmpViewTableCol - 1

```

```

            dtTMP.Fields(j).Value = objOLE.Cells(i + 2, j + 1).Value

```

```

        Next j

```

```

        dtTMP.Update

```

```

    Next i

```

```

    dtTMP.Close

```

```

    Set objOLE = Nothing

```

```

End Sub

```

```
Attribute VB_Name = "frmUtilMatReg"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
```

```
Private Sub cmdAdd_Click()
```

```
Screen.MousePointer = 11
```

```
lblNewMat.Enabled = True
txtNewMat.Enabled = True
lblNewBiko.Enabled = True
txtNewBiko.Enabled = True
lblNewID.Enabled = True
txtNewID.Text = GetMaxID("材料マスタ", "id") + 1
cmdReg.Enabled = True
cmdCancel.Enabled = True
cmdAdd.Enabled = False
cmdDelete.Enabled = False
cmdExit.Enabled = False
```

```
Screen.MousePointer = 0
```

```
End Sub
```

```
Private Sub cmdAdd_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
```

```
If Button = 2 Then cmdAdd.ShowWhatsThis
```

```
End Sub
```

```
Private Sub cmdCancel_Click()
```

```
txtNewMat.Text = ""
txtNewBiko.Text = ""
txtNewID.Text = ""
```

```
lblNewMat.Enabled = False
txtNewMat.Enabled = False
lblNewBiko.Enabled = False
txtNewBiko.Enabled = False
lblNewID.Enabled = False
txtNewID.Enabled = False
cmdReg.Enabled = False
cmdCancel.Enabled = True
cmdAdd.Enabled = True
cmdDelete.Enabled = True
cmdExit.Enabled = True
```

```
End Sub
```

```
Private Sub cmdCancel_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
```

```
If Button = 2 Then cmdCancel.ShowWhatsThis
```

```
End Sub
```

```
Private Sub cmdDelete_Click()
```

```
Dim LI As Long
Dim Ret As Long
Dim MName$, SQL$
Dim IFERR As Integer
```

```
On Error GoTo Err1
```

```
' 選択された材料名の取得
LI = lstMat.ListIndex
MName$ = lstMat.List(LI)
```

```
Ret = MsgBox(MName$ & "を参照している試験条件マスタがあるかもしれません。削除しますか?", vbYesNo, "STAR")
```

```
If Ret <> vbYes Then Exit Sub
```

```
Screen.MousePointer = 11
```

```
'バックアップマスタの作成
Call CopyTable("材料マスタ", "材料マスタBACKUP", IFERR)
If IFERR = 1 Then GoTo Err1
```

```
On Error GoTo Err2
```

```
'マスタからその材料名を削除
SQL$ = "DELETE 材料マスタ WHERE 材料名 = '" & MName$ & "'" ;
Debug.Print SQL$
dbSTAR5.Execute SQL$
```

```

'リストボックス他の更新
lstMat.RemoveItem LI
lstMat.ListIndex = -1
cmdDelete.Enabled = False

'バックアップしておいたテーブルを削除(処理が終了したので)
Call DeleteTable("材料テーブルBACKUP", IFERR)

Screen.MousePointer = 0

Exit Sub

Err1:
Screen.MousePointer = 0
MsgBox "Error in frmUtilMatReg/cmdDelete 1"
Call ShowErrorMessage
Exit Sub

Err2:
Screen.MousePointer = 0
MsgBox "Error in frmUtilMatReg/cmdDelete 2"
Call ShowErrorMessage
Call CopyTable("材料テーブルBACKUP", "材料テーブル", IFERR)
Call DeleteTable("材料テーブルBACKUP", IFERR)
Form_Load 'リストボックスの更新のため

End Sub

Private Sub cmdDelete_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then cmdDelete.ShowWhatsThis
End Sub

Private Sub cmdExit_Click()
    Unload Me
End Sub

Private Sub cmdExit_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then cmdExit.ShowWhatsThis
End Sub

Private Sub cmdReg_Click()
    Dim dtMAT As rdoResultset
    Dim SQL$
    Dim IsEOF As Boolean
    Dim IFERR As Integer
    Dim hMAT As Integer

    On Error GoTo Err1

    '材料名が入力されているかをチェック
    If txtNewMat.Text = "" Then
        MsgBox "新規材料名が指定されていません。"
        Exit Sub
    End If

    'その材料名が重複していないかをチェック
    Screen.MousePointer = 11
    SQL$ = "SELECT 材料名 FROM 材料テーブル WHERE " & "材料名 = " & txtNewMat.Text & ";"
    'Set dtMAT = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
    Set dtMAT = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hMAT)
    If hMAT = -1 Then GoTo Err1

    IsEOF = dtMAT.EOF
    dtMAT.Close
    xCloseResultset dtMAT, hMAT
    Screen.MousePointer = 0
    If IsEOF = False Then
        MsgBox "その材料名はすでに登録されています。"
        Exit Sub
    End If

    '-----

    Screen.MousePointer = 11

    'バックアップテーブルの作成
    Call CopyTable("材料テーブル", "材料テーブルBACKUP", IFERR)
    If IFERR = 1 Then GoTo Err1

    On Error GoTo Err2

    '新規材料名の登録
    SQL$ = "SELECT * FROM 材料テーブル;"

```

```

' Set dtMAT = dbSTAR5.OpenResultset(SQL$, rdOpenDynamic, rdConcurLock)
Set dtMAT = xOpenResultset(SQL$, rdOpenDynamic, rdConcurLock, hMAT)
If hMAT = -1 Then GoTo Err2

dtMAT.AddNew
dtMAT![id] = txtNewID.Text
dtMAT![材料名] = txtNewMat.Text
dtMAT![備考] = txtNewBiko.Text
dtMAT.Update

'dtMAT.Close
xCloseResultset dtMAT, hMAT

'リストボックスの更新
lstMat.AddItem txtNewMat.Text
lstMat.Refresh

'コントロールのフロンティの設定
txtNewMat.Text = ""
txtNewBiko.Text = ""
txtNewID.Text = ""
lblNewMat.Enabled = False
txtNewMat.Enabled = False
lblNewBiko.Enabled = False
txtNewBiko.Enabled = False
lblNewID.Enabled = False
txtNewID.Enabled = False
cmdReg.Enabled = False
cmdCancel.Enabled = False
cmdAdd.Enabled = True
cmdDelete.Enabled = True
cmdExit.Enabled = True

'バックアップしておいたテーブルを削除(処理が終了したので)
Call DeleteTable("材料テーブルBACKUP", IFERR)

Screen.MousePointer = 0

Exit Sub

Err1:
Screen.MousePointer = 0
MsgBox "Error in frmUtilMatReg/cmdReg 1"
Call ShowErrorMessage
xCloseResultset dtMAT, hMAT
Exit Sub

Err2:
Screen.MousePointer = 0
MsgBox "Error in frmUtilMatReg/cmdReg 2"
xCloseResultset dtMAT, hMAT
Call ShowErrorMessage
Call CopyTable("材料テーブルBACKUP", "材料テーブル", IFERR)
Call DeleteTable("材料テーブルBACKUP", IFERR)
Form_Load 'リストボックスの更新のため

End Sub

Private Sub cmdReg_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
If Button = 2 Then cmdReg.ShowWhatsThis
End Sub

Private Sub Form_Load()
Dim dtMAT As rdoResultset
Dim SQL$
Dim i As Long
Dim IFERR As Integer
Dim hMAT As Integer

On Error GoTo Err1

Screen.MousePointer = 11

'材料名をリストボックスに表示
SQL$ = "SELECT 材料名 FROM 材料テーブル ORDER BY id;"
' Set dtMAT = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
Set dtMAT = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hMAT)
If hMAT = -1 Then GoTo Err1

lstMat.Clear
Do Until dtMAT.EOF
lstMat.AddItem dtMAT![材料名]
dtMAT.MoveNext
Loop

'dtMAT.Close
xCloseResultset dtMAT, hMAT

cmdDelete.Enabled = False
txtNewMat.Text = ""

```

```

txtNewBiko.Text = ""
txtNewID.Text = ""

.Screen.MousePointer = 0
Exit Sub

Err1:
.Screen.MousePointer = 0
MsgBox "Error in frmUtilMatReg/Load"
Call ShowErrorMessage
On Error Resume Next
'dtMAT.Close
xCloseResultset dtMAT, hMAT
Unload Me          '強制終了-

End Sub

Private Sub lblMat_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblMat.ShowWhatsThis
End Sub

Private Sub lblNewBiko_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblNewBiko.ShowWhatsThis
End Sub

Private Sub lblNewID_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblNewID.ShowWhatsThis
End Sub

Private Sub lblNewMat_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblNewMat.ShowWhatsThis
End Sub

Private Sub lstMat_Click()
cmdDelete.Enabled = True
End Sub

Private Sub lstMat_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lstMat.ShowWhatsThis
End Sub

Private Sub txtNewBiko_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then txtNewBiko.ShowWhatsThis
End Sub

Private Sub txtNewID_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then txtNewID.ShowWhatsThis
End Sub

Private Sub txtNewMat_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then txtNewMat.ShowWhatsThis
End Sub

```



```

Attribute VB_Name = "frmUtilSetup"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Option Explicit

```

```

Private Sub Command1_Click()
    SaveSetting "STAR6", "STAR", "StarDir", "C:%STAR6%"
    SaveSetting "STAR6", "STAR", "WorkDir", "C:%STAR6%"
    SaveSetting "STAR6", "STAR", "DBDir", "C:%STAR6%"
    MsgBox GetSetting("STAR6", "STAR", "StarDir")
End Sub

```

```

Private Sub cmdBrowse_Click()

    CommonDialog1.Filter = "Access (*.MDB)|*.mdb"
    CommonDialog1.Flags = cdIOFNReadOnly

On Error Resume Next
    CommonDialog1.ShowOpen
On Error GoTo 0

    ' 復帰ボタンのチェック
    If Err.Number <> 0 Then
        If Err.Number = 32755 Then ' ユーザーがキャンセルを選択
            Exit Sub
        Else
            MsgBox "[ファイルを開く] アイコン ボックスで予期しないエラーが発生しました。"
            Exit Sub
        End If
    End If

    If CommonDialog1.FileName <> "" Then ' 入力ファイル名の取得
        txtQDB.Text = CommonDialog1.FileName
    End If

End Sub

```

```

Private Sub cmdBrowse_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then cmdBrowse.ShowWhatsThis
End Sub

```

```

Private Sub cmdCancel_Click()
    Unload Me
End Sub

```

```

Private Sub cmdCancel_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then cmdCancel.ShowWhatsThis
End Sub

```

```

Private Sub cmdOK_Click()
    On Error GoTo Err1

    SaveSetting "STAR7", "STAR", "DataSourceName", txtDSN.Text
    SaveSetting "STAR7", "STAR", "DatabaseName", txtDBN.Text
    SaveSetting "STAR7", "STAR", "QueryDB", txtQDB.Text
    SaveSetting "STAR7", "STAR", "BTView", cmbBTView.ListIndex
    SaveSetting "STAR7", "STAR", "MaxRetrieve", txtMaxRet
    SaveSetting "STAR7", "STAR", "WorkDir", txtWorkDir.Text
    SaveSetting "STAR7", "STAR", "WSketchDir", txtWSketchDir.Text
    MsgBox "STARを再起動してください。"
    Unload Me

End

```

```

Err1:
    MsgBox "Error in frmUtilSetup/cmdOK"
    Call ShowErrorMessage
End Sub

```

```

Private Sub cmdOK_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then cmdOK.ShowWhatsThis
End Sub

```

```

Private Sub Form_Load()
    On Error GoTo Err1

    cmbBTView.AddItem "クワット*コントロール"
    cmbBTView.AddItem "EXCEL(ファイル/OLE)"

    txtDSN.Text = GetSetting("STAR7", "STAR", "DataSourceName")
    txtDBN.Text = GetSetting("STAR7", "STAR", "DatabaseName")
    txtQDB.Text = GetSetting("STAR7", "STAR", "QueryDB")
    cmbBTView.ListIndex = Val(GetSetting("STAR7", "STAR", "BTView"))
    txtMaxRet.Text = Val(GetSetting("STAR7", "STAR", "MaxRetrieve"))
    txtWorkDir.Text = GetSetting("STAR7", "STAR", "WorkDir")
    txtWSketchDir.Text = GetSetting("STAR7", "STAR", "WSketchDir")

    Exit Sub

Err1:
    MsgBox "Error in frmUtilSetup/Load"
    Call ShowErrorMessage
    Unload Me ' 強制終了-
End Sub

Private Sub lblBTView_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblBTView.ShowWhatsThis
End Sub

Private Sub lblDBN_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblDBN.ShowWhatsThis
End Sub

Private Sub lblDSN_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblDSN.ShowWhatsThis
End Sub

Private Sub lblMaxRet_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblMaxRet.ShowWhatsThis
End Sub

Private Sub lblNote_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblNote.ShowWhatsThis
End Sub

Private Sub lblQDB_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblQDB.ShowWhatsThis
End Sub

Private Sub lblWorkDir_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblWorkDir.ShowWhatsThis
End Sub

Private Sub lblWSketchDir_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblWSketchDir.ShowWhatsThis
End Sub

Private Sub txtDBN_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then txtDBN.ShowWhatsThis

```

End Sub

```
Private Sub txtDSN_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then txtDSN.ShowWhatsThis
End Sub
```

```
Private Sub txtMaxRet_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then txtMaxRet.ShowWhatsThis
End Sub
```

```
Private Sub txtQDB_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then txtQDB.ShowWhatsThis
End Sub
```

```
Private Sub txtWorkDir_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then txtWorkDir.ShowWhatsThis
End Sub
```

```
Private Sub txtWSketchDir_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then txtWSketchDir.ShowWhatsThis
End Sub
```

```

Attribute VB_Name = "frmUtilSQL"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Private ActionQ As Long

```

```

Private Sub chkSelect_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)

```

```

    If Button = 2 Then chkSelect.ShowWhatsThis

```

```

End Sub

```

```

Private Sub cmdCancel_Click()

```

```

    Unload Me

```

```

End Sub

```

```

Private Sub cmdCancel_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)

```

```

    If Button = 2 Then cmdCancel.ShowWhatsThis

```

```

End Sub

```

```

Private Sub cmdOK_Click()

```

```

    Dim dtRET As rdoResultset

```

```

    Dim i As Long

```

```

    Dim j As Long

```

```

    Dim Rows As Long

```

```

    Dim Cols As Long

```

```

    Dim hRET As Integer

```

```

    On Error GoTo Err1

```

```

    If UCase(Left(txtSQL.Text, 6)) = "SELECT" Then

```

```

        chkSelect.Value = 1

```

```

    ElseIf UCase(Left(txtSQL.Text, 6)) = "UPDATE" Then

```

```

        chkSelect.Value = 0

```

```

    ElseIf UCase(Left(txtSQL.Text, 6)) = "DELETE" Then

```

```

        chkSelect.Value = 0

```

```

    End If

```

```

    Screen.MousePointer = 11

```

```

    If chkSelect.Value = 1 Then      '選択クエリ-

```

```

        frmUtilSQL.Height = 7600

```

```

        'Set dtRET = dbSTAR5.OpenResultset(txtSQL.Text, rdOpenStatic, rdConcurReadOnly)

```

```

        Set dtRET = xOpenResultset(txtSQL.Text, rdOpenStatic, rdConcurReadOnly, hRET)

```

```

        If hRET = -1 Then GoTo Err1

```

```

        If dtRET.EOF = False Then

```

```

            dtRET.MoveLast

```

```

            Rows = dtRET.RowCount

```

```

            Cols = dtRET.rdoColumns.Count

```

```

            grdSelect.Rows = Rows + 1

```

```

            grdSelect.Cols = Cols + 1

```

```

            grdSelect.Row = 0

```

```

            For j = 0 To Cols - 1

```

```

                grdSelect.ColWidth(j + 1) = 1000

```

```

                grdSelect.Col = j + 1

```

```

                grdSelect.Text = dtRET.rdoColumns(j).Name

```

```

            Next j

```

```

            dtRET.MoveFirst

```

```

            For i = 1 To Rows

```

```

                grdSelect.Row = i

```

```

                For j = 0 To Cols - 1

```

```

                    grdSelect.Col = j + 1

```

```

                    If IsNull(dtRET.rdoColumns(j).Value) = False Then

```

```

                        grdSelect.Text = dtRET.rdoColumns(j).Value

```

```

                    End If

```

```

                Next j

```

```

                dtRET.MoveNext

```

```

            Next i

```

```

        End If

```

```

        'dtRET.Close

```

```

        xCloseResultset dtRET, hRET

```

```

    Else

```

```

        'アクションクエリ-

```

```

        frmUtilSQL.Height = 4000

```

```

        dbSTAR5.Execute txtSQL.Text

```

```

        Rows = dbSTAR5.RowsAffected

```

```

        MsgBox Rows & " 行を処理しました。"

```

```

    If Rows > 0 Then
        ActionQ = 1
    End If

End If

Screen.MousePointer = 0

Exit Sub

Err1:
Screen.MousePointer = 0
MsgBox "Error in frmUtilSQL/cmdOK"
Call ShowErrorMessage
xCloseResultset dtRET, hRET

End Sub

Private Sub cmdOK_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then cmdOK.ShowWhatsThis
End Sub

Private Sub Form_Load()
    frmUtilSQL.Height = 4000
    txtSQL.Text = ""
    ActionQ = 0          'アクションエラーを実行したか
End Sub

Private Sub Form_Unload(Cancel As Integer)
    Dim IFERR As Integer

    '管理データベースの更新
    Call KanriRead(IFERR)
    Call KanriSequence(IFERR)

End Sub

Private Sub grdSelect_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then grdSelect.ShowWhatsThis
End Sub

Private Sub lblSQL_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblSQL.ShowWhatsThis
End Sub

Private Sub txtSQL_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then txtSQL.ShowWhatsThis
End Sub

```

```
Attribute VB_Name = "frmUtilKanriView"  
Attribute VB_GlobalNameSpace = False  
Attribute VB_Creatable = False  
Attribute VB_PredeclaredId = True  
Attribute VB_Exposed = False  
Option Explicit
```

```
Private Sub cmdClose_Click()  
    Unload Me  
End Sub
```

```
Private Sub Form_Initialize()  
    'rdcKanri.DataSourceName = modSTAR.DSN$  
    'MsgBox rdcKanri.DataSourceName  
    'rdcKanri.SQL = "SELECT * FROM 管理テ-クテ-フ" & I;  
End Sub
```

```
Private Sub Form_Load()  
    'dtaKanri.DatabaseName = modSTAR.StarDB$  
    'dtaKanri.RecordSource = "管理テ-クテ-フ" & I  
    rdcKanri.DataSourceName = modSTAR.DSN$  
    rdcKanri.SQL = "SELECT * FROM 管理テ-クテ-フ" & I;  
    rdcKanri.Enabled = True  
End Sub
```

```
Private Sub Form_Resize()  
    dbgKanri.Width = ScaleWidth  
    dbgKanri.Height = ScaleHeight - cmdClose.Height * 1.5  
  
    cmdClose.Top = ScaleHeight - cmdClose.Height * 1  
    cmdClose.Width = ScaleWidth  
End Sub
```

```

Attribute VB_Name = "frmNormal"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Option Explicit

```

```

Private Sub Command1_Click()
    Dim IFERR As Integer
    Call KanriAdd(1)
    Dim dtKANRI As rdoResultset
    Dim SQL$

    On Error GoTo Err1

    'この方法だと、修正した行が最終行へ移動してしまう。
    SQL$ = "UPDATE 管理テ-タ-フ* SET 評価マ-数 = 10 WHERE 参照id = 1;"
    dbSTAR5.Execute SQL$

    'この方法なら、移動しない。
    SQL$ = "SELECT 評価マ-数 FROM 管理テ-タ-フ* WHERE 参照id = 1;"
    Set dtKANRI = dbSTAR5.OpenResultset(SQL$, rdOpenDynamic, rdConcurLock)
    If dtKANRI.EOF = False Then
        dtKANRI.Edit
        dtKANRI![評価マ-数] = 10
        dtKANRI.Update
    End If
    dtKANRI.Close

Err1:
    Call ShowErrorMessage
End Sub

```

```

Private Sub Command2_Click()
    Dim InsPos As Long
    Dim IFERR As Integer
    InsPos = 1
    Call KanriSort(InsPos, IFERR)

    dbSTAR5.Execute "UPDATE 管理テ-タ-フ* SET id = 0 WHERE 参照id = 1;"
    dbSTAR5.Execute "UPDATE 管理テ-タ-フ* SET id = 1 WHERE 参照id = 2;"
    dbSTAR5.Execute "UPDATE 管理テ-タ-フ* SET id = 2 WHERE 参照id = 3;"
End Sub

```

```

Private Sub Command3_Click()
    Dim dtTMP As rdoResultset
    Dim SQL$
    Dim i As Long
    Dim xHandle As Integer

    SQL$ = "SELECT id FROM 管理テ-タ-フ*;"
    Set dtTMP = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
    Set dtTMP = xOpenResultset(SQL$, rdOpenDynamic, rdConcurLock, xHandle)

    dtTMP.MoveLast
    MsgBox dtTMP.RowCount

    dtTMP.Close
    xCloseResultset dtTMP, xHandle
End Sub

```

```

Private Sub Form_Load()
    Dim SV As Boolean

    SV = modSTAR.SuperVisor

    'テ-タ-選択(D)メニュー
    mnuSelBubun.Enabled = True
    mnuSelCase.Enabled = False
    mnuSelView.Enabled = True
    mnuSelShape.Enabled = False '未作成

    '1次テ-タ-(I)メニュー
    mnuPriCrackReg.Enabled = False
    mnuPriCrackShow.Enabled = False
    mnuPriCrackInt.Enabled = False
    mnuPriCondReg.Enabled = False
    mnuPriCondShow.Enabled = False
    mnuPriTempReg.Enabled = False
    mnuPriTempShow.Enabled = False
    mnuPriShapeReg.Enabled = False
    mnuPriShapeShow.Enabled = False

    '2次テ-タ-(N)メニュー
    mnuSecAnalReg.Enabled = False
    mnuSecAnalShow.Enabled = False
    mnuSecTableAll.Enabled = SV
    mnuSecTableTai.Enabled = False

```

```

mnuSecTableCase.Enabled = False
mnuSecTableShow.Enabled = False
mnuSecDamageSel.Enabled = SV
mnuSecDamageAll.Enabled = SV
mnuSecDamageTai.Enabled = False
mnuSecDamageCase.Enabled = False
mnuSecDamageShow.Enabled = False
mnuSecQCalc.Enabled = False
mnuSecFinasReg.Enabled = SV
mnuSecFinasShow.Enabled = True

```

```

' 分布比較(B)メニュー-
mnuBunExec.Enabled = False
' mnuBunShow.Enabled = False
' mnuBunExternal.Enabled = False

```

```

' 統計(T)メニュー-
mnuTokCrack.Enabled = False
mnuTokInterAll.Enabled = SV
mnuTokInterTai.Enabled = False
mnuTokInterCase.Enabled = False
mnuTokDataView.Enabled = False
mnuTokExec.Enabled = True
' mnuTokShow.Enabled = True
' mnuTokExternal.Enabled = True

```

```

' STAR制御(S)メニュー-
mnuUtilView.Enabled = True
mnuUtilKanri.Enabled = SV
mnuUtilCase.Enabled = SV
mnuUtilHyoka.Enabled = SV
mnuUtilMat.Enabled = SV
' mnuUtilCompress.Enabled = SV
' mnuUtilPassword.Enabled = SV
mnuUtilSetup.Enabled = True
mnuUtilSQL.Enabled = SV

```

End Sub

```

Private Sub Form_Resize()
Dim FObj As Object

```

```

Set FObj = modSTAR.frmMain

```

```

FObj.lblSochi.Width = Max2(500, FObj.Width - FObj.lblSochi.Left - 400)
FObj.lblTai.Width = Max2(500, FObj.Width - FObj.lblTai.Left - 400)
FObj.lblBubun.Width = Max2(500, FObj.Width - FObj.lblBubun.Left - 400)
FObj.lblCase.Width = Max2(500, FObj.Width - FObj.lblCase.Left - 400)

```

```

Set FObj = Nothing

```

End Sub

```

Private Sub Form_Unload(Cancel As Integer)
'MsgBox "Child Unload"
Cancel = 1
End Sub

```

```

Private Sub lblBubun_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)

```

```

If Button = 2 Then Me.lblBubun.ShowWhatsThis

```

End Sub

```

Private Sub lblBubunX_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)

```

```

If Button = 2 Then Me.lblBubunX.ShowWhatsThis

```

End Sub

```

Private Sub lblCase_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)

```

```

If Button = 2 Then Me.lblCase.ShowWhatsThis

```

End Sub

```

Private Sub lblCaseX_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)

```

```

If Button = 2 Then Me.lblCaseX.ShowWhatsThis

```

End Sub

```

Private Sub lblSochi_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)

```



```

    If Button = 2 Then Me.lblSochi.ShowWhatsThis
End Sub

Private Sub lblSochiX_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then Me.lblSochiX.ShowWhatsThis
End Sub

Private Sub lblTai_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then Me.lblTai.ShowWhatsThis
End Sub

Private Sub lblTaiX_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then Me.lblTaiX.ShowWhatsThis
End Sub

Private Sub mnuBunExec_Click()
    Dim dbQ As Database
    Dim dtQ As Recordset
    Dim qdQ As QueryDef
    Dim frmGRD As New frmResGRD
    Dim frmDBG As New frmResDBG
    Dim frmOLE As New frmResOLE
    Dim objOLE As Object
    Dim AFG As Object
    Dim QName$, CName$
    Dim i As Long
    Dim j As Long
    Dim Rows As Long
    Dim Ret As Long
    Dim ResType As Long
    Dim UserInput As Variant

    On Error GoTo Err1

    ResType = 2
                '1:DBGrid
                '2:Grid
                '3:Excel(OLE)
                '4:Excel(ファイル)
                '5:Excel(クリップボード)

    If modSTAR.BTView = 0 Then ResType = 2
    If modSTAR.BTView = 1 Then ResType = 4

    ' 検索条件設定用のフォームを表示
    frmBunSelect.Show 1

    If modSTAR.tmpFormCloseType = vbCancel Then Exit Sub

    modSTAR.tmpFormName$ = modSTAR.CurBunQueryName$
    QName$ = modSTAR.CurBunQueryName$

    ' クエリ用のデータベースを開く
    If modSTAR.StarDB$ <> modSTAR.QueryDB$ Then
        Set dbQ = Workspaces(0).OpenDatabase(modSTAR.QueryDB$)
    Else
        Set dbQ = dbSTAR
    End If
    Set qdQ = dbQ.QueryDefs(QName$)

    ' 未定義のパラメータの設定
    For i = 0 To qdQ.Parameters.Count - 1
        Debug.Print i, qdQ.Parameters(i).Name
        Select Case qdQ.Parameters(i).Name
            Case "試験装置"
                qdQ.Parameters(i).Value = modSTAR.nmSochi$
            Case "試験体"
                qdQ.Parameters(i).Value = modSTAR.nmTai$
            Case "試験部分"
                qdQ.Parameters(i).Value = modSTAR.nmBubun$
            Case "解析ケース"
                qdQ.Parameters(i).Value = modSTAR.CurrentCase$
            Case "ライン番号"
                qdQ.Parameters(i).Value = modSTAR.CurBunLineNo + 1
            Case Else
                UserInput = InputBox(qdQ.Parameters(i).Name, "パラメータの入力")
                If UserInput <> "" Then
                    qdQ.Parameters(i).Value = UserInput
                Else
                    If modSTAR.StarDB$ <> modSTAR.QueryDB$ Then dbQ.Close
                End If
            End Select
        Next i
    End For

```

```

Exit Sub
End If
End Select
Next i

' 検索結果の表示-----
Screen.MousePointer = 11
If ResType = 1 Then          ' 結果をDBGridコントロールに表示
    ' 結果表示用の子フォームをオープン
    modSTAR.CurQuerySQL$ = qdQ.SQL
    frmDBG.Show
    frmStarMain.ActiveForm.Caption = QName$
ElseIf ResType = 2 Then     ' 結果をGridコントロールに表示
    ' QueryDefから検索
    Set dtQ = qdQ.OpenRecordset()
    If dtQ.RecordCount > 0 Then
        dtQ.MoveLast
        Rows = dtQ.RecordCount
        Ret = vbYes
        If Rows > modSTAR.MaxRet Then
            Ret = MsgBox(Rows & "件検索されました。結果を表示しますか?", vbYesNo)
        End If
        If Ret = vbYes Then
            ' 結果表示用の子フォームをオープン
            frmGRD.Show
            frmStarMain.ActiveForm.Caption = QName$
            Set AFG = frmStarMain.ActiveForm.grdRes
            ' dtQ.MoveLast
            ' AFG.Rows = dtQ.RecordCount + 2
            AFG.Rows = Rows + 2
            AFG.Cols = dtQ.Fields.Count + 1
            dtQ.MoveFirst
            For j = 0 To dtQ.Fields.Count - 1
                AFG.ColWidth(j + 1) = 1000          ' テキストは630twip
                AFG.Col = j + 1
                AFG.Row = 0
                AFG.Text = dtQ.Fields(j).Name
                AFG.Row = 1
                AFG.Text = dtQ.Fields(j).Name
            Next j
            i = 1
            Do Until dtQ.EOF
                AFG.Row = i + 1
                AFG.Col = 0
                AFG.Text = i
                For j = 0 To dtQ.Fields.Count - 1
                    AFG.Col = j + 1
                    AFG.Text = dtQ.Fields(j).Value
                Next j
                dtQ.MoveNext
                i = i + 1
            Loop
        End If
    Else
        MsgBox "検索件数は0件でした。"
    End If
    dtQ.Close
ElseIf ResType = 3 Then     ' 結果をExcel(OLE)シートに表示
    ' QueryDefから検索
    Set dtQ = qdQ.OpenRecordset()
    If dtQ.RecordCount > 0 Then
        ' 結果表示用の子フォームをオープン
        frmOLE.Show
        frmStarMain.ActiveForm.Caption = QName$
        frmStarMain.ActiveForm.oleCtrl.Visible = False
        frmStarMain.ActiveForm.oleCtrl.CreateEmbed "", "Excel.Sheet.5"
        Set objOLE = frmStarMain.ActiveForm.oleCtrl.object
        For j = 0 To dtQ.Fields.Count - 1
            objOLE.Cells(1, j + 1).Value = dtQ.Fields(j).Name
        Next j
        i = 2
        Do Until dtQ.EOF
            For j = 0 To dtQ.Fields.Count - 1

```

```

        objOLE.Cells(i, j + 1).Value = dtQ.Fields(j).Value
    Next j
    dtQ.MoveNext
    i = i + 1
Loop

Set objOLE = Nothing

frmStarMain.ActiveForm.oleCtrl.Visible = True

Else
    MsgBox "検索件数は0件でした。"

End If

dtQ.Close

ElseIf ResType = 4 Then          '結果をExcel(ファイル)シートに表示

'QueryDefから検索
Set dtQ = qdQ.OpenRecordset()

'結果をCSVファイルに格納
If dtQ.RecordCount > 0 Then

'結果表示用の子フォームをオープン
frmOLE.Show
frmStarMain.ActiveForm.Caption = QName$

'検索結果をテキストファイルに出力
modSTAR.tmpBunNo = modSTAR.tmpBunNo + 1
CName$ = modSTAR.WorkDir$ & "$BUN" & modSTAR.tmpBunNo & ".XLS"
Open CName$ For Output As #1
For j = 0 To dtQ.Fields.Count - 1
    Print #1, dtQ.Fields(j).Name; Chr$(9);
Next j
Print #1, Chr$(13); Chr$(10);
Do Until dtQ.EOF
    For j = 0 To dtQ.Fields.Count - 1
        Print #1, dtQ.Fields(j).Value; Chr$(9);
    Next j
    Print #1, Chr$(13); Chr$(10);
    dtQ.MoveNext
Loop
Close #1

'Excelを起動し、テキストファイルをオープン
frmStarMain.ActiveForm.oleCtrl.CreateEmbed CName$, "Excel.Sheet.5"
frmStarMain.ActiveForm.oleCtrl.CreateLink CName$

Else
    MsgBox "検索件数は0件でした。"

End If

dtQ.Close

ElseIf ResType = 5 Then          '結果をExcel(Clip)シートに表示

'QueryDefから検索
Set dtQ = qdQ.OpenRecordset()

'結果をCSVファイルに格納
If dtQ.RecordCount > 0 Then

'結果表示用の子フォームをオープン
frmOLE.Show
frmStarMain.ActiveForm.Caption = QName$

'検索結果をテキストファイルに出力
modSTAR.tmpBunNo = modSTAR.tmpBunNo + 1
CName$ = modSTAR.WorkDir$ & "$BUN" & modSTAR.tmpBunNo & ".XLS"
Open CName$ For Output As #1
For j = 0 To dtQ.Fields.Count - 1
    Print #1, dtQ.Fields(j).Name; Chr$(9);
Next j
Print #1, Chr$(13); Chr$(10);
Do Until dtQ.EOF
    For j = 0 To dtQ.Fields.Count - 1
        Print #1, dtQ.Fields(j).Value; Chr$(9);
    Next j
    Print #1, Chr$(13); Chr$(10);
    dtQ.MoveNext
Loop
Close #1

'Excelを起動し、テキストファイルをオープン
frmStarMain.ActiveForm.oleCtrl.CreateEmbed CName$, "Excel.Sheet.5"
frmStarMain.ActiveForm.oleCtrl.CreateLink CName$

Else
    MsgBox "検索件数は0件でした。"

End If

dtQ.Close

End If

```

```

If modSTAR.StarDB$ <> modSTAR.QueryDB$ Then
  dbQ.Close
End If

Screen.MousePointer = 0

Exit Sub

Err1:
Screen.MousePointer = 0
MsgBox "Error in mnuBunExec"
Call ShowErrorMessage
On Error Resume Next
Close #1
dtQ.Close

End Sub

Private Sub mnuHelpIndex_Click()
frmStarMain.CommonDialog1.HelpCommand = cd1HelpContents
frmStarMain.CommonDialog1.HelpCommand = cd1HelpContext
frmStarMain.CommonDialog1.HelpFile = App.HelpFile
frmStarMain.CommonDialog1.ShowHelp
End Sub

Private Sub mnuSecAnalReg_Click()
Dim dbEX_ANAL As Database      ' Excelのシート
Dim dtEX_ANAL As Recordset    ' Excelのワークシート上のデータ-テーブル
Dim dtANL As rdoResultset     ' 解析テーブル
Dim dtJT1 As rdoResultset
Dim dtJT2 As rdoResultset
Dim i As Long
Dim j As Long
Dim k As Long
Dim m As Long
Dim n As Long
Dim i1 As Long
Dim i2 As Long
Dim i3 As Long
Dim j1 As Long
Dim j2 As Long
Dim j3 As Long
Dim FName1$, FName2$, FName3$
Dim TName1$, TName2$, TName3$
Dim FExtension$, EXName$, DBName$
Dim SQL$
Dim PrevR1 As Double
Dim PrevZ1 As Double
Dim PrevT1 As Double
Dim LL As Single
Dim OpenFileDialogType As Long
Dim AMaxID As Long
Dim BMaxID As Long
Dim JMaxID As Long
Dim RMaxID As Long
Dim IFERR As Integer
Dim hANL As Integer
Dim hJT1 As Integer
Dim hJT2 As Integer

On Error GoTo Err1

' =====
OpenFileDialogType = 2      ' 1:コメントファイル
                          ' 2:X_SELD839ファイル

If OpenFileDialogType = 1 Then ' コメントファイルを使用
  frmSecAnalReg.Show 1

  If modSTAR.tmpFormCloseType = vbCancel Then Exit Sub

  ' 入力ファイル名の取得
  FName1$ = modSTAR.tmpInpFName$(0)
  FName2$ = modSTAR.tmpInpFName$(1)
  FName3$ = modSTAR.tmpInpFName$(2)
  TName1$ = "Database"
  TName2$ = "Database"
  TName3$ = "Database"

ElseIf OpenFileDialogType = 2 Then ' X_SELD839ファイルを使用
  modSTAR.tmpInpFilter = "Excel 8.0 (*.XLS)|" &
    & "FT037711 (*.FT3)|" &
    & "標準テキスト (*.STF)|" &
    & "CSVファイル (*.CSV)|"

  frmSelDB3.Show 1

  If modSTAR.tmpFormCloseType = vbCancel Then Exit Sub

```

```

'入力ファイル名の取得
FName1$ = modSTAR.tmpInpFName$(0)
FName2$ = modSTAR.tmpInpFName$(1)
FName3$ = modSTAR.tmpInpFName$(2)
TName1$ = modSTAR.tmpInpTName$(0)
TName2$ = modSTAR.tmpInpTName$(1)
TName3$ = modSTAR.tmpInpTName$(2)

End If

'-----
FExtension$ = Right$(FName1$, 3)

If StrComp(FExtension$, "STF", 1) = 0 Then      '標準テキストファイル

' STFファイルをCSV形式に変換
' Call AnalStfToCsv
For i = 0 To modSTAR.NoOfLine - 1
    Call StfToCsv(modSTAR.tmpInpFName$(i), IFERR)
    If IFERR = 1 Then GoTo Err1
Next i

' CSVファイルを一時的に表示
i1 = InStrRev(FName1$, "#")      '*の位置
i2 = InStrRev(FName2$, "#")
i3 = InStrRev(FName3$, "#")
j1 = Len(FName1$) - i1 - 4      '拡張子を除くファイル名の長さ
j2 = Len(FName2$) - i2 - 4
j3 = Len(FName3$) - i3 - 4
modSTAR.tmpDBConnect$ = "Text;"
modSTAR.tmpDBName$(0) = Left$(FName1$, i1)
modSTAR.tmpDBName$(1) = Left$(FName2$, i2)
modSTAR.tmpDBName$(2) = Left$(FName3$, i3)
modSTAR.tmpDBRecSrc$(0) = Mid$(FName1$, i1 + 1, j1) & "#CSV"
modSTAR.tmpDBRecSrc$(1) = Mid$(FName2$, i2 + 1, j2) & "#CSV"
modSTAR.tmpDBRecSrc$(2) = Mid$(FName3$, i3 + 1, j3) & "#CSV"
frmSecAnalView.Show 1
If modSTAR.tmpFormCloseType = vbCancel Then
    Exit Sub
End If

ElseIf StrComp(FExtension$, "XLS", 1) = 0 Then  'Excelシート

' Excelシートを一時的に表示
modSTAR.tmpDBConnect$ = "Excel 8.0;"
modSTAR.tmpDBName$(0) = FName1$
modSTAR.tmpDBName$(1) = FName2$
modSTAR.tmpDBName$(2) = FName3$
modSTAR.tmpDBRecSrc$(0) = TName1$
modSTAR.tmpDBRecSrc$(1) = TName2$
modSTAR.tmpDBRecSrc$(2) = TName3$
frmSecAnalView.Show 1
If modSTAR.tmpFormCloseType = vbCancel Then
    Exit Sub
End If

ElseIf StrComp(FExtension$, "MDB", 1) = 0 Then  'Access

' Excelシートを一時的に表示
modSTAR.tmpDBConnect$ = "Access 2.0;"
modSTAR.tmpDBName$(0) = FName1$
modSTAR.tmpDBName$(1) = FName2$
modSTAR.tmpDBName$(2) = FName3$
modSTAR.tmpDBRecSrc$(0) = TName1$
modSTAR.tmpDBRecSrc$(1) = TName2$
modSTAR.tmpDBRecSrc$(2) = TName3$
frmSecAnalView.Show 1
If modSTAR.tmpFormCloseType = vbCancel Then
    Exit Sub
End If

ElseIf StrComp(FExtension$, "CSV", 1) = 0 Then  'CSVファイル

' CSVファイルを一時的に表示
i1 = InStrRev(FName1$, "#")      '*の位置
i2 = InStrRev(FName2$, "#")
i3 = InStrRev(FName3$, "#")
j1 = Len(FName1$) - i1 - 4      '拡張子を除くファイル名の長さ
j2 = Len(FName2$) - i2 - 4
j3 = Len(FName3$) - i3 - 4
modSTAR.tmpDBConnect$ = "Text;"
modSTAR.tmpDBName$(0) = Left$(FName1$, i1)
modSTAR.tmpDBName$(1) = Left$(FName2$, i2)
modSTAR.tmpDBName$(2) = Left$(FName3$, i3)
modSTAR.tmpDBRecSrc$(0) = Mid$(FName1$, i1 + 1, j1) & "#CSV"
modSTAR.tmpDBRecSrc$(1) = Mid$(FName2$, i2 + 1, j2) & "#CSV"
modSTAR.tmpDBRecSrc$(2) = Mid$(FName3$, i3 + 1, j3) & "#CSV"

frmSecAnalView.Show 1

If modSTAR.tmpFormCloseType = vbCancel Then
    Exit Sub
End If

ElseIf StrComp(FExtension$, "FT3", 1) = 0 Then  'FT03ファイル

```

```

' CSVファイルを一時的に表示
i1 = InStrRev(FName1$, "¥")          ' ¥の位置
i2 = InStrRev(FName2$, "¥")
i3 = InStrRev(FName3$, "¥")
j1 = Len(FName1$) - i1 - 4          ' 拡張子を除くファイル名の長さ
j2 = Len(FName2$) - i2 - 4
j3 = Len(FName3$) - i3 - 4
modSTAR.tmpDBConnect$ = "Text;"
modSTAR.tmpDBName$(0) = Left$(FName1$, i1)
modSTAR.tmpDBName$(1) = Left$(FName2$, i2)
modSTAR.tmpDBName$(2) = Left$(FName3$, i3)
modSTAR.tmpDBRecSrc$(0) = Mid$(FName1$, i1 + 1, j1) & "#FT3"
modSTAR.tmpDBRecSrc$(1) = Mid$(FName2$, i2 + 1, j2) & "#FT3"
modSTAR.tmpDBRecSrc$(2) = Mid$(FName3$, i3 + 1, j3) & "#FT3"
frmSecAnalView.Show 1
If modSTAR.tmpFormCloseType = vbCancel Then
    Exit Sub
End If

Else
    MsgBox "認識できないファイル(拡張子)です。"
    Exit Sub

End If

' =====

' 入力ファイルのファイル名チェック
Screen.MousePointer = 11
For k = 0 To modSTAR.NoOfLine - 1
    Set dbEX_ANAL = OpenDatabase(modSTAR.tmpDBName$(k), False, False, modSTAR.tmpDBConnect$)
    Set dtEX_ANAL = dbEX_ANAL.OpenRecordset(modSTAR.tmpDBRecSrc$(k), dbOpenSnapshot)
    Call InputFieldCheck(41, dtEX_ANAL, IFERR)
    If IFERR > 0 Then GoTo Err1
    ' dtEX_ANAL.Close
    ' dbEX_ANAL.Close
    ' If IFERR = 1 Then
    '     Screen.MousePointer = 0
    '     Exit Sub
    ' End If
Next k
Screen.MousePointer = 0

' =====

' 解析結果の格納ケース番号の取得
frmSecAnalCase.Show 1

If modSTAR.tmpFormCloseType = vbCancel Then
    Exit Sub
End If

' =====

Screen.MousePointer = 11

If modSTAR.tmpMaxID = -1 Then
    RMaxID = GetMaxID("管理データファイル", "解析id") + 1 ' 新規
Else
    RMaxID = modSTAR.tmpMaxID ' 更新
End If

' 古い解析データを削除しておく
SQL$ = "SELECT 時点1id,時点2id FROM 解析 WHERE " & " " & " " & "(参照id = " & modSTAR.nmID & ");"
Set dtANL = dbSTAR5.OpenResultset(SQL$, rdOpenDynamic, rdConcurLock)
Do Until dtANL.EOF
    ' i1 = dtANL![時点1id]
    ' i2 = dtANL![時点2id]
    Call DeleteJiten(dtANL![時点1id], dtANL![時点2id])
    dtANL.MoveNext
Loop
dtANL.Close
Call DeleteDataRC("解析", modSTAR.nmID, modSTAR.tmpCurrentCase$)
Call DeleteAnalRC("解析", "時点", RMaxID, modSTAR.tmpCurrentCase$)
Call DeleteAnalA("解析", "時点", RMaxID)

' =====

AMaxID = GetMaxID("解析", "id")
JMaxID = GetMaxID("時点", "id")
AMaxID0 = AMaxID

' 解析ファイルのオープン
SQL$ = "SELECT * FROM 解析 WHERE " & " " & " " & "(解析id = " & RMaxID & ");"
Set dtANL = dbSTAR5.OpenResultset(SQL$, rdOpenDynamic, rdConcurLock)
Set dtANL = xOpenResultset(SQL$, rdOpenDynamic, rdConcurLock, hANL)
If hANL = -1 Then GoTo Err1

SQL$ = "SELECT * FROM 時点 WHERE (id = 0);"
Set dtJT1 = dbSTAR5.OpenResultset(SQL$, rdOpenDynamic, rdConcurLock)

```

```

Set dtJT1 = xOpenResultset(SQL$, rdOpenDynamic, rdConcurLock, hJT1)
If hJT1 = -1 Then GoTo Err1

' Set dtJT2 = dbSTAR5.OpenResultset(SQL$, rdOpenDynamic, rdConcurLock)
Set dtJT2 = xOpenResultset(SQL$, rdOpenDynamic, rdConcurLock, hJT2)
If hJT2 = -1 Then GoTo Err1

For k = 0 To modSTAR.NoOfLine - 1

' Excelシートのおフツ
Set dbEX_ANAL = OpenDatabase(modSTAR.tmpDBName$(k), False, False, modSTAR.tmpDBConnect$)
Set dtEX_ANAL = dbEX_ANAL.OpenRecordset(modSTAR.tmpDBRecSrc$(k), dbOpenSnapshot)

If dtEX_ANAL.RecordCount > 0 Then

LL = 0
PrevR1 = 99999
PrevZ1 = 99999
Do Until dtEX_ANAL.EOF
dtANL.AddNew
dtJT1.AddNew
dtJT2.AddNew
For j = 0 To dtEX_ANAL.Fields.Count - 1
EXName$ = dtEX_ANAL.Fields(j).Name ' 外部ファイルのフィールド名
For m = 0 To modSTAR.AFldMax - 1
' If EXName$ = modSTAR.AFName$(m, 0) Or _
' EXName$ = modSTAR.AFName$(m, 1) Or _
' EXName$ = modSTAR.AFName$(m, 2) Then
n = modSTAR.AFType(m, 1)
If modSTAR.AFType(m, 0) = 0 Then
dtANL.rdoColumns(n).Value = dtEX_ANAL.Fields(j).Value
Elseif modSTAR.AFType(m, 0) = 1 Then
dtJT1.rdoColumns(n).Value = dtEX_ANAL.Fields(j).Value
Elseif modSTAR.AFType(m, 0) = 2 Then
dtJT2.rdoColumns(n).Value = dtEX_ANAL.Fields(j).Value
End If
' Debug.Print "m="; m; dtANL.rdoColumns(m).Name; dtANL.rdoColumns(m).Value
Exit For
End If
Call TransAnalField(EXName$, n, DBName$)
' Debug.Print n; DBName$
If n = 0 Then dtANL.rdoColumns(DBName$).Value = dtEX_ANAL.Fields(j).Value
If n = 1 Then dtJT1.rdoColumns(DBName$).Value = dtEX_ANAL.Fields(j).Value
If n = 2 Then dtJT2.rdoColumns(DBName$).Value = dtEX_ANAL.Fields(j).Value
If n = 0 Or n = 1 Or n = 2 Then Exit For
Next m
Next j
dtANL![id] = AMaxID + 1
dtANL![参照id] = modSTAR.nmID
dtANL![解析id] = RMaxID
dtANL![試験条件id] = modSTAR.nmID
dtANL![日付] = Date
dtANL![解析ケース] = modSTAR.tmpCurrentCase$
dtANL![ライン番号] = k + 1
If PrevR1 = 99999 And PrevZ1 = 99999 Then
LL = 0
Else
LL = LL + Sqr((dtANL![起点r1] - PrevR1) ^ 2 + (dtANL![起点z1] - PrevZ1) ^ 2)
End If
' Debug.Print LL
dtANL![L] = LL
dtANL![時点1id] = JMaxID + 1 ' 時点idの設定
dtANL![時点2id] = JMaxID + 2 ' 時点idの設定
dtJT1![id] = JMaxID + 1 ' 時点idの設定
dtJT2![id] = JMaxID + 2 ' 時点idの設定
dtJT1![参照id] = modSTAR.nmID ' 参照idの設定
dtJT2![参照id] = modSTAR.nmID ' 参照idの設定
dtJT1![解析id] = RMaxID ' 解析idの設定
dtJT2![解析id] = RMaxID ' 解析idの設定
dtJT1![解析ケース] = modSTAR.tmpCurrentCase$ ' 解析ケースの設定
dtJT2![解析ケース] = modSTAR.tmpCurrentCase$ ' 解析ケースの設定
JMaxID = JMaxID + 2 ' 時点idの更新
AMaxID = AMaxID + 1 ' 解析idの更新
PrevR1 = dtANL![起点r1]
PrevZ1 = dtANL![起点z1]
dtANL.Update
dtJT1.Update
dtJT2.Update
dtEX_ANAL.MoveNext
Loop

End If

dtEX_ANAL.Close ' Excelシートのクローズ
dbEX_ANAL.Close ' Excelシートのクローズ

Next k

' dtJT2.Close ' 時点テーブルのクローズ
' dtJT1.Close ' 時点テーブルのクローズ
' dtANL.Close ' 解析テーブルのクローズ
xCloseResultset dtJT2, hJT2 ' 時点テーブルのクローズ
xCloseResultset dtJT1, hJT1 ' 時点テーブルのクローズ
xCloseResultset dtANL, hANL ' 解析テーブルのクローズ

' Call SetKanriXID("解析id", modSTAR.nmID, modSTAR.tmpCurrentCase$, 0)
' Call SetKanriXID("解析id", RMaxID, modSTAR.nmID, modSTAR.tmpCurrentCase$, 0)
Call SetKanriXID6("解析id", RMaxID, modSTAR.nmID, modSTAR.tmpCurrentCase$, 0)

```

```

' 現解析ケース名の設定
modSTAR.CurrentCase$ = modSTAR.tmpCurrentCase$
modSTAR.nmAID = RMaxID

' 各種コマンドを実行可に設定
modSTAR.frmMain.mnuSelShape.Enabled = True
modSTAR.frmMain.mnuPriCrackInt.Enabled = True
modSTAR.frmMain.mnuSecAnalShow.Enabled = True
modSTAR.frmMain.mnuSecTableCase.Enabled = True
modSTAR.frmMain.mnuSecTableShow.Enabled = True
modSTAR.frmMain.mnuSecDamageCase.Enabled = True
modSTAR.frmMain.mnuSecDamageShow.Enabled = True
modSTAR.frmMain.mnuBunExec.Enabled = True
modSTAR.frmMain.mnuTokInterCase.Enabled = True
modSTAR.frmMain.mnuTokDataView.Enabled = True
' Normalフォームの更新
modSTAR.frmMain.tblCase.Caption = modSTAR.CurrentCase$

Screen.MousePointer = 0

MsgBox (AMaxID - AMaxID0) & " 行を書込みました。"

Exit Sub

Err1:
Screen.MousePointer = 0
MsgBox "Error in mnuSecAnalReg"
Call ShowErrorMessage
On Error Resume Next
dtEX_ANAL.Close
dbEX_ANAL.Close
'dtANL.Close
'dtJT1.Close
'dtJT2.Close
xCloseResultset dtANL, hANL
xCloseResultset dtJT1, hJT1
xCloseResultset dtJT2, hJT2

End Sub

Private Sub mnuSecAnalShow_Click()
Dim frmGRD As New frmPriViewGRD
Dim frmOLE As New frmPriViewOLE
Dim dtTMP As Recordset
Dim SQL$, CName$, QName$
Dim ResType As Long
Dim j As Long

On Error GoTo Err1

QName$ = "解析結果(" & modSTAR.CurrentCase$ & ")"
SQL$ = "SELECT * FROM 解析 WHERE " & " & "(参照id = " & modSTAR.nmAID & ")" _
& " ORDER BY ライン番号, L;"
SQL$ = "SELECT 解析.*, 時点1.*, 時点2.* " & "FROM (解析 INNER JOIN 時点 AS 時点1 ON 解析.時点1id = 時点1.id) " _
& "INNER JOIN 時点 AS 時点2 ON 解析.時点2id = 時点2.id " _
& "WHERE (解析.解析id = " & modSTAR.nmAID & ");"

ResType = 2
' 1:DBGrid
' 2:Grid
' 3:Excel(OLE)
' 4:Excel(ファイル)

If ResType = 1 Then
' DBGridコントロールに表示
' 解析フォームを表示
modSTAR.tmpFormName$ = QName$
modSTAR.tmpDBConnect$ = "Access"
modSTAR.tmpDBDatabaseName$ = modSTAR.StarDB$
modSTAR.tmpDBRecordSource$ = SQL$
modSTAR.tmpFormOpenType = vbOKCancel

frmPriViewDBG.Show 1
ElseIf ResType = 2 Then
' Gridコントロールに表示
' 表計算フォームを表示
modSTAR.tmpFormName$ = QName$
modSTAR.tmpViewSQL$ = SQL$

frmPriViewGRD.Show
frmGRD.Show
ElseIf ResType = 3 Then
' Excel(OLE)シートに表示
ElseIf ResType = 4 Then
' Excel(ファイル)シートに表示

```



```

'QueryDefから検索
Set dtTMP = dbSTAR.OpenRecordset(SQL$, dbOpenSnapshot)

'結果をCSVファイルに格納
If dtTMP.RecordCount > 0 Then
    '結果表示用の子フォームをオープン
    frmOLE.Show
    frmStarMain.ActiveForm.Caption = QName$

    '検索結果をテキストファイルに出力
    modSTAR.tmpViewNo = modSTAR.tmpViewNo + 1
    CName$ = modSTAR.WorkDir$ & "$VIEW" & modSTAR.tmpViewNo & ".XLS"
    Open CName$ For Output As #1
    For j = 0 To dtTMP.Fields.Count - 1
        Print #1, dtTMP.Fields(j).Name; Chr$(9);
    Next j
    Print #1, Chr$(13); Chr$(10);
    Do Until dtTMP.EOF
        For j = 0 To dtTMP.Fields.Count - 1
            If IsNull(dtTMP.Fields(j).Value) = True Then
                Debug.Print "NULL"
                Print #1, Chr$(9);
            Else
                Print #1, dtTMP.Fields(j).Value; Chr$(9);
            End If
        Next j
        Print #1, Chr$(13); Chr$(10);
        dtTMP.MoveNext
    Loop
    Close #1

    modSTAR.tmpViewSQL$ = SQL$
    modSTAR.tmpViewTableName$ = "解析"
    modSTAR.tmpViewTableRow = dtTMP.RecordCount
    modSTAR.tmpViewTableCol = dtTMP.Fields.Count

    'Excelを起動し、テキストファイルをオープン
    'frmStarMain.ActiveForm.oleView.CreateEmbed CName$, "Excel.Sheet.5"
    frmStarMain.ActiveForm.oleView.CreateLink CName$

End If

dtTMP.Close

End If

Exit Sub

Err1:
MsgBox "Error in mnuSecAnalShow"
Call ShowErrorMessage

End Sub

Private Sub mnuPriCondReg_Click()
Dim dtCOND As rdoResultset
Dim SQL$
Dim IsEOF As Boolean
Dim hCOND As Integer

On Error GoTo Err1

Screen.MousePointer = 11
SQL$ = "SELECT * FROM 試験条件 WHERE "
    & "(参照id = " & modSTAR.nmID & ")";
' Set dtCOND = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
Set dtCOND = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hCOND)
If hCOND = -1 Then GoTo Err1

IsEOF = dtCOND.EOF

'dtCOND.Close
xCloseResultset dtCOND, hCOND

Screen.MousePointer = 0

If IsEOF = False Then
    'すでに入力済み
    MsgBox "すでに登録されています。試験条件確認を使用してください。"
Else
    '新規登録
    modSTAR.tmpCondInpCtrl$ = "REG"
    frmPriCondReg.Show 1
End If

Exit Sub

Err1:
Screen.MousePointer = 0
MsgBox "Error in mnuPriCondReg"
Call ShowErrorMessage
On Error Resume Next
'dtCOND.Close
xCloseResultset dtCOND, hCOND

```

End Sub

```
Private Sub mnuPriCondShow_Click()
    Dim dtCOND As rdoResultset
    Dim SQL$
    Dim IsEOF As Boolean
    Dim hCOND As Integer

    On Error GoTo Err1

    Screen.MousePointer = 11
    SQL$ = "SELECT * FROM 試験条件 WHERE "
        & "(参照id = " & modSTAR.nmID & ")";
    'Set dtCOND = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
    Set dtCOND = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hCOND)
    If hCOND = -1 Then GoTo Err1

    IsEOF = dtCOND.EOF

    'dtCOND.Close
    xCloseResultset dtCOND, hCOND

    Screen.MousePointer = 0

    If IsEOF = True Then
        MsgBox "データが録されていません。試験条件入力を実行してください。"
    Else
        modSTAR.tmpCondInpCtrl$ = "SHOW"
        frmPriCondReg.Show 1
    End If

Exit Sub
```

```
Err1:
    Screen.MousePointer = 0
    MsgBox "Error in mnuPriCondShow"
    Call ShowErrorMessage
    On Error Resume Next
    'dtCOND.Close
    xCloseResultset dtCOND, hCOND
```

End Sub

```
Private Sub mnuPriCrackInt_Click()
    frmPriCrackInt.Show 1
End Sub
```

```
Private Sub mnuPriCrackReg_Click()
    Dim dbEX_CRACK As Database 'Excelのシート
    Dim dtEX_CRACK As Recordset 'Excelのワークシート上のデータへステータス
    Dim dtCRACK As rdoResultset 'き裂データ
    Dim i As Long
    Dim j As Long
    Dim SQL$, FName$, TName$, Buf$, FType$
    Dim OpenDialogType As Long
    Dim MaxID As Long
    Dim MaxIDO As Long
    Dim Clen As Single
    Dim IFERR As Integer
    Dim hCRACK As Integer
```

On Error GoTo Err1

modSTAR.tmpFormName\$ = "き裂データの入力"

```
OpenDialogType = 2 '1:コンタクト
                  '2:X_SELECT1*イロク'
```

```
If OpenDialogType = 1 Then 'コンタクト使用
    frmStarMain.CommonDialog1.Filter = "Excel 5.0 (*.xls)|*.xls| "
        & "テキスト出力 (*.csv)|*.csv"
    frmStarMain.CommonDialog1.Flags = cdIOFNReadOnly
```

```
On Error Resume Next
frmStarMain.CommonDialog1.ShowOpen
On Error GoTo 0
```

```
'復帰ボタンのチェック
If Err.Number <> 0 Then
    If Err.Number = 32755 Then 'ユーザがキャンセルを選択
        Exit Sub
    Else
        MsgBox "[ファイルを開く]がイロク*ボックスで予期しないエラーが発生しました。"
        Exit Sub
    End If
```

```

End If

'入力ファイル名の取得
FName$ = frmStarMain.CommonDialog1.FileName
TName$ = "Database"

ElseIf OpenDialogType = 2 Then 'X_SELDB1がイロク使用
modSTAR.tmpInpFilter = "Excel 8.0 (*.XLS)|"
& "データ出力 (*.CSV)|"
frmSelDB1.Show 1

If modSTAR.tmpFormCloseType = vbCancel Then Exit Sub

'入力ファイル名の取得
FName$ = modSTAR.tmpInpFName$(0)
TName$ = modSTAR.tmpInpTName$(0)

End If

'=====

If StrComp(Right$(FName$, 3), "XLS", 1) = 0 Then 'Excelシート

'Excelシートを一時的に表示
modSTAR.tmpDBConnect$ = "Excel 8.0;"
modSTAR.tmpDBDatabaseName$ = FName$
modSTAR.tmpDBRecordSource$ = TName$
modSTAR.tmpFormOpenType = vbOKOnly
FType$ = "XLS"

frmPriViewDBG.Show 1

If modSTAR.tmpFormCloseType = vbCancel Then
Exit Sub
End If

ElseIf StrComp(Right$(FName$, 3), "CSV", 1) = 0 Then 'CSVファイル

'CSVファイルにフォーマット行を付加
Open modSTAR.WorkDir$ & "$CRACK.CSV" For Output As #1
Print #1, "データ番号,日付,ライン番号,起点r1,起点z1,起点θ1,終点r2,終点z2";
Print #1, ",き裂長さ,母液-HAZ,溶接法,余盛有無,余盛形状,溶接備考";
Print #1, ",粒内・粒界,き裂形状,き裂発生サイクル,ストレイン,ヒートワーク";
Print #1, ",統計処理タイプ,結晶粒度,備考,テリミック,コート"
Open FName$ For Input As #2
Do While Not EOF(2)
Line Input #2, Buf$
Print #1, Buf$
Loop
Close #2
Close #1

'CSVファイルを一時的に表示
FName$ = modSTAR.WorkDir$ & "$CRACK.CSV"
i = InStrRev(FName$, ",") 'その位置
j = Len(FName$) - i - 4 '拡張子を除外するファイル名の長さ
modSTAR.tmpDBConnect$ = "Text;"
modSTAR.tmpDBDatabaseName$ = Left$(FName$, i)
modSTAR.tmpDBRecordSource$ = Mid$(FName$, i + 1, j) & "#CSV"
modSTAR.tmpFormOpenType = vbOKOnly
FType$ = "CSV"

frmPriViewDBG.Show 1

If modSTAR.tmpFormCloseType = vbCancel Then
Exit Sub
End If

Else
MsgBox "認識できないファイル(拡張子)です。"
Exit Sub

End If

'=====

Screen.MousePointer = 11

'古いき裂データを削除しておく
'Call DeleteData("き裂", "参照id", modSTAR.nmID)

'外部データのオフ*
Set dbEX_CRACK = OpenDatabase(tmpDBDatabaseName$, False, False, modSTAR.tmpDBConnect$)
Set dtEX_CRACK = dbEX_CRACK.OpenRecordset(tmpDBRecordSource$, dbOpenSnapshot)

'入力ファイルのフォーマット名チェック
Call InputFieldCheck(12, dtEX_CRACK, IFERR)
If IFERR > 0 Then GoTo Err1
'If IFERR = 1 Then
' dtEX_CRACK.Close
' dbEX_CRACK.Close
' Screen.MousePointer = 0
' Exit Sub
'End If

'古いき裂データを削除しておく

```

```

Call DeleteDataR("き裂", modSTAR.nmID)

'き裂テーブルのオープン
SQL$ = "SELECT * FROM き裂 WHERE ("
      & "参照id = " & modSTAR.nmID & ")";"
Set dtCRACK = dbSTAR5.OpenResultset(SQL$, rdOpenDynamic, rdConcurLock)
Set dtCRACK = xOpenResultset(SQL$, rdOpenDynamic, rdConcurLock, hCRACK)
If hCRACK = -1 Then GoTo Err1

MaxID = GetMaxID("き裂", "id") + 1
MaxID0 = MaxID

Do Until dtEX_CRACK.EOF
dtCRACK.AddNew
dtCRACK![id] = MaxID
dtCRACK![参照id] = modSTAR.nmID
dtCRACK![テ-ラ番号] = dtEX_CRACK![テ-ラ番号]
dtCRACK![日付] = dtEX_CRACK![日付]
dtCRACK![ライン番号] = dtEX_CRACK![ライン番号]
If FType$ = "CSV" Then 'CSVファイル
dtCRACK![起点r1] = dtEX_CRACK![起点r1]
dtCRACK![起点z1] = dtEX_CRACK![起点z1]
If Left(dtEX_CRACK![起点θ1], 2) = "C-" Then
dtCRACK![起点θ1] = Mid(dtEX_CRACK![起点θ1], 3)
Else
dtCRACK![起点θ1] = dtEX_CRACK![起点θ1]
End If
dtCRACK![終点r2] = dtEX_CRACK![起点r1] + dtEX_CRACK![終点r2]
dtCRACK![終点z2] = dtEX_CRACK![起点z1] + dtEX_CRACK![終点z2]
dtCRACK![終点θ2] = dtCRACK![起点θ1]
Else 'XLS, MDBファイル
dtCRACK![起点r1] = dtEX_CRACK![起点r1]
dtCRACK![起点z1] = dtEX_CRACK![起点z1]
dtCRACK![起点θ1] = dtEX_CRACK![起点θ1]
dtCRACK![終点r2] = dtEX_CRACK![終点r2]
dtCRACK![終点z2] = dtEX_CRACK![終点z2]
dtCRACK![終点θ2] = dtEX_CRACK![終点θ2]
End If
dtCRACK![き裂長さ] = dtEX_CRACK![き裂長さ]
dtCRACK![母・溶・HAZ] = GetDBData2(dtEX_CRACK![母・溶・HAZ])
dtCRACK![溶接法] = GetDBData2(dtEX_CRACK![溶接法])
dtCRACK![余盛有無] = GetDBData2(dtEX_CRACK![余盛有無])
dtCRACK![余盛形状] = GetDBData2(dtEX_CRACK![余盛形状])
dtCRACK![溶接備考] = GetDBData2(dtEX_CRACK![溶接備考])
dtCRACK![粒内・粒界] = GetDBData2(dtEX_CRACK![粒内・粒界])
dtCRACK![き裂形状] = GetDBData2(dtEX_CRACK![き裂形状])
dtCRACK![き裂発生ケイム] = GetDBData2(dtEX_CRACK![き裂発生ケイム])
dtCRACK![ストライエーション] = GetDBData2(dtEX_CRACK![ストライエーション])
dtCRACK![ヒ-マーク] = GetDBData2(dtEX_CRACK![ヒ-マーク])
dtCRACK![統計処理アクティブ] = GetDBData2(dtEX_CRACK![統計処理アクティブ])
dtCRACK![結晶粒度] = GetDBData2(dtEX_CRACK![結晶粒度])
dtCRACK![備考] = GetDBData2(dtEX_CRACK![備考])
' dtCRACK![母・溶・HAZ] = -dtEX_CRACK![母・溶・HAZ]
' dtCRACK![溶接法] = dtEX_CRACK![溶接法]
' dtCRACK![余盛有無] = dtEX_CRACK![余盛有無]
' dtCRACK![余盛形状] = dtEX_CRACK![余盛形状]
' dtCRACK![溶接備考] = dtEX_CRACK![溶接備考]
' dtCRACK![粒内・粒界] = dtEX_CRACK![粒内・粒界]
' dtCRACK![き裂形状] = dtEX_CRACK![き裂形状]
' dtCRACK![き裂発生ケイム] = dtEX_CRACK![き裂発生ケイム]
' dtCRACK![ストライエーション] = dtEX_CRACK![ストライエーション]
' dtCRACK![ヒ-マーク] = dtEX_CRACK![ヒ-マーク]
' dtCRACK![統計処理アクティブ] = dtEX_CRACK![統計処理アクティブ]
' dtCRACK![結晶粒度] = dtEX_CRACK![結晶粒度]
' dtCRACK![備考] = dtEX_CRACK![備考]

Clen = Sqr((dtCRACK![終点r2] - dtCRACK![起点r1]) ^ 2 + (dtCRACK![終点z2] - dtCRACK![起点z1]) ^ 2)
dtCRACK![き裂長さ] = Max2(Clen, 1#)
dtCRACK.Update

MaxID = MaxID + 1
dtEX_CRACK.MoveNext
Loop

dtCRACK.Close 'き裂テーブルのクローズ
xCloseResultset dtCRACK, hCRACK 'き裂テーブルのクローズ

dtEX_CRACK.Close 'Excelシートのクローズ
dbEX_CRACK.Close 'Excelシートのクローズ

'Call SetKanriXID("き裂id", modSTAR.nmID, modSTAR.nmID, "-", 0)
Call SetKanriXID6("き裂id", modSTAR.nmID, modSTAR.nmID, "-", 0)

Screen.MousePointer = 0

MsgBox (MaxID - MaxID0) & " 行を蓄込みました。"

Exit Sub

Err1:
Screen.MousePointer = 0
MsgBox "Error in mnuPriCrackReg"
Call ShowErrorMessage
On Error Resume Next
Close #1
Close #2
dtCRACK.Close
xCloseResultset dtCRACK, hCRACK
dtEX_CRACK.Close
dbEX_CRACK.Close

End Sub

Private Sub mnuPriCrackShow_Click()
Dim frmGRD As New frmPriViewGRD
Dim frmOLE As New frmPriViewOLE
Dim dtTMP As Recordset
Dim SQL$, SQLWS$, CName$, QName$

```

```

Dim ResType As Long
Dim j As Long

On Error GoTo Err1

QName$ = "き裂データ"
SQLW$ = " WHERE (参照id = " & modSTAR.nmID & ");"
SQL$ = "SELECT * FROM き裂" & SQLW$

ResType = 2          ' 1:DBGGrid
                    ' 2:Grid
                    ' 3:Excel(OLE)
                    ' 4:Excel(ファイル)

If ResType = 1 Then      ' DBGGridコントロールに表示
    ' き裂フォームを表示
    modSTAR.tmpFormName$ = QName$
    modSTAR.tmpDBConnect$ = "Access"
    modSTAR.tmpDBDatabaseName$ = modSTAR.StarDB$
    modSTAR.tmpDBRecordSource$ = SQL$
    modSTAR.tmpFormOpenType = vbOKCancel

    frmPriViewDBG.Show 1
ElseIf ResType = 2 Then  ' Gridコントロールに表示
    ' き裂フォームを表示
    modSTAR.tmpFormName$ = QName$
    modSTAR.tmpViewSQL$ = SQL$
    modSTAR.tmpDeleteSQL$ = "DELETE き裂" & SQLW$

    ' frmPriViewGRD.Show
    frmGRD.Show
ElseIf ResType = 3 Then  ' Excel(OLE)シートに表示
ElseIf ResType = 4 Then  ' Excel(ファイル)シートに表示
    ' QueryDefから検索
    Set dtTMP = dbSTAR.OpenRecordset(SQL$, dbOpenSnapshot)

    ' 結果をCSVファイルに格納
    If dtTMP.RecordCount > 0 Then
        ' 結果表示用の子フォームをオープン
        frmOLE.Show
        frmStarMain.ActiveForm.Caption = QName$

        ' 検索結果をテキストファイルに出力
        modSTAR.tmpViewNo = modSTAR.tmpViewNo + 1
        CName$ = modSTAR.WorkDir$ & "$VIEW" & modSTAR.tmpViewNo & ".XLS"
        Open CName$ For Output As #1
        For j = 0 To dtTMP.Fields.Count - 1
            Print #1, dtTMP.Fields(j).Name; Chr$(9);
        Next j
        Print #1, Chr$(13); Chr$(10);
        Do Until dtTMP.EOF
            For j = 0 To dtTMP.Fields.Count - 1
                If IsNull(dtTMP.Fields(j).Value) = True Then
                    Debug.Print "NULL"
                    Print #1, Chr$(9);
                Else
                    Print #1, dtTMP.Fields(j).Value; Chr$(9);
                End If
            Next j
            Print #1, Chr$(13); Chr$(10);
            dtTMP.MoveNext
        Loop
        Close #1

        modSTAR.tmpViewSQL$ = SQL$
        modSTAR.tmpViewTableName$ = "き裂"
        modSTAR.tmpViewTableRow = dtTMP.RecordCount
        modSTAR.tmpViewTableCol = dtTMP.Fields.Count

        ' Excelを起動し、テキストファイルをオープン
        ' frmStarMain.ActiveForm.oleView.CreateEmbed CName$, "Excel.Sheet.5"
        ' frmStarMain.ActiveForm.oleView.CreateLink CName$
    End If

    dtTMP.Close

End If

Exit Sub

Err1:
MsgBox "Error in mnuPriCrackShow"
Call ShowErrorMessage

End Sub

```

```

Private Sub mnuPriShapeReg_Click()
Dim dbEX_SHAPE As Database
Dim dtEX_SHAPE As Recordset
Dim dtSHAPE As rdoResultset      'データ用のデータベース上の形状データ
Dim FName$, TName$, SQL$, Buf$
Dim i As Long
Dim j As Long
Dim MaxID As Long
Dim MaxID0 As Long
Dim RCount As Long
Dim OpenFileDialogType As Long
Dim Value As Variant
Dim X0 As Double
Dim Y0 As Double
Dim X1 As Double
Dim Y1 As Double
Dim X2 As Double
Dim Y2 As Double
Dim IFERR As Integer
Dim hSHAPE As Integer

On Error GoTo Err1

modSTAR.tmpFormName$ = "形状データの入力"

' =====
OpenFileDialogType = 2          '1:コンタクト
                                '2:X_SELDB1データ

If OpenFileDialogType = 1 Then 'コンタクト使用
frmStarMain.CommonDialog1.Filter = "DXFファイル (*.DXF)|*.dxf|"
                                & "Excel 5.0 (*.XLS)|*.xls|"
                                & "CSVファイル (*.CSV)|*.csv|"
                                & "標準テキスト (*.STF)|*.stf"
frmStarMain.CommonDialog1.Flags = cdIOFNReadOnly

On Error Resume Next
frmStarMain.CommonDialog1.ShowOpen
On Error Goto 0

'復帰ボタンのシーク
If Err.Number <> 0 Then
If Err.Number = 32755 Then 'ユーザーがキャンセルを選択
Exit Sub
Else
MsgBox "[ファイルを開く]がコンタクトボックスで予期しないエラーが発生しました。"
Exit Sub
End If
End If

'入力ファイル名の取得
FName$ = frmStarMain.CommonDialog1.FileName
TName$ = "Database"

ElseIf OpenFileDialogType = 2 Then 'X_SELDB1データ使用
modSTAR.tmpInpFilter = "DXFファイル (*.DXF)|"
                        & "Excel 8.0 (*.XLS)|"
                        & "CSVファイル (*.CSV)|"
                        & "標準テキスト (*.STF)|"
frmSelDB1.Show 1

If modSTAR.tmpFormCloseType = vbCancel Then Exit Sub

'入力ファイル名の取得
FName$ = modSTAR.tmpInpFName$(0)
TName$ = modSTAR.tmpInpTName$(0)

End If

' =====

If StrComp(Right$(FName$, 3), "DXF", 1) = 0 Then 'DXFファイル
' DXFファイルを一時的に表示
modSTAR.tmpDBDatabaseName$ = FName$
modSTAR.tmpFormOpenType = vbOKOnly

frmPriViewDXF.Show 1

If modSTAR.tmpFormCloseType = vbCancel Then
Exit Sub
End If

ElseIf StrComp(Right$(FName$, 3), "XLS", 1) = 0 Then 'XLSファイル
' Excelシートを一時的に表示
modSTAR.tmpDBConnect$ = "Excel 8.0;"
modSTAR.tmpDBDatabaseName$ = FName$
modSTAR.tmpDBRecordSource$ = TName$
modSTAR.tmpFormOpenType = vbOKOnly

```

```

frmPriViewDBG.Show 1

If modSTAR.tmpFormCloseType = vbCancel Then
    Exit Sub
End If

ElseIf StrComp(Right$(FName$, 3), "MDB", 1) = 0 Then 'MDBファイル

' Accessファイルを一時的に表示
modSTAR.tmpDBConnect$ = "Access 2.0;"
modSTAR.tmpDBDatabaseName$ = FName$
modSTAR.tmpDBRecordSource$ = TName$
modSTAR.tmpFormOpenType = vbOKOnly

frmPriViewDBG.Show 1

If modSTAR.tmpFormCloseType = vbCancel Then
    Exit Sub
End If

ElseIf StrComp(Right$(FName$, 3), "CSV", 1) = 0 Then 'CSVファイル

' CSVファイルを一時的に表示
i = InStrRev(FName$, "¥")
j = Len(FName$) - i - 4
modSTAR.tmpDBConnect$ = "Text;"
modSTAR.tmpDBDatabaseName$ = Left$(FName$, i)
modSTAR.tmpDBRecordSource$ = Mid$(FName$, i + 1, j) & "#CSV"
modSTAR.tmpFormOpenType = vbOKOnly

frmPriViewDBG.Show 1

If modSTAR.tmpFormCloseType = vbCancel Then
    Exit Sub
End If

ElseIf StrComp(Right$(FName$, 3), "STF", 1) = 0 Then 'STFファイル

' STFファイルをCSV形式に変換
modSTAR.tmpInpFName$(0) = FName$
Call ShapeStfToCsv
Call StfToCsv(modSTAR.tmpInpFName$(0), IFERR)
If IFERR = 1 Then GoTo Err1

' CSVファイルを一時的に表示
i = InStrRev(FName$, "¥")
j = Len(FName$) - i - 4
modSTAR.tmpDBConnect$ = "Text;"
modSTAR.tmpDBDatabaseName$ = Left$(FName$, i)
modSTAR.tmpDBRecordSource$ = Mid$(FName$, i + 1, j) & "#CSV"
modSTAR.tmpFormOpenType = vbOKOnly

frmPriViewDBG.Show 1

If modSTAR.tmpFormCloseType = vbCancel Then
    Exit Sub
End If

Else

MsgBox "認識できないファイル(拡張子)です。"
Exit Sub

End If

If StrComp(Right$(FName$, 3), "DXF", 1) = 0 Then 'DXFファイル

' DXFファイルを一時的に表示
modSTAR.tmpDBDatabaseName$ = FName$
modSTAR.tmpFormOpenType = vbOKOnly

frmPriViewDXF.Show 1

If modSTAR.tmpFormCloseType = vbCancel Then
    Exit Sub
End If

' -----

Screen.MousePointer = 11

' 古い形状データを削除しておく
Call DeleteDataR("形状", modSTAR.nmID)

' -----

MaxID = GetMaxID("形状", "id") + 1

' 形状ファイルのオープン
SQL$ = "SELECT * FROM 形状 WHERE (参照id = " & modSTAR.nmID & ");"
Set dtSHAPE = dbSTAR5.OpenResultset(SQL$, rdOpenDynamic, rdConcurLock)
Set dtSHAPE = xOpenResultset(SQL$, rdOpenDynamic, rdConcurLock, hSHAPE)
If hSHAPE = -1 Then GoTo Err1

Call ShapeDxfRead(FName$, MaxID, dtSHAPE, RCount, IFERR)
If IFERR = 1 Then GoTo Err1

```

```

'形状テーブルのクロス
'dtSHAPE.Close
xCloseResultset dtSHAPE, hSHAPE

Screen.MousePointer = 0

'ElseIf StrComp(Right$(FName$, 3), "XLS", 1) = 0 Then 'Excelシート
'Excelシートを一時的に表示
modSTAR.tmpDBConnect$ = "Excel 5.0;"
modSTAR.tmpDBDatabaseName$ = FName$
modSTAR.tmpDBRecordSource$ = TName$
modSTAR.tmpFormOpenType = vbOKOnly

frmPriViewDBG.Show 1

If modSTAR.tmpFormCloseType = vbCancel Then
Exit Sub
End If

Call ShapeExtToDB

'ElseIf StrComp(Right$(FName$, 3), "MDB", 1) = 0 Then 'Access
'Accessテーブルを一時的に表示
modSTAR.tmpDBConnect$ = "Access 2.0;"
modSTAR.tmpDBDatabaseName$ = FName$
modSTAR.tmpDBRecordSource$ = TName$
modSTAR.tmpFormOpenType = vbOKOnly

frmPriViewDBG.Show 1

If modSTAR.tmpFormCloseType = vbCancel Then
Exit Sub
End If

Call ShapeExtToDB

'ElseIf StrComp(Right$(FName$, 3), "CSV", 1) = 0 Then 'CSVファイル
'CSVファイルを一時的に表示
i = InStrRev(FName$, "#")
j = Len(FName$) - i - 4
modSTAR.tmpDBConnect$ = "Text;"
modSTAR.tmpDBDatabaseName$ = Left$(FName$, i)
modSTAR.tmpDBRecordSource$ = Mid$(FName$, i + 1, j) & "#CSV"
modSTAR.tmpFormOpenType = vbOKOnly

frmPriViewDBG.Show 1

If modSTAR.tmpFormCloseType = vbCancel Then
Exit Sub
End If

Call ShapeExtToDB

'ElseIf StrComp(Right$(FName$, 3), "STF", 1) = 0 Then 'STFファイル
'STFファイルをCSV形式に変換
modSTAR.tmpInoFName$(0) = FName$
Call ShapeStfToCsv

'CSVファイルを一時的に表示
i = InStrRev(FName$, "#")
j = Len(FName$) - i - 4
modSTAR.tmpDBConnect$ = "Text;"
modSTAR.tmpDBDatabaseName$ = Left$(FName$, i)
modSTAR.tmpDBRecordSource$ = Mid$(FName$, i + 1, j) & "#CSV"
modSTAR.tmpFormOpenType = vbOKOnly

frmPriViewDBG.Show 1

If modSTAR.tmpFormCloseType = vbCancel Then
Exit Sub
End If

Call ShapeExtToDB

Else
MsgBox "認識できないファイル(拡張子)です。"
Exit Sub

Else
'XLS/MDB/CSV/STFファイル

Screen.MousePointer = 11

''古い形状テーブルを削除しておく
''SQL$ = "DELETE 形状 WHERE (参照id = " & modSTAR.nmID & ");"
'dbSTAR5.Execute SQL$
'Call DeletedDataR("形状", "参照id", modSTAR.nmID)

'外部ファイルをオープン
Set dbEX_SHAPE = OpenDatabase(modSTAR.tmpDBDatabaseName$, False, False, modSTAR.tmpDBConnect$)
Set dtEX_SHAPE = dbEX_SHAPE.OpenRecordset(modSTAR.tmpDBRecordSource$, dbOpenSnapshot)

'入力ファイルのファイル名チェック

```



```

Call InputFieldCheck(31, dtEX_SHAPE, IFERR)
If IFERR > 0 Then GoTo Err1
' If IFERR = 1 Then
' dtEX_SHAPE.Close
' dbEX_SHAPE.Close
' Screen.MousePointer = 0
' Exit Sub
' End If

'古い形状データを削除しておく
Call DeleteData("形状", modSTAR.nmID)

'形状テーブルのオープン
SQL$ = "SELECT * FROM 形状 WHERE (参照id = " & modSTAR.nmID & ");"
Set dtSHAPE = dbSTAR5.OpenResultset(SQL$, rdOpenDynamic, rdConcurLock)
Set dtSHAPE = xOpenResultset(SQL$, rdOpenDynamic, rdConcurLock, hSHAPE)
If hSHAPE = -1 Then GoTo Err1

MaxID = GetMaxID("形状", "id")
MaxIDO = MaxID

'形状テーブルのコピー
If dtEX_SHAPE.RecordCount > 0 Then
Do Until dtEX_SHAPE.EOF
dtSHAPE.AddNew
MaxID = MaxID + 1
dtSHAPE![id] = MaxID
dtSHAPE![参照id] = modSTAR.nmID
dtSHAPE![起点r1] = dtEX_SHAPE.Fields(0).Value
dtSHAPE![起点z1] = dtEX_SHAPE.Fields(1).Value
dtEX_SHAPE.MoveNext
dtSHAPE![終点r2] = dtEX_SHAPE.Fields(0).Value
dtSHAPE![終点z2] = dtEX_SHAPE.Fields(1).Value
dtEX_SHAPE.MoveNext
dtSHAPE.Update
Loop
RCount = MaxID - MaxIDO
End If

'形状テーブルのクローズ
dtSHAPE.Close
xCloseResultset dtSHAPE, hSHAPE

'外部ファイルをクローズ
dtEX_SHAPE.Close
dbEX_SHAPE.Close

'Call SetKanriXID("形状id", modSTAR.nmID, modSTAR.nmID, "-", 0)
Call SetKanriXID6("形状id", modSTAR.nmID, modSTAR.nmID, "-", 0)

Screen.MousePointer = 0

End If

MsgBox RCount & " 行をコピーしました。"

Exit Sub

Err1:
Screen.MousePointer = 0
MsgBox "Error in mnuPriShapeReg"
Call ShowErrorMessage
On Error Resume Next
dtSHAPE.Close
xCloseResultset dtSHAPE, hSHAPE
dtEX_SHAPE.Close
dbEX_SHAPE.Close

End Sub

Private Sub mnuPriShapeShow_Click()
Dim frmGRD As New frmPriViewGRD
Dim frmOLE As New frmPriViewOLE
Dim dtTMP As Recordset
Dim SQL$, SQLW$, CName$, QName$
Dim ResType As Long
Dim j As Long

On Error GoTo Err1

QName$ = "形状データ"
SQLW$ = " WHERE (参照id = " & modSTAR.nmID & ");"
SQL$ = "SELECT * FROM 形状" & SQLW$

ResType = 2
' 1:DBGrid
' 2:Grid
' 3:Excel(OLE)
' 4:Excel(ファイル)

If ResType = 1 Then
' DBGridコントロールに表示
'形状テーブルを表示
modSTAR.tmpFormName$ = QName$

```

```

modSTAR.tmpDBConnect$ = "Access"
modSTAR.tmpDBDatabaseName$ = modSTAR.StarDB$
modSTAR.tmpDBRecordSource$ = SQL$
modSTAR.tmpFormOpenType = vbOKCancel

frmPriViewDBG.Show 1

ElseIf ResType = 2 Then          'Gridコントロールに表示

'形状フォームを表示
modSTAR.tmpFormName$ = QName$
modSTAR.tmpViewSQL$ = SQL$
modSTAR.tmpDeleteSQL$ = "DELETE 形状" & SQLWS

'frmPriViewGRD.Show
frmGRD.Show

ElseIf ResType = 3 Then          'Excel(OLE)シートに表示

ElseIf ResType = 4 Then          'Excel(ファイル)シートに表示

'QueryDefから検索
Set dtTMP = dbSTAR.OpenRecordset(SQL$, dbOpenSnapshot)

'結果をCSVファイルに格納
If dtTMP.RecordCount > 0 Then

'結果表示用の子フォームをオープン
frmOLE.Show
frmStarMain.ActiveForm.Caption = QName$

'検索結果をテキストファイルに出力
modSTAR.tmpViewNo = modSTAR.tmpViewNo + 1
CName$ = modSTAR.WorkDir$ & "$VIEW" & modSTAR.tmpViewNo & ".XLS"
Open CName$ For Output As #1
For j = 0 To dtTMP.Fields.Count - 1
.Print #1, dtTMP.Fields(j).Name; Chr$(9);
Next j
Print #1, Chr$(13); Chr$(10);
Do Until dtTMP.EOF
For j = 0 To dtTMP.Fields.Count - 1
If IsNull(dtTMP.Fields(j).Value) = True Then
Debug.Print "NULL"
Print #1, Chr$(9);
Else
Print #1, dtTMP.Fields(j).Value; Chr$(9);
End If
Next j
Print #1, Chr$(13); Chr$(10);
dtTMP.MoveNext
Loop
Close #1

modSTAR.tmpViewSQL$ = SQL$
modSTAR.tmpViewTableName$ = "形状"
modSTAR.tmpViewTableRow = dtTMP.RecordCount
modSTAR.tmpViewTableCol = dtTMP.Fields.Count

'Excelを起動し、テキストファイルをオープン
frmStarMain.ActiveForm.oleView.CreateEmbed CName$, "Excel.Sheet.5"
frmStarMain.ActiveForm.oleView.CreateLink CName$

End If

dtTMP.Close

End If

Exit Sub

Err1:
MsgBox "Error in mnuPriShapeShow"
Call ShowErrorMessage

End Sub

Private Sub mnuPriTempReg_Click()
Dim dbEX_TEMP As Database
Dim dtEX_TEMP As Recordset
Dim dtTEMP As rdoResultset
Dim FName$, TName$, SQL$
Dim i As Long
Dim j As Long
Dim OpenDialogType As Long
Dim MaxID As Long
Dim MaxIDO As Long
Dim IFERR As Integer
Dim hTEMP As Integer

On Error GoTo Err1

modSTAR.tmpFormName$ = "温度データの入力"

```

```

=====
OpenDialogType = 2                '1:モンタ'イロク'
                                   '2:X_SELDB1'イロク'

If OpenDialogType = 1 Then        'モンタ'イロク'使用
frmStarMain.CommonDialog1.Filter = "Excel 5.0 (*.XLS)|*.xls|" -
& "CSVファイル (*.CSV)|*.csv|" -
& "標準テキスト (*.STF)|*.stf" -
frmStarMain.CommonDialog1.Flags = cd1OFNReadOnly

On Error Resume Next
frmStarMain.CommonDialog1.ShowOpen
On Error GoTo 0

'復帰ボタンのシテック
If Err.Number <> 0 Then
If Err.Number = 32755 Then 'ユーザ'がキャンセルを選択
Exit Sub
Else
MsgBox "[ファイルを開く]タ'イロク'ボックスで予期しないエラーが発生しました。"
Exit Sub
End If
End If

'入力ファイル名の取得
FName$ = frmStarMain.CommonDialog1.FileName
TName$ = "Database"

ElseIf OpenDialogType = 2 Then    'X_SELDB1'イロク'使用
modSTAR.tmpInpFilter = "Excel 8.0 (*.XLS)|" -
& "CSVファイル (*.CSV)|" -
& "標準テキスト (*.STF)|" -
frmSelDB1.Show 1

If modSTAR.tmpFormCloseType = vbCancel Then Exit Sub

'入力ファイル名の取得
FName$ = modSTAR.tmpInpFName$(0)
TName$ = modSTAR.tmpInpTName$(0)

End If

=====
If StrComp(Right$(FName$, 3), "XLS", 1) = 0 Then 'Excelシート

'Excelシートを一時的に表示
modSTAR.tmpDBConnect$ = "Excel 8.0;"
modSTAR.tmpDBDatabaseName$ = FName$
modSTAR.tmpDBRecordSource$ = TName$
modSTAR.tmpFormOpenType = vbOKOnly

frmPriViewDBG.Show 1

If modSTAR.tmpFormCloseType = vbCancel Then
Exit Sub
End If

ElseIf StrComp(Right$(FName$, 3), "MDB", 1) = 0 Then 'Access

'Accessデータベースを一時的に表示
modSTAR.tmpDBConnect$ = "Access 2.0;"
modSTAR.tmpDBDatabaseName$ = FName$
modSTAR.tmpDBRecordSource$ = TName$
modSTAR.tmpFormOpenType = vbOKOnly

frmPriViewDBG.Show 1

If modSTAR.tmpFormCloseType = vbCancel Then
Exit Sub
End If

ElseIf StrComp(Right$(FName$, 3), "CSV", 1) = 0 Then 'CSVファイル

'CSVファイルを一時的に表示
i = InStrRev(FName$, "#")          ' #の位置
j = Len(FName$) - i - 4           ' 拡張子を除くファイル名の長さ
modSTAR.tmpDBConnect$ = "Text;"
modSTAR.tmpDBDatabaseName$ = Left$(FName$, i)
modSTAR.tmpDBRecordSource$ = Mid$(FName$, i + 1, j) & "#CSV"
modSTAR.tmpFormOpenType = vbOKOnly

frmPriViewDBG.Show 1

If modSTAR.tmpFormCloseType = vbCancel Then
Exit Sub
End If

ElseIf StrComp(Right$(FName$, 3), "STF", 1) = 0 Then '標準テキスト

'STFファイルをCSV形式に変換
modSTAR.tmpInpFName$(0) = FName$
'Call TempStfToCsv
Call StfToCsv(modSTAR.tmpInpFName$(0), IFERR)
If IFERR = 1 Then GoTo Err!

```

```

' CSVファイルを一時的に表示
i = InStrRev(FName$, ",") ' #の位置
j = Len(FName$) - i - 4 ' 拡張子を除くファイル名の長さ
modSTAR.tmpDBConnect$ = "Text;"
modSTAR.tmpDBDatabaseName$ = Left$(FName$, i)
modSTAR.tmpDBRecordSource$ = Mid$(FName$, i + 1, j) & "#CSV"
modSTAR.tmpFormOpenType = vbOKOnly

frmPriViewDBG.Show 1

If modSTAR.tmpFormCloseType = vbCancel Then
    Exit Sub
End If

Else
    MsgBox "認識できないファイル(拡張子)です。"
    Exit Sub
End If

' =====

Screen.MousePointer = 11

' 古い温度データを削除しておく
Call DeleteData("温度", "参照id", modSTAR.nmID)

' Excelシートのオープン
Set dbEX_TEMP = OpenDatabase(modSTAR.tmpDBDatabaseName$, False, False, modSTAR.tmpDBConnect$)
Set dtEX_TEMP = dbEX_TEMP.OpenRecordset(modSTAR.tmpDBRecordSource$, dbOpenSnapshot)

' 入力ファイルのフィールド名チェック
Call InputFieldCheck(21, dtEX_TEMP, IFERR)
If IFERR > 0 Then GoTo Err1
' If IFERR = 1 Then
'     dtEX_TEMP.Close
'     dbEX_TEMP.Close
'     Screen.MousePointer = 0
'     Exit Sub
' End If

' 古い温度データを削除しておく
Call DeleteData("温度", modSTAR.nmID)

' 温度データのオープン
SQL$ = "SELECT * FROM 温度 WHERE ("
    & "(参照id = " & modSTAR.nmID & ")");"
Set dtTEMP = dbSTAR5.OpenResultset(SQL$, rdOpenDynamic, rdConcurLock)
Set hTEMP = xOpenResultset(SQL$, rdOpenDynamic, rdConcurLock, hTEMP)
If hTEMP = -1 Then GoTo Err1

MaxID = GetMaxID("温度", "id")
MaxID0 = MaxID

If dtEX_TEMP.RecordCount > 0 Then
    i = 0
    Do Until dtEX_TEMP.EOF
        dtTEMP.AddNew
        MaxID = MaxID + 1
        dtTEMP![id] = MaxID
        dtTEMP![参照id] = modSTAR.nmID
        For j = 0 To dtEX_TEMP.Fields.Count - 1
            dtTEMP.rdcColumns(j + 2).Value = dtEX_TEMP.Fields(j).Value
        Next j
        dtTEMP.Update
        dtEX_TEMP.MoveNext
        i = i + 1
        Debug.Print i
        Debug.Print MaxID - MaxID0
    Loop
End If

dtTEMP.Close ' 温度データのクローズ
xCloseResultset dtTEMP, hTEMP ' 温度データのクローズ

dtEX_TEMP.Close ' Excelシートのクローズ
dbEX_TEMP.Close ' Excelシートのクローズ

' Call SetKanriXID("温度id", modSTAR.nmID, modSTAR.nmID, "-", 0)
' Call SetKanriXID6("温度id", modSTAR.nmID, modSTAR.nmID, "-", 0)

Screen.MousePointer = 0

MsgBox (MaxID - MaxID0) & " 行を蓄込みました。"

Exit Sub

Err1:
Screen.MousePointer = 0
MsgBox "Error in mnuPriTempReg"
Call ShowErrorMessage
On Error Resume Next
dtTEMP.Close
xCloseResultset dtTEMP, hTEMP
dtEX_TEMP.Close

```

```

dbEX_TEMP.Close
End Sub

Private Sub mnuPriTempShow_Click()
Dim frmGRD As New frmPriViewGRD
Dim frmOLE As New frmPriViewOLE
Dim dtTMP As Recordset
Dim SQL$, SQLWS, CName$, QName$
Dim ResType As Long
Dim j As Long

On Error GoTo Err1

QName$ = "温度データ"
SQLWS$ = " WHERE (参照id = " & modSTAR.nmID & ");"
SQL$ = "SELECT * FROM 温度" & SQLWS

ResType = 2          ' 1:DBGrid
                    ' 2:Grid
                    ' 3:Excel(OLE)
                    ' 4:Excel(ファイル)

If ResType = 1 Then          ' DBGridコントロールに表示

' 温度フォームを表示
modSTAR.tmpFormName$ = QName$
modSTAR.tmpDBCConnect$ = "Access"
modSTAR.tmpDBDatabaseName$ = modSTAR.StarDB$
modSTAR.tmpDBRecordSource$ = SQL$
modSTAR.tmpFormOpenType = vbOKCancel

frmPriViewDBG.Show 1

ElseIf ResType = 2 Then          ' Gridコントロールに表示

' 温度フォームを表示
modSTAR.tmpFormName$ = QName$
modSTAR.tmpViewSQL$ = SQL$
modSTAR.tmpDeleteSQL$ = "DELETE 温度" & SQLWS

' frmPriViewGRD.Show
frmGRD.Show

ElseIf ResType = 3 Then          ' Excel(OLE)シートに表示

ElseIf ResType = 4 Then          ' Excel(ファイル)シートに表示

' QueryDefから検索
Set dtTMP = dbSTAR.OpenRecordset(SQL$, dbOpenSnapshot)

' 結果をCSVファイルに格納
If dtTMP.RecordCount > 0 Then

' 結果表示用の子フォームをオープン
frmOLE.Show
frmStarMain.ActiveForm.Caption = QName$

' 検索結果をテキストファイルに出力
modSTAR.tmpViewNo = modSTAR.tmpViewNo + 1
CName$ = modSTAR.WorkDir$ & "$VIEW" & modSTAR.tmpViewNo & ".XLS"
Open CName$ For Output As #1
For j = 0 To dtTMP.Fields.Count - 1
Print #1, dtTMP.Fields(j).Name; Chr$(9);
Next j
Print #1, Chr$(13); Chr$(10);
Do Until dtTMP.EOF
For j = 0 To dtTMP.Fields.Count - 1
If IsNull(dtTMP.Fields(j).Value) = True Then
Debug.Print "NULL"
Print #1, Chr$(9);
Else
Print #1, dtTMP.Fields(j).Value; Chr$(9);
End If
Next j
Print #1, Chr$(13); Chr$(10);
dtTMP.MoveNext
Loop
Close #1

modSTAR.tmpViewSQL$ = SQL$
modSTAR.tmpViewTableName$ = "温度"
modSTAR.tmpViewTableRow = dtTMP.RecordCount
modSTAR.tmpViewTableCol = dtTMP.Fields.Count

' Excelを起動し、テキストファイルをオープン
frmStarMain.ActiveForm.oleView.CreateEmbed CName$, "Excel.Sheet.5"
frmStarMain.ActiveForm.oleView.CreateLink CName$

End If

dtTMP.Close

```

```
End If
Exit Sub
```

```
Err1:
MsgBox "Error in mnuPriTempShow"
Call ShowErrorMessage
End Sub
```

```
Private Sub mnuSecDamageAll_Click()
modSTAR.tmpSecDamageSel = 1
frmSecDmgCtrl.Show 1
End Sub
```

```
Private Sub mnuSecDamageCase_Click()
modSTAR.tmpSecDamageSel = 3
frmSecDmgCtrl.Show 1
End Sub
```

```
Private Sub mnuSecDamageSel_Click()
frmSecDmgSel.Show 1
End Sub
```

```
Private Sub mnuSecDamageShow_Click()
Dim frmGRD As New frmPriViewGRD
Dim frmOLE As New frmPriViewOLE
Dim dtTMP As Recordset
Dim SQL$, SQLW$, CName$, QName$
Dim ResType As Long
Dim j As Long

On Error GoTo Err1

frmDmgViewSel.Show 1

If modSTAR.tmpFormCloseType = vbCancel Then Exit Sub

QName$ = "損傷値データ (" & modSTAR.CurrentCase$ & " - " & modSTAR.tmpViewDamageName$ & ")"
SQLW$ = " WHERE (解析id = " & modSTAR.nmAID & ");"
SQL$ = "SELECT * FROM " & modSTAR.tmpViewDamageName$ & SQLW$

ResType = 2          '1:DBGrid
                   '2:Grid
                   '3:Excel(OLE)
                   '4:Excel(ファイル)

If ResType = 1 Then   'DBGridコントロールに表示

'表計算フォームを表示
modSTAR.tmpFormName$ = QName$
modSTAR.tmpDBConnect$ = "Access"
modSTAR.tmpDBDatabaseName$ = modSTAR.StarDB$
modSTAR.tmpDBRecordSource$ = SQL$
modSTAR.tmpFormOpenType = vbOKCancel

frmPriViewDBG.Show 1

ElseIf ResType = 2 Then   'Gridコントロールに表示

'表計算フォームを表示
modSTAR.tmpFormName$ = QName$
modSTAR.tmpViewSQL$ = SQL$
modSTAR.tmpDeleteSQL$ = "DELETE " & modSTAR.tmpViewDamageName$ & SQLW$

'frmPriViewGRD.Show
frmGRD.Show

ElseIf ResType = 3 Then   'Excel(OLE)シートに表示

ElseIf ResType = 4 Then   'Excel(ファイル)シートに表示

'QueryDefから検索
Set dtTMP = dbSTAR.OpenRecordset(SQL$, dbOpenSnapshot)

'結果をCSVファイルに格納
If dtTMP.RecordCount > 0 Then

'結果表示用の子フォームをオープン
frmOLE.Show
```

```

frmStarMain.ActiveForm.Caption = QName$

' 検索結果をテキストファイルに出力
modSTAR.tmpViewNo = modSTAR.tmpViewNo + 1
CName$ = modSTAR.WorkDir$ & "$VIEW" & modSTAR.tmpViewNo & ".XLS"
Open CName$ For Output As #1
For j = 0 To dtTMP.Fields.Count - 1
    Print #1, dtTMP.Fields(j).Name; Chr$(9);
Next j
Print #1, Chr$(13); Chr$(10);
Do Until dtTMP.EOF
    For j = 0 To dtTMP.Fields.Count - 1
        If IsNull(dtTMP.Fields(j).Value) = True Then
            Debug.Print "NULL"
            Print #1, Chr$(9);
        Else
            Print #1, dtTMP.Fields(j).Value; Chr$(9);
        End If
    Next j
    Print #1, Chr$(13); Chr$(10);
    dtTMP.MoveNext
Loop
Close #1

modSTAR.tmpViewSQL$ = SQL$
modSTAR.tmpViewTableName$ = "表計算"
modSTAR.tmpViewTableRow = dtTMP.RecordCount
modSTAR.tmpViewTableCol = dtTMP.Fields.Count

' Excelを起動し、テキストファイルをオープン
frmStarMain.ActiveForm.oleView.CreateEmbed CName$, "Excel.Sheet.5"
frmStarMain.ActiveForm.oleView.CreateLink CName$

End If

dtTMP.Close

End If

Exit Sub

Err1:
MsgBox "Error in mnuSecDamageShow"
Call ShowErrorMessage

End Sub

Private Sub mnuSecDamageTai_Click()
    modSTAR.tmpSecDamageSel = 2
    frmSecDmgCtrl.Show 1
End Sub

Private Sub mnuSecFinasReg_Click()
    modSTAR.tmpFormOpenType = vbOKCancel
    frmSecFinasView.Show 1
End Sub

Private Sub mnuSecFinasShow_Click()
    modSTAR.tmpFormOpenType = vbOKOnly
    frmSecFinasView.Show 1
End Sub

Private Sub mnuSecQCalc_Click()
    Dim dtKANRI As rdoResultset
    Dim SQL$
    Dim iRec As Long
    Dim hKANRI As Integer

    On Error GoTo Err1

    iRec = 0
    Screen.MousePointer = 11

    SQL$ = "SELECT 解析ケース.評価法id FROM 管理データベース WHERE (" & _
        & "(試験装置 = '" & modSTAR.nmSochi$ & "') AND " & _
        & "(試験体 = '" & modSTAR.nmTai$ & "') AND " & _
        & "(試験部分 = '" & modSTAR.nmBubun$ & "') AND " & _
        & "(解析ケース LIKE 'ケース%') AND " & _
        & "(評価法id = 0));"
    Debug.Print SQL$

    'Set dtKANRI = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
    Set dtKANRI = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hKANRI)
    If hKANRI = -1 Then GoTo Err1

    Debug.Print dtKANRI.EOF

```

```

If dtKANRI.EOF = False Then
    dtKANRI.MoveLast
    Debug.Print dtKANRI.RowCount
    'If dtKANRI.RowCount >= 2 Then
    '    frmSecQCalcSel.Show 1          'ケースの選択フォームの表示
    '    dtKANRI.Close
    '    Exit Sub
    'End If
    iRec = dtKANRI.RowCount
End If

'dtKANRI.Close
xCloseResultset dtKANRI, hKANRI

Screen.MousePointer = 0

If iRec >= 2 Then
    frmSecQCalcSel.Show 1          'ケースの選択フォームの表示
Else
    MsgBox "登録されている解析ケースが不足しています。"
End If

'dtKANRI.Close

Exit Sub

Err1:
Screen.MousePointer = 0
MsgBox "Error in mnuSecQCalc"
Call ShowErrorMessage
On Error Resume Next
'dtKANRI.Close
xCloseResultset dtKANRI, hKANRI

End Sub

Private Sub mnuSecTableAll_Click()
    modSTAR.tmpSecTableSel = 1
    frmSecTblCtrl.Show 1

End Sub

Private Sub mnuSecTableCase_Click()
    modSTAR.tmpSecTableSel = 3
    frmSecTblCtrl.Show 1

End Sub

Private Sub mnuSecTableShow_Click()
    Dim frmGRD As New frmPriviewGRD
    Dim frmOLE As New frmPriviewOLE
    Dim dtTMP As Recordset
    Dim SQL$, SQLW$, CName$, QName$
    Dim ResType As Long
    Dim j As Long

    On Error GoTo Err1

    QName$ = "表計算データ(" & modSTAR.CurrentCase$ & ")"
    SQLW$ = " WHERE (解析id = " & modSTAR.nmAID & ");"
    SQL$ = "SELECT * FROM 表計算" & SQLW$

    ResType = 2
    '1:DBGrid
    '2:Grid
    '3:Excel(OLE)
    '4:Excel(777)

    If ResType = 1 Then          'DBGridコントロールに表示
        '表計算フォームを表示
        modSTAR.tmpFormName$ = QName$
        modSTAR.tmpDBConnect$ = "Access"
        modSTAR.tmpDBDatabaseName$ = modSTAR.StarDBS
        modSTAR.tmpDBRecordSource$ = SQL$
        modSTAR.tmpFormOpenType = vbOKCancel

        frmPriviewDBG.Show 1
    ElseIf ResType = 2 Then    'Gridコントロールに表示
        '表計算フォームを表示
        modSTAR.tmpFormName$ = QName$
        modSTAR.tmpViewSQL$ = SQL$
        modSTAR.tmpDeleteSQL$ = "DELETE 表計算" & SQLW$

```



```

' frmPriViewGRD.Show
frmGRD.Show

ElseIf ResType = 3 Then          ' Excel(OLE)シートに表示

ElseIf ResType = 4 Then        ' Excel(ファイル)シートに表示

' QueryDefから検索
Set dtTMP = dbSTAR.OpenRecordset(SQL$, dbOpenSnapshot)
' 結果をCSVファイルに格納
If dtTMP.RecordCount > 0 Then
    ' 結果表示用の子フォームをオープン
    frmOLE.Show
    frmStarMain.ActiveForm.Caption = QName$

    ' 検索結果をテキストファイルに出力
    modSTAR.tmpViewNo = modSTAR.tmpViewNo + 1
    CName$ = modSTAR.WorkDir$ & "$VIEW" & modSTAR.tmpViewNo & ".XLS"
    Open CName$ For Output As #1
    For j = 0 To dtTMP.Fields.Count - 1
        Print #1, dtTMP.Fields(j).Name; Chr$(9);
    Next j
    Print #1, Chr$(13); Chr$(10);
    Do Until dtTMP.EOF
        For j = 0 To dtTMP.Fields.Count - 1
            If IsNull(dtTMP.Fields(j).Value) = True Then
                Debug.Print "NULL"
                Print #1, Chr$(9);
            Else
                Print #1, dtTMP.Fields(j).Value; Chr$(9);
            End If
        Next j
        Print #1, Chr$(13); Chr$(10);
        dtTMP.MoveNext
    Loop
    Close #1

    modSTAR.tmpViewSQL$ = SQL$
    modSTAR.tmpViewTableName$ = "表計算"
    modSTAR.tmpViewTableRow = dtTMP.RecordCount
    modSTAR.tmpViewTableCol = dtTMP.Fields.Count

    ' Excelを起動し、テキストファイルをオープン
    ' frmStarMain.ActiveForm.oleView.CreateEmbed CName$, "Excel.Sheet.5"
    frmStarMain.ActiveForm.oleView.CreateLink CName$

End If

dtTMP.Close

End If

Exit Sub

Err1:
MsgBox "Error in mnuSecTableShow"
Call ShowErrorMessage

End Sub

Private Sub mnuSecTableTai_Click()
    modSTAR.tmpSecTableSel = 2
    frmSecTblCtrl.Show 1
End Sub

Private Sub mnuSelBubun_Click()
    frmSelBubun2.Show 1
End Sub

Private Sub mnuSelCase_Click()
    frmSelCase2.Show 1
End Sub

Private Sub mnuSelExit_Click()
    On Error Resume Next

    Screen.MousePointer = 11
    dbSTAR5.Close
    ' Close #9

```

```

Screen.MousePointer = 0
End

End Sub

Private Sub mnuSelShape_Click()
Dim i As Long
Dim Ret As Variant
Dim MName$, DName$
Dim FP As Long
Dim NBS As Long, NBC As Long, NBL As Long, NBD As Long

On Error GoTo Err1          ' Err

' modSTAR.WorkDir$: 作業ファイルを作成するディレクトリ名
' modSTAR.NoOfLine: 試験部分の評価ラインの本数
' NBS: BLOCKSセクションに書込んだ形状データの数(通常は1)
' NBC: BLOCKSセクションに書込んだき裂データの数(通常は1)
' NBL: BLOCKSセクションに書込んだ評価ラインデータの数(通常はNoOfLineと同じ)
' NBD: BLOCKSセクションに書込んだ評価断面データの数(通常はNoOfLineと同じ)

Screen.MousePointer = 11

DName$ = modSTAR.WorkDir$ & "$SHAPE.DXF"
MName$ = modSTAR.WorkDir$ & "$SHAPE.MCR"

' DXF, MCRファイルを削除しておく
Call KillFile(DName$)
Call KillFile(MName$)

' DXFファイルを開く
FP = 1
Open DName$ For Output As #FP

' TABLESセクションの記述
Call CreateDxfLayer(FP)

' BLOCKSセクションの記述
Print #FP, "0"
Print #FP, "SECTION"
Print #FP, "2"
Print #FP, "BLOCKS"

' 形状データの作成
Call CreateDxfShape(FP, NBS)
If NBS = 0 Then
MsgBox "形状データがありません。"
Print #FP, "0"
Print #FP, "ENDSEC"
Print #FP, "0"
Print #FP, "EOF"
Close #FP
Screen.MousePointer = 0
Exit Sub
End If

' き裂データの作成
Call CreateDxfCrack(FP, NBC)

' 評価ラインデータの作成
Call CreateDxfHLine(FP, modSTAR.NoOfLine, NBL)

' 評価断面データの作成
Call CreateDxfHsect(FP, modSTAR.NoOfLine, NBD)

Print #FP, "0"
Print #FP, "ENDSEC"

' ENTITIESセクションの記述
Print #FP, "0"
Print #FP, "SECTION"
Print #FP, "2"
Print #FP, "ENTITIES"
If NBS > 0 Then Call CreateDxfInsert(FP, "SHAPE")
For i = 1 To NBL
Call CreateDxfInsert(FP, "HLINE" & i)
Next i
For i = 1 To NBD
Call CreateDxfInsert(FP, "HSECT" & i)
Next i
If NBC > 0 Then Call CreateDxfInsert(FP, "CRACK")
Print #FP, "0"
Print #FP, "ENDSEC"

' EOFの記述
Print #FP, "0"
Print #FP, "EOF"

' DXFファイルをクローズ
Close #FP

' マクロファイルの作成
Open MName$ For Output As #1
Print #1, "REM r2.1 - AutoSketch for Windows Release 2.1 Macro File."

```

```

Print #1, "IMPORT"
Print #1, "SET DXFFILENAME " & "$SHAPE"
Print #1, "SET DXFDIR " & modSTAR.WorkDir$
Print #1, "DialogBoxReturn 1"
Print #1, "DialogBoxReturn 1"
Print #1, "ZOOMFULL"
Close #1

Screen.MousePointer = 0

' AutoSketchの起動とマクロの実行
If modSTAR.WSketchDir$ <> "" Then
  Ret = Shell(modSTAR.WSketchDir & "SKETCH.EXE -m " & MName$, vbNormalFocus)
  ' SendKeys "{ENTER}%CP" & MName$ & "{ENTER}", True
  ' Debug.Print "{ENTER}%CP" & MName$ & "{ENTER}"
End If

Exit Sub

Err1:
Screen.MousePointer = 0          ' Err
MsgBox "Error in mnuSelShape"   ' Err
Call ShowErrorMessage          ' Err
On Error Resume Next           ' Err
Close #FP                       ' Err

End Sub

Private Sub mnuSelView_Click()

  frmSelView.Show 1

End Sub

Private Sub mnuTokCrack_Click()
Dim dtCRACK As rdoResultset
Dim dtCRACKS As rdoResultset
Dim SQL$
Dim i As Long
Dim iC As Long
Dim hCRACK As Integer
Dim hCRACKS As Integer

On Error GoTo Err1

Screen.MousePointer = 11

' 古いき裂データを削除しておく
SQL$ = "DELETE き裂 S WHERE (参照id = " & modSTAR.nmID & ");"
dbSTAR5.Execute SQL$
Call DeleteDataR("き裂_S", modSTAR.nmID)

' き裂データの[統計処理フラグ]がTRUEの行を検索
SQL$ = "SELECT * FROM き裂 WHERE ("
  & "(参照id = " & modSTAR.nmID & ") AND " _
  & "(統計処理フラグ = 1));"
' Set dtCRACK = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
Set dtCRACK = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hCRACK)
If hCRACK = -1 Then GoTo Err1

' き裂データからき裂データヘッダを転記
SQL$ = "SELECT * FROM き裂 S WHERE ("
  & "(参照id = " & modSTAR.nmID & ") AND " _
  & "(統計処理フラグ = 1));"
' Set dtCRACKS = dbSTAR5.OpenResultset(SQL$, rdOpenDynamic, rdConcurLock)
Set dtCRACKS = xOpenResultset(SQL$, rdOpenDynamic, rdConcurLock, hCRACKS)
If hCRACKS = -1 Then GoTo Err1

iC = 0
Do Until dtCRACK.EOF
  iC = iC + 1
  dtCRACKS.AddNew
  For i = 0 To dtCRACK.rdoColumns.Count - 1
    dtCRACKS.rdoColumns(i).Value = dtCRACK.rdoColumns(i).Value
  Next i
  dtCRACKS.Update
  dtCRACK.MoveNext
Loop

' dtCRACKS.Close
xCloseResultset dtCRACKS, hCRACKS

' dtCRACK.Close
xCloseResultset dtCRACK, hCRACK

' Call SetKanriXID("き裂id", -modSTAR.nmID, "-", 0)
' Call SetKanriXID("き裂Sid", modSTAR.nmID, modSTAR.nmID, "-", 0)
Call SetKanriXID6("き裂Sid", modSTAR.nmID, modSTAR.nmID, "-", 0)

Screen.MousePointer = 0

MsgBox "統計用のき裂データを " & iC & " 行抽出しました。"

```

```

Exit Sub

Err1:
Screen.MousePointer = 0
MsgBox "Error in mnuTokCrack"
Call ShowErrorMessage
On Error Resume Next
'dtCRACK.Close
'dtCRACKS.Close
xCloseResultset dtCRACK, hCRACK
xCloseResultset dtCRACKS, hCRACKS

End Sub

Private Sub mnuTokDataView_Click()
Dim frmGRD As New frmPriViewGRD

On Error GoTo Err1

frmTokDataSel.Show 1

If modSTAR.tmpFormCloseType = vbOK Then
frmGRD.Show
End If

Exit Sub

Err1:
MsgBox "Error in mnuTokDataView"
Call ShowErrorMessage

End Sub

Private Sub mnuTokExec_Click()
Dim dbQ As Database
Dim dtQ As Recordset
Dim qdQ As QueryDef
Dim frmGRD As New frmResGRD
Dim frmDBG As New frmResDBG
Dim frmOLE As New frmResOLE
Dim objOLE As Object
Dim AFG As Object
Dim QName$, CName$, BigStr$
Dim i As Long
Dim j As Long
Dim Rows As Long
Dim Ret As Long
Dim ResType As Long
Dim UserInput As Variant

On Error GoTo Err1

ResType = 2          '1:DBGrid
                   '2:Grid
                   '3:Excel(OLE)
                   '4:Excel(ファイル)
                   '5:Excel(クリップボード)

If modSTAR.BTView = 0 Then ResType = 2
If modSTAR.BTView = 1 Then ResType = 4

' 検索条件設定用のフォームを表示
frmTokSelect.Show 1

If modSTAR.tmpFormCloseType = vbCancel Then Exit Sub

modSTAR.tmpFormName$ = modSTAR.CurTokQueryName$
QName$ = modSTAR.CurTokQueryName$

' キー用のデータベースをオープン
If modSTAR.StarDB$ <> modSTAR.QueryDB$ Then
Set dbQ = Workspaces(0).OpenDatabase(modSTAR.QueryDB$)
Else
Set dbQ = dbSTAR
End If
Set qdQ = dbQ.QueryDefs(QName$)

' 未定義のパラメータの設定
For i = 0 To qdQ.Parameters.Count - 1
UserInput = InputBox(qdQ.Parameters(i).Name, "パラメータの入力")
If UserInput <> "" Then
qdQ.Parameters(i).Value = UserInput
Else
If modSTAR.StarDB$ <> modSTAR.QueryDB$ Then dbQ.Close
Exit Sub
End If
End For

```

Next i

```
' 検索結果の表示-----
Screen.MousePointer = 11
If ResType = 1 Then          ' 結果をDBGridコントロールに表示
' 結果表示用の子フォームをオープン
modSTAR.CurQuerySQL$ = qdQ.SQL
frmDBG.Show
frmStarMain.ActiveForm.Caption = QName$
ElseIf ResType = 2 Then      ' 結果をGridコントロールに表示
' QueryDefから検索
Set dtQ = qdQ.OpenRecordset()
If dtQ.RecordCount > 0 Then
dtQ.MoveLast
Rows = dtQ.RecordCount
Ret = vbYes
If Rows > modSTAR.MaxRet Then
Ret = MsgBox(Rows & "件検索されました。結果を表示しますか?", vbYesNo)
End If
If Ret = vbYes Then
' 結果表示用の子フォームをオープン
frmGRD.Show
frmStarMain.ActiveForm.Caption = QName$
Set AFG = frmStarMain.ActiveForm.grdRes
' dtQ.MoveLast
' AFG.Rows = dtQ.RecordCount + 2
AFG.Rows = Rows + 2
AFG.Cols = dtQ.Fields.Count + 1
dtQ.MoveFirst
For j = 0 To dtQ.Fields.Count - 1
AFG.ColWidth(j + 1) = 1000          ' デフォルトは630twip
AFG.Col = j + 1
AFG.Row = 0
AFG.Text = dtQ.Fields(j).Name
AFG.Row = 1
AFG.Text = dtQ.Fields(j).Name
Next j
i = 1
Do Until dtQ.EOF
AFG.Row = i + 1
AFG.Col = 0
AFG.Text = i
For j = 0 To dtQ.Fields.Count - 1
AFG.Col = j + 1
AFG.Text = dtQ.Fields(j).Value
Next j
dtQ.MoveNext
i = i + 1
Loop
End If
Else
MsgBox "検索件数は0件でした。"
End If
dtQ.Close
ElseIf ResType = 3 Then      ' 結果をExcel(OLE)シートに表示
' QueryDefから検索
Set dtQ = qdQ.OpenRecordset()
If dtQ.RecordCount > 0 Then
' 結果表示用の子フォームをオープン
frmOLE.Show
frmStarMain.ActiveForm.Caption = QName$
frmStarMain.ActiveForm.oleCtrl.Visible = False
frmStarMain.ActiveForm.oleCtrl.CreateEmbed "", "Excel.Sheet.5"
Set objOLE = frmStarMain.ActiveForm.oleCtrl.object
For j = 0 To dtQ.Fields.Count - 1
objOLE.Cells(1, j + 1).Value = dtQ.Fields(j).Name
Next j
i = 2
Do Until dtQ.EOF
For j = 0 To dtQ.Fields.Count - 1
objOLE.Cells(i, j + 1).Value = dtQ.Fields(j).Value
Next j
dtQ.MoveNext
```

```

    i = i + 1
Loop

Set objOLE = Nothing

frmStarMain.ActiveForm.oleCtrl.Visible = True

Else
MsgBox "検索件数は0件でした。"

End If

dtQ.Close

ElseIf ResType = 4 Then      '結果をExcel(ファイル)シートに表示

'QueryDefから検索
Set dtQ = qdQ.OpenRecordset()

'結果をCSVファイルに格納
If dtQ.RecordCount > 0 Then

'結果表示用の子フォームをオープン
frmOLE.Show
frmStarMain.ActiveForm.Caption = QName$

'検索結果をテキストファイルに出力
modSTAR.tmpTokNo = modSTAR.tmpTokNo + 1
CName$ = modSTAR.WorkDir$ & "$TOK" & modSTAR.tmpTokNo & ".XLS"
Open CName$ For Output As #1
For j = 0 To dtQ.Fields.Count - 1
Print #1, dtQ.Fields(j).Name; Chr$(9);
Next j
Print #1, Chr$(13); Chr$(10);
Do Until dtQ.EOF
For j = 0 To dtQ.Fields.Count - 1
Print #1, dtQ.Fields(j).Value; Chr$(9);
Next j
Print #1, Chr$(13); Chr$(10);
dtQ.MoveNext
Loop
Close #1

'Excelを起動し、テキストファイルをオープン
frmStarMain.ActiveForm.oleCtrl.CreateEmbed CName$, "Excel.Sheet.5"
frmStarMain.ActiveForm.oleCtrl.CreateLink CName$

Else
MsgBox "検索件数は0件でした。"

End If

dtQ.Close

ElseIf ResType = 5 Then      '結果をExcel(Clip)シートに表示

'QueryDefから検索
Set dtQ = qdQ.OpenRecordset()

'結果をCSVファイルに格納
If dtQ.RecordCount > 0 Then

'結果表示用の子フォームをオープン
frmOLE.Show
frmStarMain.ActiveForm.Caption = QName$

'検索結果をテキストファイルに出力
modSTAR.tmpTokNo = modSTAR.tmpTokNo + 1
CName$ = modSTAR.WorkDir$ & "$TOK" & modSTAR.tmpTokNo & ".XLS"
Open CName$ For Output As #1
For j = 0 To dtQ.Fields.Count - 1
Print #1, dtQ.Fields(j).Name; Chr$(9);
Next j
Print #1, Chr$(13); Chr$(10);
Do Until dtQ.EOF
For j = 0 To dtQ.Fields.Count - 1
Print #1, dtQ.Fields(j).Value; Chr$(9);
Next j
Print #1, Chr$(13); Chr$(10);
dtQ.MoveNext
Loop
Close #1

'ファイルを文字列変数に読み込む
Open CName$ For Input As #1
BigStr$ = Input(FileLen(CName$), #1)
Close #1

'Excelを起動
frmStarMain.ActiveForm.oleCtrl.CreateEmbed "", "Excel.Sheet.5"
frmStarMain.ActiveForm.oleCtrl.CreateLink
Set objOLE = frmStarMain.ActiveForm.oleCtrl.object

'データをクリップボードへ転送
Clipboard.SetText BigStr$, vbCFText
objOLE.Range("A1").Select
objOLE.ActiveSheet.Paste

```

```

'Excelワークシートを開放
Set objOLE = Nothing

Else
  MsgBox "検索件数は0件でした。"

End If

dtQ.Close

End If

If modSTAR.StarDB$ <> modSTAR.QueryDB$ Then
  dbQ.Close
End If

Screen.MousePointer = 0

Exit Sub

Err1:
Screen.MousePointer = 0
MsgBox "Error in mnuTokExec"
Call ShowErrorMessage
On Error Resume Next
Close #1
dtQ.Close
Set objOLE = Nothing

End Sub

Private Sub mnuTokInterAll_Click()
  modSTAR.tmpTokInterSel = 1
  frmTokInterSel.Show 1
End Sub

Private Sub mnuTokInterCase_Click()
  modSTAR.tmpTokInterSel = 3
  frmTokInterSel.Show 1
End Sub

Private Sub mnuTokInterTai_Click()
  modSTAR.tmpTokInterSel = 2
  frmTokInterSel.Show 1
End Sub

Private Sub mnuUtilCase_Click()
  frmUtilCaseReg.Show 1
End Sub

Private Sub mnuUtilCompress_Click()
  Dim OldMDB$, NewMDB$
  Dim SV As Boolean

  Exit Sub

  Screen.MousePointer = 11

  'STAR.MDBを一度クローズ
  dbSTAR.Close

  -----
  OldMDB$ = modSTAR.DBDir$ & "$STAR.MDB"
  NewMDB$ = modSTAR.DBDir$ & "STAR.MDB"

  '古いバックアップを削除しておく
  Call KillFile(OldMDB$)

  'STAR.MDBのバックアップを作成
  Name NewMDB$ As OldMDB$

  'STAR.MDBの最適化
  DBEngine.CompactDatabase OldMDB$, NewMDB$

  '管理データベースを再度オープン
  SV = modSTAR.SuperVisor
  Set dbSTAR = Workspaces(0).OpenDatabase(StarDB$, SV, Not (SV))

  Screen.MousePointer = 0

End Sub

```

```

Private Sub mnuUtilHyoka_Click()
    frmUtilHyokaReg.Show 1
End Sub

Private Sub mnuUtilExit_Click()
    On Error Resume Next

    Screen.MousePointer = 11
    dbSTAR5.Close
    'Close #1
    Screen.MousePointer = 0
End

End Sub

Private Sub mnuUtilKanri_Click()
    frmUtilKanriReg2.Show 1
End Sub

Private Sub mnuUtilMat_Click()
    frmUtilMatReg.Show 1
End Sub

Private Sub mnuUtilPassword_Click()
    Exit Sub

    'frmUtilPassword.Show 1
End Sub

Private Sub mnuUtilSetup_Click()
    frmUtilSetup.Show 1
End Sub

Private Sub mnuUtilSQL_Click()
    frmUtilSQL.Show 1
End Sub

Private Sub mnuUtilView_Click()
    Dim frmGRD As New frmPriViewGRD
    Dim QName$, SQL$, SQLW$
    Dim ResType As Long
    Dim IFERR As Integer

    On Error GoTo Err1

    ResType = 2
        '1:DBGrid
        '2:Grid
        '3:Excel(OLE)
        '4:Excel(ファイル)

    If ResType = 1 Then
        frmUtilKanriView.Show 1
    ElseIf ResType = 2 Then
        QName$ = "管理データベースの一覧"
        SQL$ = "SELECT * FROM 管理データベース;"

        '管理データベースを表示
        modSTAR.tmpFormName$ = QName$
        modSTAR.tmpViewSQL$ = SQL$
        modSTAR.tmpDeleteSQL$ = "DELETE 管理データベース;"

        frmGRD.Show

        '管理データベースの再読み込み
        Call KanriRead(IFERR)
    End If

    Exit Sub

Err1:
    MsgBox "Error in mnuUtilView"
    Call ShowErrorMessage

End Sub

```



```
Attribute VB_Name = "frmResGRD"  
Attribute VB_GlobalNameSpace = False  
Attribute VB_Creatable = False  
Attribute VB_PredeclaredId = True  
Attribute VB_Exposed = False  
Option Explicit
```

```
Private Sub Form_Resize()
```

```
    grdRes.Width = ScaleWidth  
    grdRes.Height = ScaleHeight
```

```
End Sub
```

```
Private Sub grdRes_KeyDown(KeyCode As Integer, Shift As Integer)
```

```
    If KeyCode = 67 And Shift = 2 Then 'Ctrl+C  
        Clipboard.SetText grdRes.Clip, vbCFText 'クリップボードへ  
    End If
```

```
End Sub
```

```
Private Sub grdRes_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
```

```
    If Button = 2 Then grdRes.ShowWhatsThis
```

```
End Sub
```

```
Private Sub mnuHelp_Click()
```

```
    CommonDialog1.HelpCommand = cdHelpContext  
    CommonDialog1.HelpFile = App.HelpFile  
    CommonDialog1.ShowHelp  
End Sub
```

```
Private Sub mnuResClose_Click()
```

```
    Unload frmStarMain.ActiveForm  
End Sub
```

```
Private Sub mnuResEditAll_Click()
```

```
    grdRes.SelStartRow = 1  
    grdRes.SelStartCol = 1  
    grdRes.SelEndRow = grdRes.Rows - 1  
    grdRes.SelEndCol = grdRes.Cols - 1  
End Sub
```

```
Private Sub mnuResEditCopy_Click()
```

```
    Clipboard.SetText grdRes.Clip, vbCFText  
End Sub
```

```
Attribute VB_Name = "frmResOLE"  
Attribute VB_GlobalNameSpace = False  
Attribute VB_Creatable = False  
Attribute VB_PredeclaredId = True  
Attribute VB_Exposed = False  
Option Explicit
```

```
Private Sub Form_Resize()  
    frmStarMain.ActiveForm.ScaleMode = 2 ' 絶対  
    'oleCtrl.Move 25, 17, Me.ScaleWidth - 25, Me.ScaleHeight - 17  
    oleCtrl.Move 0, 0, Me.ScaleWidth, Me.ScaleHeight  
End Sub
```

```
Private Sub mnuClose_Click()  
    Unload Me  
End Sub
```

```
Private Sub mnuResClose_Click()  
    'Unload Me  
    Unload frmStarMain.ActiveForm  
End Sub
```

```
Private Sub oleCtrl_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)  
    If Button = 2 Then oleCtrl.ShowWhatsThis  
End Sub
```

```

Attribute VB_Name = "frmSelDB1"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Option Explicit

```

```

Private Sub cmbFileType_Click()
    Dim tmp As String

    tmp = Mid(cmbFileType, InStr(cmbFileType, "(") + 1)
    filFileList.Pattern = Left(tmp, Len(tmp) - 1)
End Sub

```

```

Private Sub cmdCancel_Click()
    modSTAR.tmpFormCloseType = vbCancel
    Unload Me
End Sub

```

```

Private Sub cmdCancel_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then cmdCancel.ShowWhatsThis
End Sub

```

```

Private Sub cmdOK_Click()
    Dim Ext As String

    On Error GoTo Err1

    '入力データのチェック
    If lblCurFile.Caption = "" Then
        MsgBox "ファイルが選択されていません。"
        Exit Sub
    End If
    Ext = UCase(Right(lblCurFile.Caption, 3))
    If (Ext = "MDB" Or Ext = "XLS") And _
        lstDataRange.ListIndex < 0 Then
        MsgBox "データ範囲が選択されていません。"
        Exit Sub
    End If

    'ファイル名とフォルム名を設定
    If Right(dirDirList.Path, 1) <> "¥" Then
        modSTAR.tmpInpFName$(0) = dirDirList.Path & "¥" & lblCurFile.Caption
    Else
        modSTAR.tmpInpFName$(0) = dirDirList.Path & lblCurFile.Caption
    End If

    If Ext = "MDB" Or Ext = "XLS" Then
        modSTAR.tmpInpTName$(0) = lstDataRange.List(lstDataRange.ListIndex)
    Else
        modSTAR.tmpInpTName$(0) = ""
    End If

    modSTAR.InpDir$ = dirDirList.Path
    modSTAR.InpDrv$ = Left(dirDirList.Path, 2)

    modSTAR.tmpFormCloseType = vbOK
    Unload Me

    Exit Sub

```

```

Err1:
    MsgBox "Error in frmSelDB1/cmdOK"
    Call ShowErrorMessage
    modSTAR.tmpFormCloseType = vbCancel
End Sub

```

```

Private Sub cmdOK_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then cmdOK.ShowWhatsThis
End Sub

```

```

Private Sub dirDirList_Change()
    lblCurDir.Caption = dirDirList.Path
    filFileList.Path = dirDirList.Path
End Sub

```

```

Private Sub dirDirList_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)

```

```

    If Button = 2 Then dirDirList.ShowWhatsThis
End Sub

```

```

Private Sub drvDrive_Change()
    Dim oldPath As String

    oldPath = dirDirList.Path
    On Error GoTo trapDrv:
    dirDirList.Path = drvDrive.Drive
    On Error GoTo 0
    Exit Sub

```

```

trapDrv:
    Beep
    MsgBox "ドライブの準備ができていません。", 48, "エラー"
    dirDirList.Path = oldPath
    drvDrive.Drive = Left(oldPath, 2)

End Sub

```

```

Private Sub filFileList_Click()
    lblCurFile.Caption = filFileList.FileName
End Sub

```

```

Private Sub filFileList_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)

    If Button = 2 Then filFileList.ShowWhatsThis

End Sub

```

```

Private Sub Form_Load()
    Dim FileType$
    Dim m As Long
    Dim m2 As Long

    On Error GoTo Err1

    frmSelDB1.Caption = modSTAR.tmpFormName$

    lblCurDir.Caption = dirDirList.Path
    lblCurFile.Caption = ""

    'FileType$ = "Excel 5.0 (*.XLS)|Access 2.0 (*.MDB)|CSVファイル (*.CSV)|標準テキスト (*.STF)|テキストファイル (*.TXT)|"
    FileType$ = modSTAR.tmpInpFilter$

    m = 1
    Do While 1
        m2 = InStr(m, FileType$, "|", 1)
        If m2 = 0 Then Exit Do
        cmbFileType.AddItem Mid$(FileType$, m, m2 - m)
        m = m2 + 1
    Loop
    cmbFileType.ListIndex = 0

    dirDirList.Path = modSTAR.InpDir$
    drvDrive.Drive = Left(modSTAR.InpDir$, 2)

    modSTAR.tmpFormCloseType = vbCancel

    Exit Sub

```

```

Err1:
    MsgBox "Error in frmSelDB1/Load"
    Call ShowErrorMessage
    modSTAR.tmpFormCloseType = vbCancel
    Unload Me 'ここでUnload Meを実行するとエラーになるが、
              'これは親フォームにエラーが発生したことを知らせるため。

End Sub

```

```

Private Sub lblCurDir_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)

    If Button = 2 Then lblCurDir.ShowWhatsThis

End Sub

```

```

Private Sub lblCurFile_Change()
    Dim dbTMP As Database
    Dim i As Long
    Dim LCount As Long
    Dim tmpFilename As String

    Screen.MousePointer = 11

```

```

'ファイル名の取得
If Right(dirDirList.Path, 1) <> "¥" Then
    tmpFilename = dirDirList.Path & "¥" & lblCurFile.Caption
Else
    tmpFilename = dirDirList.Path & lblCurFile.Caption
End If

'データ範囲リストボックスをクリア
LCount = lstDataRange.ListCount
For i = LCount - 1 To 0 Step -1
    lstDataRange.RemoveItem i
Next i

'データ範囲リストボックスの更新
Select Case UCase(Right(tmpFilename, 3))
Case "MDB"
    Set dbTMP = Workspaces(0).OpenDatabase(tmpFilename, False, True)
    For i = 0 To dbTMP.TableDefs.Count - 1
        lstDataRange.AddItem dbTMP.TableDefs(i).Name
    Next i
    dbTMP.Close
Case "XLS"
    Set dbTMP = Workspaces(0).OpenDatabase(tmpFilename, False, True, "Excel 8.0;")
    For i = 0 To dbTMP.TableDefs.Count - 1
        lstDataRange.AddItem dbTMP.TableDefs(i).Name
    Next i
    dbTMP.Close
End Select

Screen.MousePointer = 0
End Sub

Private Sub lblCurFile_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblCurFile.ShowWhatsThis
End Sub

Private Sub lblDataRange_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblDataRange.ShowWhatsThis
End Sub

Private Sub lblDirectory_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblDirectory.ShowWhatsThis
End Sub

Private Sub lblDrive_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblDrive.ShowWhatsThis
End Sub

Private Sub lblFile_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblFile.ShowWhatsThis
End Sub

Private Sub lblFileType_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblFileType.ShowWhatsThis
End Sub

Private Sub lstDataRange_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lstDataRange.ShowWhatsThis
End Sub

```

```
Attribute VB_Name = "frmSelDB3"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Option Explicit
```

```
Private Sub cmbFileType_Click()
    Dim tmp As String

    tmp = Mid(cmbFileType.Text, InStr(cmbFileType.Text, "(") + 1)
    filFile1.Pattern = Left(tmp, Len(tmp) - 1)
    filFile2.Pattern = Left(tmp, Len(tmp) - 1)
    filFile3.Pattern = Left(tmp, Len(tmp) - 1)
End Sub
```

```
Private Sub cmdCancel_Click()
    modSTAR.tmpFormCloseType = vbCancel
    Unload Me
End Sub
```

```
Private Sub cmdCancel_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)

    If Button = 2 Then cmdCancel.ShowWhatsThis

End Sub
```

```
Private Sub cmdOK_Click()
    Dim dtKANRI As Recordset
    Dim FName$(3)
    Dim Ext As String
    Dim i As Long
    'Add 96/12/01

    On Error GoTo Err1

    '入力データのチェック
    If modSTAR.NoOffline >= 1 Then
        If lblCurFile1.Caption = "" Then
            MsgBox "評価ライン1の入力ファイル名が指定されていません!"
            Exit Sub
        End If
    End If
    If modSTAR.NoOffline >= 2 Then
        If lblCurFile2.Caption = "" Then
            MsgBox "評価ライン2の入力ファイル名が指定されていません!"
            Exit Sub
        End If
    End If
    If modSTAR.NoOffline >= 3 Then
        If lblCurFile3.Caption = "" Then
            MsgBox "評価ライン3の入力ファイル名が指定されていません!"
            Exit Sub
        End If
    End If
    Ext = UCase(Right(lblCurFile1.Caption, 3))
    If (Ext = "MDB" Or Ext = "XLS") And _
        cmbRange1.ListIndex < 0 Then
        MsgBox "データ範囲が選択されていません。"
        Exit Sub
    End If
    Ext = UCase(Right(lblCurFile2.Caption, 3))
    If (Ext = "MDB" Or Ext = "XLS") And _
        cmbRange2.ListIndex < 0 Then
        MsgBox "データ範囲が選択されていません。"
        Exit Sub
    End If
    Ext = UCase(Right(lblCurFile3.Caption, 3))
    If (Ext = "MDB" Or Ext = "XLS") And _
        cmbRange3.ListIndex < 0 Then
        MsgBox "データ範囲が選択されていません。"
        Exit Sub
    End If

    'ファイル名とテーブル名を設定
    If Right(dirDirectory.Path, 1) <> "*" Then
        modSTAR.tmpInpFName$(0) = dirDirectory.Path & "*" & lblCurFile1.Caption
        modSTAR.tmpInpFName$(1) = dirDirectory.Path & "*" & lblCurFile2.Caption
        modSTAR.tmpInpFName$(2) = dirDirectory.Path & "*" & lblCurFile3.Caption
    Else
        modSTAR.tmpInpFName$(0) = dirDirectory.Path & lblCurFile1.Caption
        modSTAR.tmpInpFName$(1) = dirDirectory.Path & lblCurFile2.Caption
        modSTAR.tmpInpFName$(2) = dirDirectory.Path & lblCurFile3.Caption
    End If

    Ext = UCase(Right(lblCurFile1.Caption, 3))
    If Ext = "MDB" Or Ext = "XLS" Then
        modSTAR.tmpInpTName$(0) = cmbRange1.List(cmbRange1.ListIndex)
    Else
        modSTAR.tmpInpTName$(0) = ""
    End If
```

```

Ext = UCase(Right(lblCurFile2.Caption, 3))
If Ext = "MDB" Or Ext = "XLS" Then
    modSTAR.tmpInpTName$(1) = cmbRange2.List(cmbRange2.ListIndex)
Else
    modSTAR.tmpInpTName$(1) = ""
End If
Ext = UCase(Right(lblCurFile3.Caption, 3))
If Ext = "MDB" Or Ext = "XLS" Then
    modSTAR.tmpInpTName$(2) = cmbRange3.List(cmbRange3.ListIndex)
Else
    modSTAR.tmpInpTName$(2) = ""
End If
Debug.Print modSTAR.tmpInpFName$(0), modSTAR.tmpInpTName$(0)
Debug.Print modSTAR.tmpInpFName$(1), modSTAR.tmpInpTName$(1)
Debug.Print modSTAR.tmpInpFName$(2), modSTAR.tmpInpTName$(2)

' 96/12/01 If modSTAR.NoOfLine < 3 Then          ' NULだと他のル-ンでエラーになる為
' 96/12/01 modSTAR.tmpInpFName$(2) = modSTAR.tmpInpFName$(0)
' 96/12/01 End If
For i = modSTAR.NoOfLine To 2          ' NULだと他のル-ンでエラーになる為
    modSTAR.tmpInpFName$(i) = modSTAR.tmpInpFName$(0)
Next i

modSTAR.InpDir$ = dirDirectory.Path
' modSTAR.InpDrv$ = Left(modSTAR.InpDir$, 2)

modSTAR.tmpFormCloseType = vbOK
Unload Me

Exit Sub

Err1:
MsgBox "Error in frmSelDB3/cmdOK"
Call ShowErrorMessage
modSTAR.tmpFormCloseType = vbCancel          ' なくとも良い?
' Unload Me

End Sub

Private Sub cmdOK_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then cmdOK.ShowWhatsThis
End Sub

Private Sub dirDirectory_Change()
    filFile1.Path = dirDirectory.Path
    filFile2.Path = dirDirectory.Path
    filFile3.Path = dirDirectory.Path
End Sub

Private Sub dirDirectory_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then dirDirectory.ShowWhatsThis
End Sub

Private Sub drvDrive_Change()
    Dim oldPath As String

    oldPath = dirDirectory.Path
    On Error GoTo trapDrv:
    dirDirectory.Path = drvDrive.Drive
    On Error GoTo 0
    Exit Sub

trapDrv:
    Beep
    MsgBox "ドライブの準備ができていません!", 48, "エラー"
    dirDirectory.Path = oldPath
    drvDrive.Drive = Left(oldPath, 2)
End Sub

Private Sub filFile1_Click()
    ' txtFile1.Text = filFile1.FileName
    lblCurFile1.Caption = filFile1.FileName
End Sub

Private Sub filFile1_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then filFile1.ShowWhatsThis
End Sub

```

```
Private Sub filFile2_Click()
    'txtFile2.Text = filFile2.FileName
    lblCurFile2.Caption = filFile2.FileName
End Sub
```

```
Private Sub filFile2_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then filFile2.ShowWhatsThis
End Sub
```

```
Private Sub filFile3_Click()
    'txtFile3.Text = filFile3.FileName
    lblCurFile3.Caption = filFile3.FileName
End Sub
```

```
Private Sub filFile3_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then filFile3.ShowWhatsThis
End Sub
```

```
Private Sub Form_Load()
    Dim FileType$
    Dim m As Long
    Dim m2 As Long

    On Error GoTo Err1

    'FileType$ = "標準テキスト (*.STF)|Excel 5.0 (*.XLS)|CSVファイル (*.CSV)|"
    FileType$ = modSTAR.tmpInpFilter$

    m = 1
    Do While 1
        m2 = InStr(m, FileType$, "|", 1)
        If m2 = 0 Then Exit Do
        cmbFileType.AddItem Mid$(FileType$, m, m2 - m)
        m = m2 + 1
    Loop
    cmbFileType.ListIndex = 0

    If modSTAR.NoOfLine < 3 Then
        filFile3.Enabled = False
        cmbRange3.Enabled = False
    End If
    If modSTAR.NoOfLine < 2 Then
        filFile2.Enabled = False
        cmbRange2.Enabled = False
    End If

    dirDirectory.Path = modSTAR.InpDir$
    drvDrive.Drive = Left(modSTAR.InpDir$, 2)

    modSTAR.tmpFormCloseType = vbCancel

    Exit Sub
```

```
Err1:
    MsgBox "Error in frmSelDB3/Load"
    Call ShowErrorMessage
    modSTAR.tmpFormCloseType = vbCancel
    Unload Me '強制エラー
End Sub
```

```
Private Sub lblCurFile1_Change()
    Dim dbTMP As Database
    Dim i As Long
    Dim LCount As Long
    Dim tmpFilename As String

    Screen.MousePointer = 11

    'ファイル名の取得
    If Right(dirDirectory.Path, 1) <> "*" Then
        tmpFilename = dirDirectory.Path & "*" & lblCurFile1.Caption
    Else
        tmpFilename = dirDirectory.Path & lblCurFile1.Caption
    End If

    'データ範囲リストボックスをクリア
    LCount = cmbRange1.ListCount
    For i = LCount - 1 To 0 Step -1
        cmbRange1.RemoveItem i
    Next i
```



```

'データ範囲リストボックスの更新
Select Case UCase(Right(tmpFilename, 3))
Case "MDB"
  Set dbTMP = Workspaces(0).OpenDatabase(tmpFilename, False, True)
  For i = 0 To dbTMP.TableDefs.Count - 1
    cmbRange1.AddItem dbTMP.TableDefs(i).Name
  Next i
  dbTMP.Close
Case "XLS"
  Set dbTMP = Workspaces(0).OpenDatabase(tmpFilename, False, True, "Excel 8.0;")
  For i = 0 To dbTMP.TableDefs.Count - 1
    cmbRange1.AddItem dbTMP.TableDefs(i).Name
  Next i
  dbTMP.Close
End Select

If cmbRange1.ListCount > 0 Then
  cmbRange1.ListIndex = 0
End If

Screen.MousePointer = 0

End Sub

Private Sub lblCurFile1_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
  If Button = 2 Then lblCurFile1.ShowWhatsThis
End Sub

Private Sub lblCurFile2_Change()
  Dim dbTMP As Database
  Dim i As Long
  Dim LCount As Long
  Dim tmpFilename As String

  Screen.MousePointer = 11

  'ファイル名の取得
  If Right(dirDirectory.Path, 1) <> "." Then
    tmpFilename = dirDirectory.Path & "." & lblCurFile2.Caption
  Else
    tmpFilename = dirDirectory.Path & lblCurFile2.Caption
  End If

  'データ範囲リストボックスをクリア
  LCount = cmbRange2.ListCount
  For i = LCount - 1 To 0 Step -1
    cmbRange2.RemoveItem i
  Next i

  'データ範囲リストボックスの更新
  Select Case UCase(Right(tmpFilename, 3))
  Case "MDB"
    Set dbTMP = Workspaces(0).OpenDatabase(tmpFilename, False, True)
    For i = 0 To dbTMP.TableDefs.Count - 1
      cmbRange2.AddItem dbTMP.TableDefs(i).Name
    Next i
    dbTMP.Close
  Case "XLS"
    Set dbTMP = Workspaces(0).OpenDatabase(tmpFilename, False, True, "Excel 8.0;")
    For i = 0 To dbTMP.TableDefs.Count - 1
      cmbRange2.AddItem dbTMP.TableDefs(i).Name
    Next i
    dbTMP.Close
  End Select

  If cmbRange2.ListCount > 0 Then
    cmbRange2.ListIndex = 0
  End If

  Screen.MousePointer = 0

End Sub

Private Sub lblCurFile2_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
  If Button = 2 Then lblCurFile2.ShowWhatsThis
End Sub

Private Sub lblCurFile3_Change()
  Dim dbTMP As Database
  Dim i As Long
  Dim LCount As Long
  Dim tmpFilename As String

  Screen.MousePointer = 11

```

```

'ファイル名の取得
If Right(dirDirectory.Path, 1) <> "¥" Then
    tmpFilename = dirDirectory.Path & "¥" & lblCurFile3.Caption
Else
    tmpFilename = dirDirectory.Path & lblCurFile3.Caption
End If

'データ範囲リストボックスをクリア
LCount = cmbRange3.ListCount
For i = LCount - 1 To 0 Step -1
    cmbRange3.RemoveItem i
Next i

'データ範囲リストボックスの更新
Select Case UCase(Right(tmpFilename, 3))
Case "MDB"
    Set dbTMP = Workspaces(0).OpenDatabase(tmpFilename, False, True)
    For i = 0 To dbTMP.TableDefs.Count - 1
        cmbRange3.AddItem dbTMP.TableDefs(i).Name
    Next i
    dbTMP.Close
Case "XLS"
    Set dbTMP = Workspaces(0).OpenDatabase(tmpFilename, False, True, "Excel 8.0;")
    For i = 0 To dbTMP.TableDefs.Count - 1
        cmbRange3.AddItem dbTMP.TableDefs(i).Name
    Next i
    dbTMP.Close
End Select

If cmbRange3.ListCount > 0 Then
    cmbRange3.ListIndex = 0
End If

Screen.MousePointer = 0
End Sub

Private Sub lblCurFile3_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblCurFile3.ShowWhatsThis
End Sub

Private Sub lblDirectory_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblDirectory.ShowWhatsThis
End Sub

Private Sub lblDrive_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblDrive.ShowWhatsThis
End Sub

Private Sub lblFileType_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblFileType.ShowWhatsThis
End Sub

Private Sub lblLine1_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblLine1.ShowWhatsThis
End Sub

Private Sub lblLine2_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblLine2.ShowWhatsThis
End Sub

Private Sub lblLine3_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblLine3.ShowWhatsThis
End Sub

Private Sub lblRange1_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblRange1.ShowWhatsThis

```

End Sub

Private Sub lblRange2_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)

 If Button = 2 Then lblRange2.ShowWhatsThis

End Sub

Private Sub lblRange3_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)

 If Button = 2 Then lblRange3.ShowWhatsThis

End Sub

```

Attribute VB_Name = "frmUserCheck2"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Option Explicit

Dim LoginOK As Boolean

Private Sub cmdOK_Click()
    Dim dtKNR As rdoResultset
    Dim Connect$
    Dim SQL$
    Dim i As Long
    Dim hKNR As Integer
    Dim IFERR As Integer

    Screen.MousePointer = 11

    modSTAR.SuperVisor = True

    UID$ = txtUID.Text
    PWD$ = txtPWD.Text

    'STARデータベースをrdoを使ってイフ
    On Error GoTo ErrorLoginFailed
    On Error GoTo ErrorLogin
    Connect$ = "ODBC;"
        & "DATABASE=" & DBN$ & "," -
        & "UID=" & UID$ & "," -
        & "PWD=" & PWD$ & "," -
        & "DSN=" & DSN$ & "," -
    Set deENV = rdoEnvironments(0)
    deENV.LoginTimeout = 30
    Set dbSTAR5 = deENV.OpenConnection("", rdDriverNoPrompt, False, Connect$)

    For i = 0 To dbSTAR5.rdoTables.Count - 1
        Debug.Print dbSTAR5.rdoTables(i).Name, dbSTAR5.rdoTables(i).Updatable
    Next i

    modSTAR.UID$ = deENV.UserName
    Debug.Print "rdoEngine.rdoDefaultUser: "; rdoEngine.rdoDefaultUser
    Debug.Print "rdoEngine.rdoDefaultPassword: "; rdoEngine.rdoDefaultPassword
    Debug.Print "deENV.UserName: "; deENV.UserName

    '管理データベースの読み込み
    Call KanriRead(IFERR)
    If IFERR = 1 Then GoTo ErrorLogin

    '管理データベースに接続
    SQL$ = "SELECT id FROM 管理データベース WHERE (id = 99999);"
    SQL$ = "SELECT id FROM 管理データベース;"
    Set dtKNR = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
    Set dtKNR = dbSTAR5.OpenResultset(SQL$, rdOpenDynamic, rdConcurLock)
    Set dtKNR = xOpenResultset(SQL$, rdOpenDynamic, rdConcurLock, hKNR)
    If hKNR = -1 Then GoTo ErrorLogin

    On Error GoTo ErrorGeneral

    dtKNR.Edit
    dtKNR![id] = dtKNR![id]
    dtKNR.Update
    dtKNR.AddNew
    dtKNR![id] = 99999
    dtKNR.Update
    '管理データベースに1行追加してみる
    '追加できれば管理者と判断する

    dtKNR.Close
    xCloseResultset dtKNR, hKNR

    SQL$ = "DELETE 管理データベース WHERE (id = 99999);"
    dbSTAR5.Execute SQL$
    On Error GoTo 0
    '追加した行を削除しておく

    modSTAR.SuperVisor = True
    LoginOK = True
    Screen.MousePointer = 0
    Unload Me

    Exit Sub

ErrorLoginFailed:
    MsgBox rdoEngine.rdoErrors(0).Number
    MsgBox rdoEngine.rdoErrors(0).Description
    End

ErrorLogin:
    MsgBox "Loginに失敗しました。"
    Call ShowErrorMessage
    Screen.MousePointer = 0
    On Error GoTo 0
    Exit Sub

```

ErrorGeneral:

```

If rdoEngine.rdoErrors(0).Number = 229 Then ' 読み込み不可
    ' dtKNR.Close
    xCloseResultset dtKNR, hKNR
    On Error GoTo 0
    modSTAR.SuperVisor = False ' General Mode
    LoginOK = True
    Screen.MousePointer = 0
    Unload Me
Else
    MsgBox rdoEngine.rdoErrors(0).Description
    ' dtKNR.Close
    xCloseResultset dtKNR, hKNR
    On Error GoTo 0
    Screen.MousePointer = 0
    dbSTAR5.Close
End
End If

End Sub

Private Sub cmdOK_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then cmdOK.ShowWhatsThis
End Sub

Private Sub cmdQuit_Click()
    End
End Sub

Private Sub cmdQuit_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then cmdQuit.ShowWhatsThis
End Sub

Private Sub cmdSetup_Click()
    frmUtilSetup.Show 1
End Sub

Private Sub cmdSetup_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then cmdSetup.ShowWhatsThis
End Sub

Private Sub Form_Load()
    txtUID.Text = ""
    txtPWD.Text = ""
    LoginOK = False
End Sub

Private Sub Form_Unload(Cancel As Integer)
    If LoginOK = False Then End
End Sub

Private Sub lblPWD_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblPWD.ShowWhatsThis
End Sub

Private Sub lblUID_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then lblUID.ShowWhatsThis
End Sub

Private Sub txtPWD_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If Button = 2 Then txtPWD.ShowWhatsThis
End Sub

```

```
Private Sub txtUID_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
  If Button = 2 Then txtUID.ShowWhatsThis
End Sub
```

```

Attribute VB_Name = "modSTAR"
Option Explicit

Public frmMain As New frmNormal

Public SuperVisor As Boolean '管理者/ユーザーモード'

Public StarDB$ 'STAR管理のデータベース名
Public StarSP$ '表計算用のシート名
Public SysQDB$ 'システムのカリ用のデータベース
Public QueryDB$ 'カリ用のデータベース

Public StarDir$ 'STARのシステムフォルダ名 (cf "C:\STAR6")
Public DBDir$ 'STARデータベースのフォルダ名
Public WorkDir$ '作業フォルダ名 (cf: "C:\WORK")
Public WSketchDir$ 'AutoSketchのフォルダ名
Public BTView As Long '検索結果の表示方法
Public MaxRet As Long '検索数の上限

Public dbSTAR As Database '管理データベースのオブジェクト

Public MaxCase As Long '解析ケース番号の最大値

Public posSochi As Long '管理TBLの現試験装置の位置
Public posTai As Long '管理TBLの現試験体の位置
Public posBubun As Long '管理TBLの現試験部分の位置

Public nmSochi$ '現試験装置名
Public nmTai$ '現試験体名
Public nmBubun$ '現試験部分名
Public nmID As Long '試験部分の参照ID
Public nmAID As Long '解析・表計算の参照ID

Public typeKanriAdd As Long '

Public KanriLine1 As Long '管理TBLの現在の行位置(試験部分へ)
Public KanriLine2 As Long '管理TBLの現在の行位置(解析ケースへ)
Public NoOfLine As Long '現試験部分の評価ライン数
Public CurrentCase$ '現在の解析ケース名

Public tmpSochi$
Public tmpTai$
Public tmpBubun$

Public tmpDBDatabaseName$
Public tmpDBRecordSource$
Public tmpDBName$(4)
Public tmpDBRecSrc$(4)
Public tmpDBConnect$

Public tmpViewNo As Long
Public tmpViewSQL$
Public tmpViewTableName$
Public tmpViewTableRow As Long
Public tmpViewTableCol As Long
Public tmpViewDamageName$
Public tmpDeleteSQL$

Public tmpFormName$
Public tmpFormOpenType As Long
Public tmpFormCloseType As Long

Public tmpCondInpCtrl$

Public tmpInpFName$(4) '読み込み(外部)ファイル名
Public tmpInpIName$(4) '外部ファイルのテーブル名
Public tmpInpFilter$
Public tmpAnalfName$(4)
Public tmpTempFName$
Public tmpShapeFName$
Public tmpCurrentCase$ '解析ケース
Public tmpAID As Long '解析ケースの参照ID
Public tmpMaxID As Long '既存データの参照/解析id

Public CnvFldType$(20) 'VBのフィールドタイプをSQLの文字列に変換

Public tmpSecTableSel As Long '1:全テーブル, 2:現試験体, 3:現解析ケース
Public tmpSecDamageSel As Long '1:全テーブル, 2:現試験体, 3:現解析ケース
Public tmpTokInterSel As Long '1:全テーブル, 2:現試験体, 3:現解析ケース

Public tmpBunNo As Long
Public tmpBunQName$
Public CurBunLineNo As Long
Public CurBunQueryNo As Long
Public CurBunQueryName$
Public tmpTokNo As Long
Public CurTokFromName$
Public CurTokQueryNo As Long
Public CurTokQueryName$
Public CurQuerySQL$

'----- STAR 7.0 -----

Public deENV As rdoEnvironment

```

```

Public dbSTAR5 As rdoConnection
'Public dtKNR5 As rdoResultset

Public AFType(100, 2)
Public AFName$(100, 3)
Public AFldMax As Long

Public Crack(500, 50) As Variant      'き裂データ (最大 500行)
Public Anal(300, 100) As Variant      '解析データ (最大 3x100行)
Public Jiten1(300, 100) As Variant    '時点データ1 (最大 3x100行)
Public Jiten2(300, 100) As Variant    '時点データ2 (最大 3x100行)
Public Table(300, 100) As Variant     '表計算データ (最大 3x100行)
Public KanriT(2000, 50) As Variant    '管理フォーム (最大 2000行)
Public HyokaT(200, 50) As Variant     '評価フォーム (最大 200行)
Public nKanriT As Long                '管理フォームの行数
Public nHyokaT As Long                '評価フォームの行数
Public cKanriT As Long                '管理フォームの列数
Public cHyokaT As Long                '評価フォームの列数
Public KanriFN$(50)                   '管理フォームのファイル名

Public DSNS$                          'Data Source Name
Public DBNS$                          'Database Name
Public QDB$                            'QueryDB (MDB)
Public UID$                            'User ID
Public PWD$                            'Password

Public Type KanriLine
    id As Long                          'id
    nm$ As String                       '試験装置名
    nmT As String                       '試験体名
    nmB As String                       '試験部分名
    NOL As Long                         '評価ライン数
    RID As Long                         '参照id
    nmC As String                       '解析ケース
    Hid As Long                         '評価法id
    Cid As Long                         'き裂id
    Sid As Long                         '試験条件id
    Oid As Long                         '温度id
    Mid As Long                         '形状id
    AID As Long                         '解析id
    Tid As Long                         '表計算id
    Did As Long                         '損傷値id
    CSid As Long                       'き裂Sid
    ASid As Long                       '解析Sid
    TSid As Long                       '表計算Sid
    DSid As Long                       '損傷値Sid
End Type

Public InpDir$                        'データ入力ディレクトリ

Public ErrNo As Integer

Public Sub AddExponent(Orig$, Xchg$)
    Dim i As Long
    Dim Digit As Long
    Dim Expon As Long
    Dim Length As Long
    Dim Char$

    Digit = 0
    Expon = 0
    Xchg$ = ""
    Length = Len(Orig$)

    For i = 1 To Length
        Char$ = Mid$(Orig$, i, 1)
        If Asc(Char$) >= Asc("0") And Asc(Char$) <= Asc("9") Then
            Digit = 1
        End If

        If Char$ = "E" Or Char$ = "e" Then
            Expon = 1
        End If

        If Char$ = "+" Or Char$ = "-" Then
            If Digit = 1 And Expon = 0 Then
                Xchg$ = Xchg$ & "E"
            End If
        End If

        Xchg$ = Xchg$ & Char$
    Next i
End Sub

Public Sub AnalStfToCsv()
    '+++++++
    'このルーチンは現在使用していない: 1998/01/27
    '+++++++
    Dim Buffer$, Buf10$, BufNew$, cTitle$
    Dim i As Long
    Dim j As Long
    Dim k As Long

```


Dim Row As Long
 Dim Col As Long
 Dim AnalFieldCount As Long
 Dim StfFiles\$(3), CsvFiles\$(3)
 Dim Sheet(200, 100) As Double
 Dim StfName\$(100)

Screen.MousePointer = 11

AnalFieldCount = 95

StfName\$(0) = "DATE"
 StfName\$(1) = "LINE"
 StfName\$(2) = "R1"
 StfName\$(3) = "Z1"
 StfName\$(4) = "H1"
 StfName\$(5) = "R2"
 StfName\$(6) = "Z2"
 StfName\$(7) = "H2"
 StfName\$(8) = "TEMPMAX"

StfName\$(9) = "TIME1"
 StfName\$(10) = "TEMP1"
 StfName\$(11) = "LTEMP1"
 StfName\$(12) = "MTEMP1"
 StfName\$(13) = "STRS11"
 StfName\$(14) = "STRS21"
 StfName\$(15) = "STRS31"
 StfName\$(16) = "STRS41"
 StfName\$(17) = "LSTRS11"
 StfName\$(18) = "LSTRS21"
 StfName\$(19) = "LSTRS31"
 StfName\$(20) = "LSTRS41"
 StfName\$(21) = "MSTRS11"
 StfName\$(22) = "MSTRS21"
 StfName\$(23) = "MSTRS31"
 StfName\$(24) = "MSTRS41"
 StfName\$(25) = "EE11"
 StfName\$(26) = "EE21"
 StfName\$(27) = "EE31"
 StfName\$(28) = "EE41"
 StfName\$(29) = "EP11"
 StfName\$(30) = "EP21"
 StfName\$(31) = "EP31"
 StfName\$(32) = "EP41"
 StfName\$(33) = "EC11"
 StfName\$(34) = "EC21"
 StfName\$(35) = "EC31"
 StfName\$(36) = "EC41"

StfName\$(37) = "TIME2"
 StfName\$(38) = "TEMP2"
 StfName\$(39) = "LTEMP2"
 StfName\$(40) = "MTEMP2"
 StfName\$(41) = "STRS12"
 StfName\$(42) = "STRS22"
 StfName\$(43) = "STRS32"
 StfName\$(44) = "STRS42"
 StfName\$(45) = "LSTRS12"
 StfName\$(46) = "LSTRS22"
 StfName\$(47) = "LSTRS32"
 StfName\$(48) = "LSTRS42"
 StfName\$(49) = "MSTRS12"
 StfName\$(50) = "MSTRS22"
 StfName\$(51) = "MSTRS32"
 StfName\$(52) = "MSTRS42"
 StfName\$(53) = "EE12"
 StfName\$(54) = "EE22"
 StfName\$(55) = "EE32"
 StfName\$(56) = "EE42"
 StfName\$(57) = "EP12"
 StfName\$(58) = "EP22"
 StfName\$(59) = "EP32"
 StfName\$(60) = "EP42"
 StfName\$(61) = "EC12"
 StfName\$(62) = "EC22"
 StfName\$(63) = "EC32"
 StfName\$(64) = "EC42"

StfName\$(65) = "TIME3"
 StfName\$(66) = "TEMP3"
 StfName\$(67) = "LTEMP3"
 StfName\$(68) = "MTEMP3"
 StfName\$(69) = "STRS13"
 StfName\$(70) = "STRS23"
 StfName\$(71) = "STRS33"
 StfName\$(72) = "STRS43"
 StfName\$(73) = "LSTRS13"
 StfName\$(74) = "LSTRS23"
 StfName\$(75) = "LSTRS33"
 StfName\$(76) = "LSTRS43"
 StfName\$(77) = "MSTRS13"
 StfName\$(78) = "MSTRS23"
 StfName\$(79) = "MSTRS33"
 StfName\$(80) = "MSTRS43"
 StfName\$(81) = "EE13"
 StfName\$(82) = "EE23"

```

StfName$(83) = "EE33"
StfName$(84) = "EE43"
StfName$(85) = "EP13"
StfName$(86) = "EP23"
StfName$(87) = "EP33"
StfName$(88) = "EP43"
StfName$(89) = "EC13"
StfName$(90) = "EC23"
StfName$(91) = "EC33"
StfName$(92) = "EC43"

StfName$(93) = "Q"
StfName$(94) = "I"

StfFile$(0) = tmpInpFName$(0)
StfFile$(1) = tmpInpFName$(1)
StfFile$(2) = tmpInpFName$(2)
CsvFile$(0) = Left$(StfFile$(0), Len(StfFile$(0)) - 3) & "CSV"
CsvFile$(1) = Left$(StfFile$(1), Len(StfFile$(1)) - 3) & "CSV"
CsvFile$(2) = Left$(StfFile$(2), Len(StfFile$(2)) - 3) & "CSV"

For k = 0 To modSTAR.NoOfLine - 1
  Debug.Print k, StfFile$(k), CsvFile$(k)

  Row = 0

  Open StfFile$(k) For Input As #1

  Do While Not EOF(1)
    Line Input #1, Buffer$

    If Left$(Buffer$, 1) = "$" Then
      'コメント行はスキップ

    ElseIf Left$(Buffer$, 1) = "P" Then
      'PTITLE名を表示

    ElseIf Mid$(Buffer$, 2, 5) = "TITLE" Then
      cTitle$ = Mid$(Buffer$, 11, 10)
      'XYTITLE名を表示
      For j = 0 To AnalFieldCount - 1
        If cTitle$ = StfName$(j) Then
          Row = 0
          Col = j
          Exit For
        End If
      Next j

    Else
      For i = 1 To 6
        Buf10$ = Mid$(Buffer$, i * 10 + 11, 10)
        If Buf10$ = "" Or Buf10$ = " " Then
          Exit For
        End If

        'If Row >= SheetMax Then
        '  Call ErrorMessage
        'End If

        Call AddExponent(Buf10$, BufNew$)
        'Sheet(Row).Value(Col) = Val(BufNew$)
        Sheet(Row, Col) = Val(BufNew$)
        Row = Row + 1
      Next i

    End If

  Loop

  Close #1

  '-----
  Open CsvFile$(k) For Output As #1

  Print #1, StfName$(0);
  For j = 1 To Col
    Print #1, ","; StfName$(j);
  Next j
  Print #1, Chr$(13); Chr$(10);

  For j = 0 To Row - 1
    Print #1, Format(Date$, "yy/mm/dd"); ",";
    Print #1, k + 1;
    For j = 2 To Col
      Print #1, ","; Sheet(i, j);
    Next j
    Print #1, Chr$(13); Chr$(10);
  Next i

  Close #1

Next k

```

```

Screen.MousePointer = 0

End Sub

Public Sub CreateDxfShape(FP As Long, NBlock As Long)
Dim dtSHAPE As rdoResultset
Dim SQL$
Dim NOShape As Long
Dim hSHAPE As Integer

'FP: データを書き出すファイル識別子
'NBlock: BLOCKセクションに書込んだブロック数を返す
'modSTAR.nmID: 試験部分の参照ID

On Error GoTo Err1

NBlock = 0
NOShape = 0

SQL$ = "SELECT 起点r1,起点z1,終点r2,終点z2 FROM 形状 WHERE " _
      & "(参照id = " & modSTAR.nmID & ");"
' Set dtSHAPE = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
Set dtSHAPE = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hSHAPE)
If hSHAPE = -1 Then GoTo Err1

If dtSHAPE.EOF = False Then

'BLOCKセクションの定義
Print #1, "0"
Print #1, "BLOCK"
Print #1, "8" '画層名
Print #1, "Shape"
Print #1, "2" 'ブロック名
Print #1, "SHAPE"
Print #1, "70"
Print #1, "0"
Print #1, "10"
Print #1, "0.0"
Print #1, "20"
Print #1, "0.0"

Do Until dtSHAPE.EOF

'形状データの書込み
Print #1, "0"
Print #1, "LINE"
Print #1, "8" '画層名
Print #1, "Shape"
Print #1, "62" '色 灰
Print #1, "8"
Print #1, "10"
Print #1, dtSHAPE![起点z1]
Print #1, "20"
Print #1, dtSHAPE![起点r1]
Print #1, "11"
Print #1, dtSHAPE![終点z2]
Print #1, "21"
Print #1, dtSHAPE![終点r2]

NOShape = NOShape + 1
dtSHAPE.MoveNext

Loop

Print #1, "0"
Print #1, "ENDBLK"

End If

'dtSHAPE.Close
xCloseResultset dtSHAPE, hSHAPE

If NOShape > 0 Then NBlock = 1

Exit Sub

Err1:
MsgBox "Error in modSTAR/CreateDxfShape"
Call ShowErrorMessage
xCloseResultset dtSHAPE, hSHAPE
NBlock = 0

End Sub

Public Sub CreateDxfCrack(FP As Long, NBlock As Long)
Dim dtCRACK As rdoResultset
Dim SQL$
Dim NLayer As Long
Dim Angle As Double
Dim hCRACK As Integer

```

```

'FP: データを書き出すファイル識別子
'NBlock: BLOCKステーションに書込んだブロック数を返す
'modSTAR.nmID: 試験部分の参照ID

On Error GoTo Err1

SQL$ = "SELECT 起点r1,起点z1,起点θ1,終点r2,終点z2 FROM き裂 WHERE " _
      & "(参照id = " & modSTAR.nmID & ")" _
      & " ORDER BY 起点θ1;"
'Set dtCRACK = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
Set dtCRACK = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hCRACK)
If hCRACK = -1 Then GoTo Err1

NBlock = 0 'ブロック数

If dtCRACK.EOF = False Then

  'BLOCKステーションの定義
  Print #1, "0"
  Print #1, "BLOCK"
  Print #1, "2" 'ブロック名
  Print #1, "CRACK"
  Print #1, "70"
  Print #1, "0"
  Print #1, "10"
  Print #1, "0.0"
  Print #1, "20"
  Print #1, "0.0"

  Angle = -9999.9
  NLayer = 0

  Do Until dtCRACK.EOF

    If dtCRACK![起点θ1] <> Angle Then
      Angle = dtCRACK![起点θ1]
      NLayer = NLayer + 1
    End If

    'き裂データの書込み
    Print #1, "0"
    Print #1, "LINE"
    Print #1, "8" '画層名
    Print #1, "Crack" & Format$(Angle, "000")
    Print #1, "62" '色赤(1)
    Print #1, "1"
    Print #1, "10"
    Print #1, dtCRACK![起点z1]
    Print #1, "20"
    Print #1, dtCRACK![起点r1]
    Print #1, "11"
    Print #1, dtCRACK![終点z2]
    Print #1, "21"
    Print #1, dtCRACK![終点r2]

    dtCRACK.MoveNext

  Loop

  Print #1, "0"
  Print #i, "ENDBLK"

End If

'dtCRACK.Close
xCloseResultset dtCRACK, hCRACK

If NLayer > 0 Then NBlock = 1 'き裂データが存在するときは、ブロック数は1
Exit Sub

Err1:
MsgBox "Error in modSTAR/CreateDxfCrack"
Call ShowErrorMessage
xCloseResultset dtCRACK, hCRACK

End Sub

Public Sub CreateDxfHLine(FPL As Long, NoOffline As Long, NBlock As Long)
Dim dtANAL As rdoResultset
Dim SQL$
Dim LineNo As Long
Dim hANAL As Integer

'FPL: データを書き出すファイル識別子
'NoOffline: 試験部分の評価ラインの本数
'NBlock: BLOCKステーションに書込んだブロック数を返す
'modSTAR.nmAID: 解析データの参照ID

On Error GoTo Err1

```

```

For LineNo = 1 To NoOfLine

SQL$ = "SELECT 起点r1,起点z1,終点r2,終点z2 FROM 解析 WHERE (" _
& "(解析id = " & modSTAR.nmAID & ") AND " _
& "(ライン番号 = " & LineNo & ");"
' Set dtANAL = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
Set dtANAL = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hANAL)
If hANAL = -1 Then GoTo Err1

' BLOCKSセクションの記述
Print #FPL, "0"
Print #FPL, "BLOCK"
Print #FPL, "2"
Print #FPL, "HLINE" & LineNo
Print #FPL, "70"
Print #FPL, "0"
Print #FPL, "10"
Print #FPL, "0.0"
Print #FPL, "20"
Print #FPL, "0.0"
Print #FPL, "0"
Print #FPL, "POLYLINE"
Print #FPL, "8"
Print #FPL, "Line" & LineNo
Print #FPL, "62"
Print #FPL, "7"
Print #FPL, "70"
Print #FPL, "0"
Print #FPL, "66"
Print #FPL, "1"
Print #FPL, "40"
Print #FPL, "2.5"
Print #FPL, "41"
Print #FPL, "2.5"
Do Until dtANAL.EOF
Print #FPL, "0"
Print #FPL, "VERTEX"
Print #FPL, "8"
Print #FPL, "Line" & Format$(LineNo, "00")
Print #FPL, "10"
Print #FPL, dtANAL![起点z1]
Print #FPL, "20"
Print #FPL, dtANAL![起点r1]
dtANAL.MoveNext
Loop
Print #FPL, "0"
Print #FPL, "SEQEND"
Print #FPL, "0"
Print #FPL, "ENDBLK"

xCloseResultset dtANAL, hANAL

Next LineNo

'dtANAL.Close

NBlock = NoOfLine

Exit Sub

Err1:
MsgBox "Error in modSTAR/CreateDxfHLine"
Call ShowErrorMessage
xCloseResultset dtANAL, hANAL

End Sub

```

```

Public Sub CreateDxfHsect(FPD As Long, NoOfLine As Long, NBlock As Long)
Dim dtANAL As rdoResultset
Dim SQL$
Dim LineNo As Long
Dim hANAL As Integer

```

```

' FPD: データを書き出すファイル識別子
' NoOfLine: 試験部分の評価ラインの本数
' NBlock: BLOCKSセクションに書込んだブロック数を返す
' modSTAR.nmAID: 解析データの参照ID

```

```

On Error GoTo Err1

```

```

For LineNo = 1 To NoOfLine

SQL$ = "SELECT 起点r1,起点z1,終点r2,終点z2 FROM 解析 WHERE (" _
& "(解析id = " & modSTAR.nmAID & ") AND " _
& "(ライン番号 = " & LineNo & ");"
' Set dtANAL = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
Set dtANAL = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hANAL)
If hANAL = -1 Then GoTo Err1

' BLOCKSセクションの記述
Print #FPD, "0"
Print #FPD, "BLOCK"
Print #FPD, "2"

```

```

Print #FPD, "HSECT" & LineNo
Print #FPD, "70"
Print #FPD, "0"
Print #FPD, "10"
Print #FPD, "0.0"
Print #FPD, "20"
Print #FPD, "0.0"
Do Until dtANAL.EOF
  Print #FPD, "0"
  Print #FPD, "LINE"
  Print #FPD, "8"
  Print #FPD, "Sect" & LineNo
  Print #FPD, "62"
  Print #FPD, "8"
  Print #FPD, "10"
  Print #FPD, dtANAL![起点z1]
  Print #FPD, "20"
  Print #FPD, dtANAL![起点r1]
  Print #FPD, "11"
  Print #FPD, dtANAL![終点z2]
  Print #FPD, "21"
  Print #FPD, dtANAL![終点r2]
  dtANAL.MoveNext
Loop
Print #FPD, "0"
Print #FPD, "ENDBLK"

  xCloseResultset dtANAL, hANAL

Next LineNo

'dtANAL.Close

NBlock = NoOffline

Exit Sub

Err1:
MsgBox "Error in modSTAR/CreateDxfHsect"
Call ShowErrorMessage
xCloseResultset dtANAL, hANAL

End Sub

Public Sub CreateDxfLayer(FP As Long)
  'FP: データを書き出すファイル識別子

  'TABLESアクションの記述
  Print #FP, "0"
  Print #FP, "SECTION"
  Print #FP, "2"
  Print #FP, "TABLES"
  Print #FP, "0"
  Print #FP, "TABLE"
  Print #FP, "2"
  Print #FP, "LAYER"
  Print #FP, "70"
  Print #FP, "10"
  Print #FP, "0"
  Print #FP, "LAYER"
  Print #FP, "2"
  Print #FP, "Shape"
  Print #FP, "70"
  Print #FP, "0"
  Print #FP, "62"
  Print #FP, "7"
  Print #FP, "6"
  Print #FP, "CONTINUOUS"
  Print #FP, "0"
  Print #FP, "ENDTAB"
  Print #FP, "0"
  Print #FP, "ENDSEC"

End Sub

Public Sub CreateDxfInsert(FP As Long, BName$)
  'FP: データを書き出すファイル識別子

  'ENTITIESアクションにINSERTを記述
  Print #FP, "0"
  Print #FP, "INSERT"
  Print #FP, "2"
  Print #FP, BName$
  Print #FP, "10"
  Print #FP, "0.0"
  Print #FP, "20"
  Print #FP, "0.0"

End Sub

```

```

Public Sub CreateTable(TblName$, Fields$)
    Dim SQL$
    SQL$ = "CREATE TABLE " & TblName$ & " " & Fields$ & ";"
    dbSTAR5.Execute SQL$
End Sub

Public Sub DeleteDataSQL(SQL$)
    Debug.Print SQL$
    dbSTAR5.Execute SQL$
End Sub

Public Sub DeleteKanriSQL(SQL$)
    Dim dtKANRI As rdoResultset
    Dim i As Long
    Dim SSS$
    Dim hKANRI As Integer
    On Error GoTo Err1

    Debug.Print SQL$
    dbSTAR5.Execute SQL$
    SSS$ = "SELECT id FROM 管理簿-夕-フ&";

    ' Set dtKANRI = dbSTAR5.OpenResultset(SSS$, rdOpenDynamic, rdConcurLock)
    Set dtKANRI = xOpenResultset(SSS$, rdOpenDynamic, rdConcurLock, hKANRI)
    If hKANRI = -1 Then GoTo Err1

    i = 0
    Do Until dtKANRI.EOF
        dtKANRI.Edit
        dtKANRI![id] = i
        dtKANRI.Update
        dtKANRI.Bookmark = dtKANRI.LastModified
        i = i + 1
        dtKANRI.MoveNext
    Loop

    ' dtKANRI.Close
    xCloseResultset dtKANRI, hKANRI

Exit Sub

Err1:
    xCloseResultset dtKANRI, hKANRI
End Sub

Public Sub DeleteTable(TblName$, IFERR As Integer)
    Dim i As Long
    Dim SQL$

    For i = 0 To dbSTAR5.rdoTables.Count - 1
        If dbSTAR5.rdoTables(i).Name = TblName$ Then
            SQL$ = "DROP TABLE " & TblName$ & ";"
            dbSTAR5.Execute SQL$
            dbSTAR5.rdoTables.Refresh
            Exit For
        End If
    Next i
End Sub

Public Function GetMaxID(nmTbl$, nmFLD$) As Long
    Dim dtMAXID As rdoResultset
    Dim SQL$
    Dim hMAXID As Integer
    On Error GoTo Err1

    ' 最終IDの取得
    SQL$ = "SELECT MAX(" & nmFLD$ & ") FROM " & nmTbl$ & ";"
    ' Set dtMAXID = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
    Set dtMAXID = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hMAXID)
    If hMAXID = -1 Then GoTo Err1

    ' If dtMAXID.EOF = False Then
    '     dtMAXID.MoveLast
    '     GetMaxID = dtMAXID.rdoColumns(0).Value
    ' Else
    '     GetMaxID = 0
    ' End If

```

```

If IsNull(dtMAXID.rdoColumns(0).Value) = True Then
  GetMaxID = 0
Else
  GetMaxID = dtMAXID.rdoColumns(0).Value
End If

Debug.Print "GetMaxID: "; GetMaxID

'dtMAXID.Close
xCloseResultset dtMAXID, hMAXID

Exit Function

Err1:
xCloseResultset dtMAXID, hMAXID
GetMaxID = 0

End Function

Public Function InStrRev(Str1 As String, Str2 As String) As Long
  Dim i As Long

  i = 0
  Do
    InStrRev = i
    i = InStr(i + 1, Str1, Str2, 1)
  Loop Until i = 0

End Function

Public Sub KanriInsert5(InsPos As Long, IFERR As Integer)
  Dim dtKANRI As rdoResultset
  Dim i As Long
  Dim j As Long
  Dim KCols As Long
  Dim SQL$
  Dim vX1(30) As Variant
  Dim vX2(30) As Variant
  Dim TName$
  Dim hKANRI As Integer

  On Error GoTo Err1

  IFERR = 0

  '管理テーブルに新しい行を挿入、移動
  SQL$ = "SELECT * FROM 管理テーブル;"
  'SQL$ = SQL$ & " WHERE ID = (SELECT MAX(ID) FROM 管理テーブル);"
  'Set dtKANRI = dbSTAR5.OpenResultset(SQL$, rdOpenDynamic, rdConcurLock)
  Set dtKANRI = xOpenResultset(SQL$, rdOpenDynamic, rdConcurLock, hKANRI)
  If hKANRI = -1 Then GoTo Err1

  KCols = dtKANRI.rdoColumns.Count

  'レートの追加
  dtKANRI.AddNew
  dtKANRI![id] = InsPos
  dtKANRI![試験装置] = "NewRecord"
  dtKANRI![評価回数] = 0
  dtKANRI![参照id] = 0
  dtKANRI![解析ケース] = "-"
  dtKANRI![評価法id] = 0
  dtKANRI![き裂id] = 0
  dtKANRI![試験条件id] = 0
  dtKANRI![温度id] = 0
  dtKANRI![形状id] = 0
  dtKANRI![解析id] = 0
  dtKANRI![表計算id] = 0
  dtKANRI![損傷値id] = 0
  dtKANRI![き裂Sid] = 0
  dtKANRI![解析Sid] = 0
  dtKANRI![表計算Sid] = 0
  dtKANRI![損傷値Sid] = 0
  dtKANRI.Update

  If dtKANRI.EOF = False Then

    'レート*の移動
    dtKANRI.MoveFirst
    For i = 1 To InsPos
      dtKANRI.MoveNext
    Next i

    vX1(0) = InsPos
    For j = 1 To KCols - 1
      vX1(j) = Null
      TName$ = TypeName(dtKANRI.rdoColumns(j).Value)
      'If TName$ = "String" Then
      '  vX1(j) = Null
      'Else
      '  vX1(j) = 0
      'End If

```



```

Next j
Do Until dtKANRI.EOF
  For j = 0 To KCols - 1
    vX2(j) = dtKANRI.rdcColumns(j).Value
  Next j

  dtKANRI.Edit
  For j = 0 To KCols - 1
    dtKANRI.rdcColumns(j).Value = vX1(j)
  Next j
  dtKANRI.Update
  dtKANRI.Bookmark = dtKANRI.LastModified

  vX1(0) = vX2(0) + 1
  For j = 1 To KCols - 1
    vX1(j) = vX2(j)
  Next j
  dtKANRI.MoveNext
Loop

End If

'管理テーブルのクリア
dtKANRI.Close
xCloseResultset dtKANRI, hKANRI

Exit Sub

Err1:
MsgBox "Error in modSTAR/KanriInsert5"
Call ShowErrorMessage
On Error Resume Next
dtKANRI.Close
xCloseResultset dtKANRI, hKANRI
IFERR = 1

End Sub

Public Sub KanriInsert6(InsPos As Long, IFERR As Integer)
Dim i As Long
Dim j As Long

On Error GoTo Err1

IFERR = 0

For i = nKanriT - 1 To InsPos Step -1 ' 行の移動
  For j = 0 To cKanriT - 1
    KanriT(i + 1, j) = KanriT(i, j)
  Next j
Next i

nKanriT = nKanriT + 1
KanriT(InsPos, 1) = "NewRecord"

For i = 0 To nKanriT - 1
  KanriT(i, 0) = i ' idの再設定
Next i

Exit Sub

Err1:
MsgBox "Error in modSTAR/KanriInsert6"
IFERR = 1

End Sub

Public Sub KillFile(fileName$)

On Error GoTo 0 ' このル-チンではエラーチェックをしない

If Dir(fileName$) <> "" Then
  Kill fileName$
End If

End Sub

Public Function Max2(Var1, Var2)

Max2 = Var1
If Var2 > Var1 Then Max2 = Var2

End Function

Public Function Min2(Var1, Var2)

```

```

Min2 = Var1
If Var2 < Var1 Then Min2 = Var2
End Function

```

```

Public Function Max3(Var1, Var2, Var3)

Max3 = Var1
If Var2 > Max3 Then Max3 = Var2
If Var3 > Max3 Then Max3 = Var3

End Function

```

```

Public Function Min3(Var1, Var2, Var3)

Min3 = Var1
If Var2 < Min3 Then Min3 = Var2
If Var3 < Min3 Then Min3 = Var3

End Function

```

```

Public Sub SecDmgHykChk(KNR As KanriLine, IFERR As Integer)
Dim dtKANRI As rdoResultset
Dim SQL$
Dim InsPos As Long
Dim InsFlag As Boolean
Dim hKANRI As Integer

On Error GoTo Err1

IFERR = 0

' この評価法がすでに登録されているかをチェック
SQL$ = "SELECT id,評価法id FROM 管理テ-タ-フ WHERE (" _
& "(試験装置 = " & KNR.nmS & ") AND " _
& "(試験体 = " & KNR.nmT & ") AND " _
& "(試験部分 = " & KNR.nmB & ") AND " _
& "(解析ケ-ス = " & KNR.nmC & ");"

' Set dtKANRI = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
' Set dtKANRI = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hKANRI)
If hKANRI = -1 Then GoTo Err1

dtKANRI.MoveLast
InsPos = dtKANRI![id] + 1
InsFlag = True
dtKANRI.MoveFirst
dtKANRI.MoveNext

Do Until dtKANRI.EOF
' If StrComp(dtKANRI![評価法名], nmHyoka$, 1) = 0 Then
If dtKANRI![評価法id] = KNR.Hid Then
InsPos = dtKANRI![id]
InsFlag = False
Exit Do

' ElseIf StrComp(dtKANRI![評価法名], nmHyoka$, 1) > 0 Then
ElseIf dtKANRI![評価法id] > KNR.Hid Then
InsPos = dtKANRI![id]
InsFlag = True
Exit Do

End If

dtKANRI.MoveNext
Loop

' dtKANRI.Close
xCloseResultset dtKANRI, hKANRI

' 管理テ-タ-フに新しいケ-スを挿入、移動
If InsFlag = True Then
' Call modSTAR.KanriInsert2(InsPos)
' Call modSTAR.KanriInsert5(InsPos, IFERR)
Call modSTAR.KanriInsert6(InsPos, IFERR)
If IFERR = 1 Then GoTo Err1

' SQL$ = "SELECT * FROM 管理テ-タ-フ WHERE (id = " & InsPos & ");"
' Set dtKANRI = dbSTAR5.OpenResultset(SQL$, rdOpenDynamic, rdConcurLock)
' Set dtKANRI = xOpenResultset(SQL$, rdOpenDynamic, rdConcurLock, hKANRI)
' If hKANRI = -1 Then GoTo Err1

' dtKANRI.Edit
dtKANRI![試験装置] = KNR.nmS
dtKANRI![試験体] = KNR.nmT
dtKANRI![試験部分] = KNR.nmB
dtKANRI![評価法id] = KNR.NOL
dtKANRI![参照id] = KNR.RID
dtKANRI![解析ケ-ス] = KNR.nmC

```

```

'dtKANRI![評価法id] = KNR.Hid
'dtKANRI![き裂id] = KNR.Cid
'dtKANRI![試験条件id] = KNR.Sid
'dtKANRI![温度id] = KNR.Oid
'dtKANRI![形状id] = KNR.Mid
'dtKANRI![解析id] = KNR.AID
'dtKANRI![表計算id] = KNR.Tid
'dtKANRI![損傷値id] = KNR.Did
'dtKANRI![き裂Sid] = 0
'dtKANRI![解析Sid] = 0
'dtKANRI![表計算Sid] = 0
'dtKANRI![損傷値Sid] = 0
'dtKANRI.Update

'' dtKANRI.Close
'xCloseResultset dtKANRI, hKANRI

KanriT(InsPos, 1) = KNR.nmS ' 試験装置
KanriT(InsPos, 2) = KNR.nmT ' 試験体
KanriT(InsPos, 3) = KNR.nmB ' 試験部分
KanriT(InsPos, 4) = KNR.NOL ' 評価回数
KanriT(InsPos, 5) = KNR.RID ' 参照id
KanriT(InsPos, 6) = KNR.nmC ' 解析ケース
KanriT(InsPos, 7) = KNR.Hid ' 評価法id
KanriT(InsPos, 8) = KNR.Cid ' き裂id
KanriT(InsPos, 9) = KNR.Sid ' 試験条件id
KanriT(InsPos, 10) = KNR.Oid ' 温度id
KanriT(InsPos, 11) = KNR.Mid ' 形状id
KanriT(InsPos, 12) = KNR.AID ' 解析id
KanriT(InsPos, 13) = KNR.Tid ' 表計算id
KanriT(InsPos, 14) = KNR.Did ' 損傷値id
KanriT(InsPos, 15) = KNR.CSid ' き裂Sid
KanriT(InsPos, 16) = KNR.ASid ' 解析Sid
KanriT(InsPos, 17) = KNR.TSid ' 表計算Sid
KanriT(InsPos, 18) = KNR.DSid ' 損傷値Sid

Call KanriWrite(IFERR)
If IFERR = 1 Then GoTo Err1

End If

Exit Sub

Err1:
MsgBox "Error in modSTAR/SecDmgHykChk"
Call ShowErrorMessage
On Error Resume Next
'dtKANRI.Close
xCloseResultset dtKANRI, hKANRI
IFERR = 1

End Sub

Public Sub ShapeAddNew(dtSHAPE As Recordset, data As String)
'+++++
'このルーチンは現在使用していない: 1998/01/28
'このルーチンはShapeExtToDxfからコールされるが、
'ShapeExtToDxf自体が使用されていない。
'+++++

dtSHAPE.AddNew
dtSHAPE![DXF] = data
dtSHAPE![参照id] = modSTAR.nmID
dtSHAPE.Update

End Sub

Public Sub ShapeExtToDB()
'+++++
'このルーチンは現在使用していない: 1998/01/27
'+++++
Dim dbEX_DATA As Database
Dim dtEX_SHAPE As Recordset
Dim dtSHAPE As rdoResultset
Dim SQL$
Dim MaxID As Long

Screen.MousePointer = 11

'古い形状データを削除しておく
'SQL$ = "DELETE 形状 WHERE (参照id = " & modSTAR.nmID & ");"
'dbSTAR5.Execute SQL$
Call DeleteData("形状", modSTAR.nmID)

'外部ファイルをオープン
Set dbEX_DATA = OpenDatabase(modSTAR.tmpDBDatabaseName$, False, False, modSTAR.tmpDBCConnect$)
Set dtEX_SHAPE = dbEX_DATA.OpenRecordset(modSTAR.tmpDBRecordSource$, dbOpenSnapshot)

'形状テーブルのオープン
SQL$ = "SELECT * FROM 形状 WHERE (参照id = " & modSTAR.nmID & ");"
Set dtSHAPE = dbSTAR5.OpenResultset(SQL$, rdOpenDynamic, rdConcurLock)

```

```

MaxID = GetMaxID("形状", "id")
'形状テーブルにコト-
If dtEX_SHAPE.RecordCount > 0 Then
  Do Until dtEX_SHAPE.EOF
    dtSHAPE.AddNew
    MaxID = MaxID + 1
    dtSHAPE![id] = MaxID
    dtSHAPE![参照id] = modSTAR.nmID
    dtSHAPE![起点r1] = dtEX_SHAPE.Fields(0).Value
    dtSHAPE![起点z1] = dtEX_SHAPE.Fields(1).Value
    dtEX_SHAPE.MoveNext
    dtSHAPE![終点r2] = dtEX_SHAPE.Fields(0).Value
    dtSHAPE![終点z2] = dtEX_SHAPE.Fields(1).Value
    dtEX_SHAPE.MoveNext
    dtSHAPE.Update
  Loop
End If
'形状テーブルのクローズ
dtSHAPE.Close
'外部ファイルをクローズ
dtEX_SHAPE.Close
dbEX_DATA.Close
'Call SetKanriXID("形状id", modSTAR.nmID, modSTAR.nmID, "-", 0)
Call SetKanriXID6("形状id", modSTAR.nmID, modSTAR.nmID, "-", 0)
Screen.MousePointer = 0
End Sub

Public Sub ShapeExtToDxf(nmShape$)
'+++++
'このルーチンは現在使用していない: 1998/01/28
'+++++
Dim dbEX_DATA As Database
Dim dtEX_SHAPE As Recordset
Dim dtSHAPE As Recordset
Dim SQL$
Dim i As Long

Screen.MousePointer = 11
'古い形状テーブルを削除しておく
SQL$ = "DELETE 形状 WHERE (参照id = " & modSTAR.nmID & ");"
dbSTAR.Execute SQL$
Call DeleteDataR("形状", modSTAR.nmID)
'-----
'外部ファイルをオープン
Set dbEX_DATA = OpenDatabase(modSTAR.tmpDBDatabaseName$, False, False, modSTAR.tmpDBConnect$)
Set dtEX_SHAPE = dbEX_DATA.OpenRecordset(modSTAR.tmpDBRecordSource$, dbOpenSnapshot)
'形状テーブルのオープン
Set dtSHAPE = dbSTAR.OpenRecordset("形状", dbOpenDynaset)
If dtEX_SHAPE.RecordCount > 0 Then
  'TABLEステーションの定義 (LAYERのみ)
  Call ShapeAddNew(dtSHAPE, "0")
  Call ShapeAddNew(dtSHAPE, "SECTION")
  Call ShapeAddNew(dtSHAPE, "2")
  Call ShapeAddNew(dtSHAPE, "TABLES")
  Call ShapeAddNew(dtSHAPE, "0")
  Call ShapeAddNew(dtSHAPE, "TABLE")
  Call ShapeAddNew(dtSHAPE, "2")
  Call ShapeAddNew(dtSHAPE, "LAYER")
  Call ShapeAddNew(dtSHAPE, "70")
  Call ShapeAddNew(dtSHAPE, "10")
  For i = 1 To 10
    Call ShapeAddNew(dtSHAPE, "0")
    Call ShapeAddNew(dtSHAPE, "LAYER")
    Call ShapeAddNew(dtSHAPE, "2")
    Call ShapeAddNew(dtSHAPE, Format$(i, "0"))
    Call ShapeAddNew(dtSHAPE, "70")
    Call ShapeAddNew(dtSHAPE, "0")
    Call ShapeAddNew(dtSHAPE, "62")
    Call ShapeAddNew(dtSHAPE, "7")
    Call ShapeAddNew(dtSHAPE, "6")
    Call ShapeAddNew(dtSHAPE, "CONTINUOUS")
  Next i
  Call ShapeAddNew(dtSHAPE, "0")
  Call ShapeAddNew(dtSHAPE, "ENDTAB")
  Call ShapeAddNew(dtSHAPE, "0")
  Call ShapeAddNew(dtSHAPE, "ENDSEC")
  'BLOCKステーションの定義
  Call ShapeAddNew(dtSHAPE, "0")
  Call ShapeAddNew(dtSHAPE, "SECTION")
  Call ShapeAddNew(dtSHAPE, "2")
  Call ShapeAddNew(dtSHAPE, "BLOCKS")
  Call ShapeAddNew(dtSHAPE, "0")

```

```

Call ShapeAddNew(dtSHAPE, "BLOCK")
Call ShapeAddNew(dtSHAPE, "8") '面層 9
Call ShapeAddNew(dtSHAPE, "9")
Call ShapeAddNew(dtSHAPE, "2")
Call ShapeAddNew(dtSHAPE, "SHAPE")
Call ShapeAddNew(dtSHAPE, "70")
Call ShapeAddNew(dtSHAPE, "0")
Call ShapeAddNew(dtSHAPE, "10")
Call ShapeAddNew(dtSHAPE, "0.0")
Call ShapeAddNew(dtSHAPE, "20")
Call ShapeAddNew(dtSHAPE, "0.0")

Do Until dtEX_SHAPE.EOF

' 形状データの書き込み
Call ShapeAddNew(dtSHAPE, "0")
Call ShapeAddNew(dtSHAPE, "LINE")
Call ShapeAddNew(dtSHAPE, "8 ") '面層 9
Call ShapeAddNew(dtSHAPE, "9")
Call ShapeAddNew(dtSHAPE, "62") '色 灰
Call ShapeAddNew(dtSHAPE, "8")
Call ShapeAddNew(dtSHAPE, "10") '始点 r
Call ShapeAddNew(dtSHAPE, dtEX_SHAPE.Fields(0).Value)
Call ShapeAddNew(dtSHAPE, "20") '始点 z
Call ShapeAddNew(dtSHAPE, dtEX_SHAPE.Fields(1).Value)
dtEX_SHAPE.MoveNext
Call ShapeAddNew(dtSHAPE, "11") '終点 r
Call ShapeAddNew(dtSHAPE, dtEX_SHAPE.Fields(0).Value)
Call ShapeAddNew(dtSHAPE, "21") '終点 z
Call ShapeAddNew(dtSHAPE, dtEX_SHAPE.Fields(1).Value)

dtEX_SHAPE.MoveNext

Loop

Call ShapeAddNew(dtSHAPE, "0")
Call ShapeAddNew(dtSHAPE, "ENDBLK")
Call ShapeAddNew(dtSHAPE, "8") '面層 9
Call ShapeAddNew(dtSHAPE, "9")
Call ShapeAddNew(dtSHAPE, "0")
Call ShapeAddNew(dtSHAPE, "ENDSEC")

' ENTITIESセクションの記述
Call ShapeAddNew(dtSHAPE, "0")
Call ShapeAddNew(dtSHAPE, "SECTION")
Call ShapeAddNew(dtSHAPE, "2")
Call ShapeAddNew(dtSHAPE, "ENTITIES")
Call ShapeAddNew(dtSHAPE, "0")
Call ShapeAddNew(dtSHAPE, "INSERT")
Call ShapeAddNew(dtSHAPE, "8") '面層 9
Call ShapeAddNew(dtSHAPE, "9")
Call ShapeAddNew(dtSHAPE, "2")
Call ShapeAddNew(dtSHAPE, "SHAPE")
Call ShapeAddNew(dtSHAPE, "10")
Call ShapeAddNew(dtSHAPE, "0.0")
Call ShapeAddNew(dtSHAPE, "20")
Call ShapeAddNew(dtSHAPE, "0.0")
Call ShapeAddNew(dtSHAPE, "0")
Call ShapeAddNew(dtSHAPE, "ENDSEC")
Call ShapeAddNew(dtSHAPE, "0")
Call ShapeAddNew(dtSHAPE, "EOF")

End If

' 形状データのクローズ
dtSHAPE.Close

' 外部ファイルをクローズ
dtEX_SHAPE.Close
dbEX_DATA.Close

Screen.MousePointer = 0

End Sub

Public Sub ShapeStfToCsv()
'+++++++
' このメソッドは現在使用していない: 1998/01/27
'+++++++
Dim Buffer$, Buf10$, BufNew$, cTitle$
Dim i As Long
Dim j As Long
Dim k As Long
Dim Row As Long
Dim Col As Long
Dim ShapeFieldCount As Long
Dim StfFile$, CsvFile$
Dim Sheet(600, 3) As Double
Dim StfName$(3)

Screen.MousePointer = 11

ShapeFieldCount = 30

```

```

StfName$(0) = "MODEL-R"
StfName$(1) = "MODEL-Z"
StfName$(2) = "MODEL-H"

StfFile$ = tmpInpFName$(0)
CsvFile$ = Left$(StfFile$, Len(StfFile$) - 3) & "CSV"

Debug.Print StfFile$, CsvFile$

Row = 0

Open StfFile$ For Input As #1
Do While Not EOF(1)
  Line Input #1, Buffer$

  If Left$(Buffer$, 1) = "$" Then
    'コメント行はスキップ'

  ElseIf Left$(Buffer$, 1) = "P" Then
    'PTITLE名を表示'

  ElseIf Mid$(Buffer$, 2, 5) = "TITLE" Then
    cTitle$ = Mid$(Buffer$, 11, 10)
    'XYTITLE名を表示'
    For j = 0 To ShapeFieldCount - 1
      If cTitle$ = StfName$(j) Then
        Row = 0
        Col = j
        Exit For
      End If
    Next j

  Else
    For i = 1 To 6
      Buf10$ = Mid$(Buffer$, i * 10 + 11, 10)
      If Buf10$ = "" Or Buf10$ = " " Then
        Exit For
      End If

      'If Row >= SheetMax Then
      '  Call ErrorMessage
      'End If

      Call AddExponent(Buf10$, BufNew$)
      Sheet(Row).Value(Col) = Val(BufNew$)
      Sheet(Row, Col) = Val(BufNew$)
      Row = Row + 1
    Next i

  End If

Loop

Close #1

'-----
Open CsvFile$ For Output As #1

Print #1, StfName$(0);
For j = 1 To Col
  Print #1, ","; StfName$(j);
Next j
Print #1, Chr$(13); Chr$(10);

For i = 0 To Row - 1
  Print #1, Sheet(i, 0);
  For j = 1 To Col
    Print #1, ","; Sheet(i, j);
  Next j
  Print #1, Chr$(13); Chr$(10);
Next i

Close #1

Screen.MousePointer = 0

End Sub

Public Sub TempStfToCsv()
'+++++
'このルーチンは現在使用していない: 1998/01/27
'+++++
Dim Buffer$, Buf10$, BufNew$, cTitle$
Dim i As Long
Dim j As Long
Dim k As Long
Dim Row As Long
Dim Col As Long
Dim TempFieldCount As Long
Dim StfFile$, CsvFile$
Dim Sheet(400, 30) As Double

```

```

Dim StfName$(30)

Screen.MousePointer = 11
TempFieldCount = 30
For i = 0 To 9
  StfName$(i * 3 + 0) = "TIME" & i + 1
  StfName$(i * 3 + 1) = "TEMP" & i + 1
  StfName$(i * 3 + 2) = "CONV" & i + 1
Next i

StfFile$ = tmpInpFName$(0)
CsvFile$ = Left$(StfFile$, Len(StfFile$) - 3) & "CSV"

Debug.Print StfFile$, CsvFile$

Row = 0
Open StfFile$ For Input As #1
Do While Not EOF(1)
  Line Input #1, Buffer$

  If Left$(Buffer$, 1) = "$" Then
    'コメント行はスキップ

  ElseIf Left$(Buffer$, 1) = "P" Then
    'PTITLE名を表示

  ElseIf Mid$(Buffer$, 2, 5) = "TITLE" Then
    cTitle$ = Mid$(Buffer$, 11, 10)
    'XYTITLE名を表示
    For j = 0 To TempFieldCount - 1
      If cTitle$ = StfName$(j) Then
        Row = 0
        Col = j
        Exit For
      End If
    Next j

  Else
    For i = 1 To 6
      Buf10$ = Mid$(Buffer$, i * 10 + 11, 10)
      If Buf10$ = "" Or Buf10$ = " " Then
        Exit For
      End If

      'If Row >= SheetMax Then
      '  Call ErrorMessage
      'End If

      Call AddExponent(Buf10$, BufNew$)
      Sheet(Row).Value(Col) = Val(BufNew$)
      Sheet(Row, Col) = Val(BufNew$)
      Row = Row + 1
    Next i

  End If
Loop
Close #1

'-----
Open CsvFile$ For Output As #1
Print #1, StfName$(0);
For j = 1 To Col
  Print #1, ","; StfName$(j);
Next j
Print #1, Chr$(13); Chr$(10);

For i = 0 To Row - 1
  Print #1, Sheet(i, 0);
  For j = 1 To Col
    Print #1, ","; Sheet(i, j);
  Next j
  Print #1, Chr$(13); Chr$(10);
Next i

Close #1

Screen.MousePointer = 0

End Sub

Public Sub UserCheck()
'+++++++
'このルーチンは現在使用していない: 1998/01/28
'+++++++

```

```

Dim UFile$, UType$

UFile$ = modSTAR.DBDir$ & "STARUSER.$$$"

On Error GoTo User

Open UFile$ For Output As #1

Close #1
Kill UFile$
modSTAR.SuperVisor = True
UType$ = "管理者"
GoTo Cont

User:
modSTAR.SuperVisor = False
UType$ = "ユーザ"

Cont:
On Error GoTo Busy

UFile$ = modSTAR.DBDir$ & "STAR.USR"

If modSTAR.SuperVisor = True Then
  Open UFile$ For Output Lock Read Write As #9
Else
  Open UFile$ For Input Lock Write As #9
End If

Exit Sub

Busy:
MsgBox UType$ & "が実行中です。作業を中止します。"
End

End Sub

Public Sub SecTableCalc5(nmID As Long, nmAID As Long, nmCase$, IFERR As Integer)
  Dim dtANL As rdoResultset
  Dim dtTBL As rdoResultset 'データ用のテーブル上の表計算テーブル
  Dim Anal(1, 100) As Double
  Dim Jiten1(1, 100) As Double
  Dim Jiten2(1, 100) As Double
  Dim Table(1, 100) As Double
  Dim JVal As Variant
  Dim AnalRows As Long
  Dim j As Long
  Dim k As Long
  Dim SQL$
  Dim obj As Object
  Dim hANL As Integer
  Dim hTBL As Integer
  Dim AAll As Long
  Dim JAll As Long

  On Error GoTo Err1

  IFERR = 0

  '解析テーブルと時点テーブルのフィールド数の取得
  SQL$ = "SELECT * FROM 解析 WHERE id = 0;"
  Set dtANL = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hANL)
  If hANL = -1 Then GoTo Err1
  AAll = dtANL.rdoColumns.Count
  xCloseResultset dtANL, hANL
  SQL$ = "SELECT * FROM 時点 WHERE id = 0;"
  Set dtANL = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hANL)
  If hANL = -1 Then GoTo Err1
  JAll = dtANL.rdoColumns.Count
  xCloseResultset dtANL, hANL
  Debug.Print AAll; JAll

  '表計算カーソルの起動
  Set obj = CreateObject("TCALCSV.TCALC")

  '古い表計算テーブルを削除しておく
  'SQL$ = "DELETE 表計算 WHERE ("
  ' & "(参照id = " & nmID & ") "
  ' & " AND (解析ケース = " & nmCase$ & "));"
  'dbSTAR5.Execute SQL$
  'Call DeleteDataRC("表計算", "解析id", nmAID, nmCase$)
  Call DeleteDataA("表計算", nmAID)

  '解析テーブルをオープン
  Debug.Print "解析テーブルをオープン"
  'SQL$ = "SELECT * FROM 解析 WHERE ("
  ' & "(参照id = " & nmID & ") "
  ' & " AND (解析ケース = " & nmCase$ & "));"
  'SQL$ = "SELECT 解析.*, 時点1.*, 時点2.* "
  ' & "FROM (解析 INNER JOIN 時点 AS 時点1 ON 解析.時点1id = 時点1.id) "
  ' & "INNER JOIN 時点 AS 時点2 ON 解析.時点2id = 時点2.id "
  ' & "WHERE (解析.解析id = " & nmAID & ") "

```



```

& ” ORDER BY 解析.ライク番号, 解析.Li”
’Set dtANL = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
Set dtANL = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hANL)
If hANL = -1 Then GoTo Err1

```

```

If dtANL.EOF = False Then

```

```

’行数の取得
dtANL.MoveLast
AnalRows = dtANL.RowCount
dtANL.MoveFirst
obj.AnalRows = AnalRows
’ReDim Anal(AnalRows, 100)
’ReDim Jiten1(AnalRows, 100)
’ReDim Jiten2(AnalRows, 100)
’ReDim Table(AnalRows, 100)

```

```

k = 0

```

```

Do Until dtANL.EOF

```

```

Anal(k, A_id) = dtANL![id]
Anal(k, A_Rid) = dtANL![参照id]
Anal(k, A_Aid) = dtANL![解析id]
Anal(k, A_DATE) = dtANL![日付]
Anal(k, A_CASE) = dtANL![解析ケース]
Anal(k, A_LINENO) = dtANL![ライク番号]
Anal(k, A_L) = dtANL![L]
Anal(k, A_Q) = dtANL![Q]
Anal(k, A_r1) = dtANL![起点r1]
Anal(k, A_z1) = dtANL![起点z1]
Anal(k, A_θ1) = dtANL![起点θ1]
Anal(k, A_r2) = dtANL![終点r2]
Anal(k, A_z2) = dtANL![終点z2]
Anal(k, A_θ2) = dtANL![終点θ2]
Anal(k, A_TEMPMAX) = dtANL![TEMPMAX]
’Anal(k, A_J1id) = dtANL![時点1id]
’Anal(k, A_J2id) = dtANL![時点2id]
’Anal(k, A_Cid) = dtANL![試験条件id]

```

```

’Jiten1(k, J_id) = dtANL![id]
’Jiten1(k, J_TIME) = dtANL![評価時点]
’Jiten1(k, J_TEMP) = dtANL![TEMP]
’Jiten1(k, J_LTEMP) = dtANL![LTEMP]
’Jiten1(k, J_MTEMP) = dtANL![MTEMP]
’Jiten1(k, J_σr) = dtANL![σr]
’Jiten1(k, J_σz) = dtANL![σz]
’Jiten1(k, J_σθ) = dtANL![σθ]
’Jiten1(k, J_τrz) = dtANL![τrz]
’Jiten1(k, J_Lσr) = dtANL![Lσr]
’Jiten1(k, J_Lσz) = dtANL![Lσz]
’Jiten1(k, J_Lσθ) = dtANL![Lσθ]
’Jiten1(k, J_Lτrz) = dtANL![Lτrz]
’Jiten1(k, J_Mσr) = dtANL![Mσr]
’Jiten1(k, J_Mσz) = dtANL![Mσz]
’Jiten1(k, J_Mσθ) = dtANL![Mσθ]
’Jiten1(k, J_Mτrz) = dtANL![Mτrz]
’Jiten1(k, J_VMS) = dtANL![VMS]
’Jiten1(k, J_TRES) = dtANL![TRES]
’Jiten1(k, J_εr) = dtANL![εr]
’Jiten1(k, J_εz) = dtANL![εz]
’Jiten1(k, J_εθ) = dtANL![εθ]
’Jiten1(k, J_γrz) = dtANL![γrz]
’Jiten1(k, J_εpr) = dtANL![εpr]
’Jiten1(k, J_εpz) = dtANL![εpz]
’Jiten1(k, J_εpθ) = dtANL![εpθ]
’Jiten1(k, J_γprz) = dtANL![γprz]
’Jiten1(k, J_εcr) = dtANL![εcr]
’Jiten1(k, J_εcz) = dtANL![εcz]
’Jiten1(k, J_εcθ) = dtANL![εcθ]
’Jiten1(k, J_γcrz) = dtANL![γcrz]

```

```

For j = 0 To J_ALL - 1

```

```

For j = 0 To JA11 - 1

```

```

’JVal = dtANL.rdoColumns(A_ALL + j).Value

```

```

JVal = dtANL.rdoColumns(AA11 + j).Value

```

```

If IsNull(JVal) = True Then

```

```

Jiten1(k, j) = 0

```

```

Else

```

```

Jiten1(k, j) = JVal

```

```

End If

```

```

Next j

```

```

’Jiten2(k, J_id) = dtANL![id]
’Jiten2(k, J_TIME) = dtANL![時点2.評価時点]
’Jiten2(k, J_TEMP) = dtANL![TEMP]
’Jiten2(k, J_LTEMP) = dtANL![LTEMP]
’Jiten2(k, J_MTEMP) = dtANL![MTEMP]
’Jiten2(k, J_σr) = dtANL![σr]
’Jiten2(k, J_σz) = dtANL![σz]
’Jiten2(k, J_σθ) = dtANL![σθ]
’Jiten2(k, J_τrz) = dtANL![τrz]
’Jiten2(k, J_Lσr) = dtANL![Lσr]
’Jiten2(k, J_Lσz) = dtANL![Lσz]
’Jiten2(k, J_Lσθ) = dtANL![Lσθ]
’Jiten2(k, J_Lτrz) = dtANL![Lτrz]
’Jiten2(k, J_Mσr) = dtANL![Mσr]
’Jiten2(k, J_Mσz) = dtANL![Mσz]
’Jiten2(k, J_Mσθ) = dtANL![Mσθ]
’Jiten2(k, J_Mτrz) = dtANL![Mτrz]
’Jiten2(k, J_VMS) = dtANL![VMS]

```

```

' Jiten2(k, J_TRES) = dtANL![TRES]
' Jiten2(k, J_er) = dtANL![er]
' Jiten2(k, J_ez) = dtANL![ez]
' Jiten2(k, J_eθ) = dtANL![eθ]
' Jiten2(k, J_γrz) = dtANL![γrz]
' Jiten2(k, J_εpr) = dtANL![εpr]
' Jiten2(k, J_εpz) = dtANL![εpz]
' Jiten2(k, J_εpθ) = dtANL![εpθ]
' Jiten2(k, J_γprz) = dtANL![γprz]
' Jiten2(k, J_εcr) = dtANL![εcr]
' Jiten2(k, J_εcz) = dtANL![εcz]
' Jiten2(k, J_εcθ) = dtANL![εcθ]
' Jiten2(k, J_γcrz) = dtANL![γcrz]
For j = 0 To J_ALL - 1
For j = 0 To JA11 - 1
  JVal = dtANL.rdoColumns(A_ALL + J_ALL + j).Value
  JVal = dtANL.rdoColumns(AA11 + JA11 + j).Value
  If IsNull(JVal) = True Then
    Jiten2(k, j) = 0
  Else
    Jiten2(k, j) = JVal
  End If
End If
Next j
' Debug.Print Jiten1(k, J_TIME); Jiten2(k, J_TIME)

k = k + 1
dtANL.MoveNext
Loop

' 計算実行
Debug.Print "Exec Start"
obj.Exec Anal, Jiten1, Jiten2, Table
Debug.Print "Exec End"

' 表計算結果の取得
SQL$ = "SELECT * FROM 表計算 WHERE " & "解析id = " & nmAID & ";"
' Set dtTBL = dbSTAR5.OpenResultset(SQL$, rdOpenDynamic, rdConcurLock)
Set dtTBL = xOpenResultset(SQL$, rdOpenDynamic, rdConcurLock, hTBL)
If hTBL = -1 Then GoTo Err1

dtANL.MoveFirst
For k = 0 To AnalRows - 1
  dtTBL.AddNew
  dtTBL![id] = Table(k, T_id)
  dtTBL![参照id] = Table(k, T_Rid)
  dtTBL![解析id] = Table(k, T_Aid)
  ' dtTBL![日付] = Table(k, T_DATE)
  dtTBL![日付] = Date
  dtTBL![解析ケース] = Table(k, T_CASE)
  dtTBL![ライン番号] = Table(k, T_LINENO)
  dtTBL![L] = Table(k, T_L)
  dtTBL![起点r1] = Table(k, T_r1)
  dtTBL![起点z1] = Table(k, T_z1)
  dtTBL![起点θ1] = Table(k, T_θ1)
  dtTBL![dσr] = Table(k, T_dσr)
  dtTBL![dσz] = Table(k, T_dσz)
  dtTBL![dσθ] = Table(k, T_dσθ)
  dtTBL![dτrz] = Table(k, T_dτrz)
  dtTBL![dLσr] = Table(k, T_dLσr)
  dtTBL![dLσz] = Table(k, T_dLσz)
  dtTBL![dLσθ] = Table(k, T_dLσθ)
  dtTBL![dLτrz] = Table(k, T_dLτrz)
  dtTBL![dMσr] = Table(k, T_dMσr)
  dtTBL![dMσz] = Table(k, T_dMσz)
  dtTBL![dMσθ] = Table(k, T_dMσθ)
  dtTBL![dMτrz] = Table(k, T_dMτrz)
  dtTBL![dσ1] = Table(k, T_dσ1)
  dtTBL![dσ2] = Table(k, T_dσ2)
  dtTBL![dσ3] = Table(k, T_dσ3)
  dtTBL![dLσ1] = Table(k, T_dLσ1)
  dtTBL![dLσ2] = Table(k, T_dLσ2)
  dtTBL![dLσ3] = Table(k, T_dLσ3)
  dtTBL![dMσ1] = Table(k, T_dMσ1)
  dtTBL![dMσ2] = Table(k, T_dMσ2)
  dtTBL![dMσ3] = Table(k, T_dMσ3)
  dtTBL![Δσr] = Table(k, T_Δσr)
  dtTBL![Δσz] = Table(k, T_Δσz)
  dtTBL![Δσθ] = Table(k, T_Δσθ)
  dtTBL![Δτrz] = Table(k, T_Δτrz)
  dtTBL![ΔLσr] = Table(k, T_ΔLσr)
  dtTBL![ΔLσz] = Table(k, T_ΔLσz)
  dtTBL![ΔLσθ] = Table(k, T_ΔLσθ)
  dtTBL![ΔLτrz] = Table(k, T_ΔLτrz)
  dtTBL![ΔMσr] = Table(k, T_ΔMσr)
  dtTBL![ΔMσz] = Table(k, T_ΔMσz)
  dtTBL![ΔMσθ] = Table(k, T_ΔMσθ)
  dtTBL![ΔMτrz] = Table(k, T_ΔMτrz)
  dtTBL![Δσ1] = Table(k, T_Δσ1)
  dtTBL![Δσ2] = Table(k, T_Δσ2)
  dtTBL![Δσ3] = Table(k, T_Δσ3)
  dtTBL![ΔLσ1] = Table(k, T_ΔLσ1)
  dtTBL![ΔLσ2] = Table(k, T_ΔLσ2)
  dtTBL![ΔLσ3] = Table(k, T_ΔLσ3)
  dtTBL![ΔMσ1] = Table(k, T_ΔMσ1)
  dtTBL![ΔMσ2] = Table(k, T_ΔMσ2)
  dtTBL![ΔMσ3] = Table(k, T_ΔMσ3)

```

```

dtTBL![ΔVMS] = Table(k, T_ΔVMS)
dtTBL![ΔTRES] = Table(k, T_ΔTRES)
dtTBL![ΔRANK] = Table(k, T_ΔRANK)
dtTBL![ΔLVMS] = Table(k, T_ΔLVMS)
dtTBL![ΔLTRES] = Table(k, T_ΔLTRES)
dtTBL![ΔLRANK] = Table(k, T_ΔLRANK)
dtTBL![ΔMVMS] = Table(k, T_ΔMVMS)
dtTBL![ΔMTRES] = Table(k, T_ΔMTRES)
dtTBL![ΔMRANK] = Table(k, T_ΔMRANK)
dtTBL![dεr] = Table(k, T_dεr)
dtTBL![dεz] = Table(k, T_dεz)
dtTBL![dεθ] = Table(k, T_dεθ)
dtTBL![dγrz] = Table(k, T_dγrz)
dtTBL![dεpr] = Table(k, T_dεpr)
dtTBL![dεpz] = Table(k, T_dεpz)
dtTBL![dεpθ] = Table(k, T_dεpθ)
dtTBL![dγprz] = Table(k, T_dγprz)
dtTBL![dεcr] = Table(k, T_dεcr)
dtTBL![dεcz] = Table(k, T_dεcz)
dtTBL![dεcθ] = Table(k, T_dεcθ)
dtTBL![dγcrz] = Table(k, T_dγcrz)
dtTBL![ΔEEPCEQ] = Table(k, T_ΔEEPCEQ)
dtTBL![ΔEPEQ] = Table(k, T_ΔEPEQ)
dtTBL![ΔECEQ] = Table(k, T_ΔECEQ)
dtTBL.Update

dtANL.MoveNext
Next k

'dtTBL.Close
xCloseResultset dtTBL, hTBL

End If

'dtANL.Close
xCloseResultset dtANL, hANL

Set obj = Nothing

Exit Sub

Err1:
MsgBox "Error in modSTAR/SecTableCalc5"
Call ShowErrorMessage
On Error Resume Next
'dtANL.Close
'dtTBL.Close
xCloseResultset dtANL, hANL
xCloseResultset dtTBL, hTBL
Set obj = Nothing

End Sub

Public Function GetHyokaID(nmHyoka$) As Long
Dim dtHH As rdoResultset
Dim SQL$
Dim hHH As Integer

On Error GoTo Err1

SQL$ = "SELECT id FROM 評価法マスタ WHERE " -
& "(評価法名 = " & nmHyoka$ & " );"
' Set dtHH = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
Set dtHH = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hHH)
If hHH = -1 Then GoTo Err1

If dtHH.EOF = False Then
GetHyokaID = dtHH![id]
Else
GetHyokaID = 0
End If

'dtHH.Close
xCloseResultset dtHH, hHH

Exit Function

Err1:
xCloseResultset dtHH, hHH
GetHyokaID = 0

End Function

Public Function GetHyokaName(Hid As Long) As String
Dim dtHH As rdoResultset
Dim SQL$
Dim hHH As Integer

On Error GoTo Err1

```

```

SQL$ = "SELECT 評価法名 FROM 評価法テーブル WHERE " _
      & "(id = " & Hid & ");"
' Set dtHH = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
Set dtHH = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hHH)
If hHH = -1 Then GoTo Err1

If dtHH.EOF = False Then
  GetHyokaName = dtHH![評価法名]
Else
  GetHyokaName = ""
End If

'dtHH.Close
xCloseResultset dtHH, hHH

Exit Function

Err1:
xCloseResultset dtHH, hHH
GetHyokaName = ""

End Function

Public Sub SetAnalField(APos As Long, FType As Long, BPos As Long, Name0$, Name1$, Name2$, DName$)

  modSTAR.AFType(APos, 0) = FType      '格納するテーブル(0:解析/1:時点1/2:時点2)
  modSTAR.AFType(APos, 1) = BPos      '格納するフィールド位置
  modSTAR.AFName$(APos, 0) = Name0$   '外部フィールド名その1
  modSTAR.AFName$(APos, 1) = Name1$   '外部フィールド名その2(DBのフィールド名)
  modSTAR.AFName$(APos, 2) = Name2$   '外部フィールド名その3
  modSTAR.AFName$(APos, 3) = DName$   'DB上のフィールド名

End Sub

Public Sub TransAnalField(EXName$, TblType As Long, DBName$)
' EXName$: 外部フィールド名
' TblType: 出力テーブル(0:解析/1:時点1/2:時点2)
' DBName$: DB側のフィールド名
Dim i As Long

TblType = -1

For i = 0 To modSTAR.AFldMax - 1
  If EXName$ = modSTAR.AFName$(i, 0) Or _
    EXName$ = modSTAR.AFName$(i, 1) Or _
    EXName$ = modSTAR.AFName$(i, 2) Then
    TblType = modSTAR.AFType(i, 0)
    DBName$ = modSTAR.AFName$(i, 1)
    DBName$ = modSTAR.AFName$(i, 3)
  Exit For
End If
Next i

End Sub

Public Sub DeleteAnalA(Anal$, Jiten$, nmAID As Long)
' Dim dtANAL As rdoResultset
' Dim SQL$
' Dim hANAL As Integer

On Error GoTo Err1

' SQL$ = "SELECT 時点1id,時点2id FROM " & Anal$ & " WHERE." _
'       & "(解析id = " & nmAID & ");"
' Set dtANAL = dbSTAR5.OpenResultset(SQL$, rdOpenDynamic, rdConcurLock)
' Set dtANAL = xOpenResultset(SQL$, rdOpenDynamic, rdConcurLock, hANAL)
' If hANAL = -1 Then GoTo Err1

' Do Until dtANAL.EOF
'   Call DeleteJiten(Jiten$, dtANAL![時点1id], dtANAL![時点2id])
'   dtANAL.MoveNext
' Loop

'dtANAL.Close
xCloseResultset dtANAL, hANAL

Call DeleteDataA(Anal$, nmAID)
Call DeleteDataA(Jiten$, nmAID)

Exit Sub

Err1:
xCloseResultset dtANAL, hANAL

End Sub

Public Sub DeleteAnalR(Anal$, Jiten$, nmID As Long)

```

```

'Dim dtANAL As rdoResultset
'Dim SQL$
'Dim hANAL As Integer

On Error GoTo Err1

SQL$ = "SELECT 時点1id,時点2id FROM " & Anal$ & " WHERE " _
      & "(参照id = " & nmID & ");"
Set dtANAL = dbSTAR5.OpenResultset(SQL$, rdOpenDynamic, rdConcurLock)
Set dtANAL = xOpenResultset(SQL$, rdOpenDynamic, rdConcurLock, hANAL)
If hANAL = -1 Then GoTo Err1

Do Until dtANAL.EOF
    Call DeleteJiten(Jiten$, dtANAL![時点1id], dtANAL![時点2id])
    dtANAL.MoveNext
Loop

dtANAL.Close
xCloseResultset dtANAL, hANAL

Call DeleteDataR(Anal$, nmID)
Call DeleteDataR(Jiten$, nmID)

Exit Sub

Err1:
xCloseResultset dtANAL, hANAL

End Sub

```

```

Public Sub DeleteAnalC(Anal$, Jiten$, nmCase$)
'Dim dtANAL As rdoResultset
'Dim SQL$
'Dim hANAL As Integer

On Error GoTo Err1

SQL$ = "SELECT 時点1id,時点2id FROM " & Anal$ & " WHERE " _
      & "(解析ケース = " & nmCase & ");"
Set dtANAL = dbSTAR5.OpenResultset(SQL$, rdOpenDynamic, rdConcurLock)
Set dtANAL = xOpenResultset(SQL$, rdOpenDynamic, rdConcurLock, hANAL)
If hANAL = -1 Then GoTo Err1

Do Until dtANAL.EOF
    Call DeleteJiten(Jiten$, dtANAL![時点1id], dtANAL![時点2id])
    dtANAL.MoveNext
Loop

dtANAL.Close
xCloseResultset dtANAL, hANAL

Call DeleteDataC(Anal$, nmCase$)
Call DeleteDataC(Jiten$, nmCase$)

Exit Sub

Err1:
xCloseResultset dtANAL, hANAL

End Sub

```

```

Public Sub DeleteAnalRC(Anal$, Jiten$, nmAID As Long, nmCase$)
'Dim dtANAL As rdoResultset
'Dim SQL$
'Dim hANAL As Integer

On Error GoTo Err1

SQL$ = "SELECT 時点1id,時点2id FROM " & Anal$ & " WHERE " _
      & "(解析id = " & nmAID & ");"
Set dtANAL = dbSTAR5.OpenResultset(SQL$, rdOpenDynamic, rdConcurLock)
Set dtANAL = xOpenResultset(SQL$, rdOpenDynamic, rdConcurLock, hANAL)
If hANAL = -1 Then GoTo Err1

Do Until dtANAL.EOF
    Call DeleteJiten(Jiten$, dtANAL![時点1id], dtANAL![時点2id])
    dtANAL.MoveNext
Loop

dtANAL.Close
xCloseResultset dtANAL, hANAL

Call DeleteDataRC(Anal$, "解析id", nmAID, nmCase$)

Exit Sub

Err1:
MsgBox "Error in modSTAR/DeleteAnalRC"

```

```

' On Error GoTo 0
' dtANAL.Close
' xCloseResultset dtANAL, hANAL
'End Sub

```

```

Public Sub DeleteDataA(nmTABLE$, nmAID As Long)
Dim SQL$

```

```

On Error GoTo Err1

```

```

SQL$ = "DELETE " & nmTABLE$ & " WHERE " _
& "(解析id = " & nmAID & ");"
Debug.Print "DeleteDataA "; SQL$
dbSTAR5.Execute SQL$

```

```

Exit Sub

```

```

Err1:
MsgBox "Error in modSTAR/DeleteDataA"

```

```

End Sub

```

```

Public Sub DeleteDataR(nmTABLE$, nmID As Long)
Dim SQL$

```

```

On Error GoTo Err1

```

```

SQL$ = "DELETE " & nmTABLE$ & " WHERE " _
& "(参照id = " & nmID & ");"
Debug.Print "DeleteDataR "; SQL$
dbSTAR5.Execute SQL$

```

```

Exit Sub

```

```

Err1:
MsgBox "Error in modSTAR/DeleteDataR"

```

```

End Sub

```

```

Public Sub DeleteDataC(nmTABLE$, nmCase$)
Dim SQL$

```

```

SQL$ = "DELETE " & nmTABLE$ & " WHERE " _
& "(解析ケース = " & nmCase$ & ");"
Debug.Print "DeleteDataC "; SQL$
dbSTAR5.Execute SQL$

```

```

End Sub

```

```

'Public Sub DeleteDataRC(nmTABLE$, nmFLD$, nmID As Long, nmCase$)
' Dim SQL$

```

```

' On Error GoTo Err1

```

```

' SQL$ = "DELETE " & nmTABLE$ & " WHERE " _
' & "(" & nmFLD$ & " = " & nmID & ")" AND " _
' & "(解析ケース = " & nmCase$ & ");"
' Debug.Print "削除 RC "; SQL$
' dbSTAR5.Execute SQL$

```

```

' Exit Sub

```

```

'Err1:
' MsgBox "Error in modSTAR/DeleteDataRC"

```

```

'End Sub

```

```

Public Sub DeleteJiten(Jiten$, i1 As Long, i2 As Long)
Dim SQL$

```

```

On Error GoTo Err1

```

```

SQL$ = "DELETE " & Jiten$ & " WHERE (id IN (" & i1 & ", " & i2 & "));"
Debug.Print "DeleteJiten "; i1; i2
dbSTAR5.Execute SQL$

```

```

Exit Sub

```

```

Err1:
MsgBox "Error in modSTAR/DeleteJiten"

```

```

End Sub

```

```

Public Sub SetKanriXID(FName$, FVal As Long, nmID As Long, nmCase$, nmHID As Long)
Dim dtKANRI As rdoResultset

```

```

Dim SQL$
Dim hKANRI As Integer

On Error GoTo Err1

If nmHID > 0 Then ' 損傷値idの設定
  SQL$ = "SELECT " & FName$ & " FROM 管理データテーブル WHERE " _
    & "((参照id = " & nmID & ") AND " _
    & "(解析ケース = " & nmCase$ & ") AND " _
    & "(評価法id = " & nmHID & "));"
ElseIf nmCase$ <> "-" Then ' 解析・表計算idの設定
  SQL$ = "SELECT " & FName$ & " FROM 管理データテーブル WHERE " _
    & "((参照id = " & nmID & ") AND " _
    & "(解析ケース = " & nmCase$ & "));"
Else ' き裂・試験条件・温度・形状idの設定
  SQL$ = "SELECT " & FName$ & " FROM 管理データテーブル WHERE " _
    & "((参照id = " & nmID & "));"
End If

' Set dtKANRI = dbSTAR5.OpenResultset(SQL$, rdOpenDynamic, rdConcurLock)
Set dtKANRI = xOpenResultset(SQL$, rdOpenDynamic, rdConcurLock, hKANRI)
If hKANRI = -1 Then GoTo Err1

' Do Until dtKANRI.EOF
'   Debug.Print dtKANRI.AbsolutePosition
'   dtKANRI.Edit
'   dtKANRI.rdoColumns(FName$).Value = FVal
'   dtKANRI.Update
'   dtKANRI.Bookmark = dtKANRI.LastModified
'   dtKANRI.MoveNext
' Loop
If dtKANRI.EOF = False Then
  dtKANRI.MoveLast
  Do Until dtKANRI.BOF
    Debug.Print dtKANRI.AbsolutePosition
    dtKANRI.Edit
    dtKANRI.rdoColumns(FName$).Value = FVal
    dtKANRI.Update
    dtKANRI.Bookmark = dtKANRI.LastModified
    dtKANRI.MovePrevious
  Loop
End If

' dtKANRI.Close
xCloseResultset dtKANRI, hKANRI

Exit Sub

Err1:
MsgBox "Error in modSTAR/SetKanriXID"
Call ShowErrorMessage
On Error Resume Next
' dtKANRI.Close
xCloseResultset dtKANRI, hKANRI

End Sub

Public Sub SetKanriXID(FName$, FVal As Long, nmID As Long, nmCase$, nmHID As Long)
  Dim dtKANRI As rdoResultset
  Dim SQL$
  Dim hKANRI As Integer
  Dim RIDP As Long ' 参照idのフィールド位置
  Dim HIDP As Long ' 評価法idのフィールド位置
  Dim CASP As Long ' 解析ケースのフィールド位置
  Dim TGTP As Long ' 更新フィールドの位置
  Dim i As Long
  Dim IFERR As Integer

  On Error GoTo Err1

  If nmHID > 0 Then ' 損傷値idの設定
    SQL$ = "SELECT " & FName$ & " FROM 管理データテーブル WHERE " _
      & "((参照id = " & nmID & ") AND " _
      & "(解析ケース = " & nmCase$ & ") AND " _
      & "(評価法id = " & nmHID & "));"

    RIDP = KFP("参照id")
    CASP = KFP("解析ケース")
    HIDP = KFP("評価法id")
    TGTP = KFP(FName$)

    For i = 0 To nKanriT - 1
      If KanriT(i, RIDP) = nmID And _
        KanriT(i, CASP) = nmCase$ And _
        KanriT(i, HIDP) = nmHID Then
        KanriT(i, TGTP) = FVal
      End If
    Next i

  ElseIf nmCase$ <> "-" Then ' 解析・表計算idの設定
    SQL$ = "SELECT " & FName$ & " FROM 管理データテーブル WHERE " _
      & "((参照id = " & nmID & ") AND " _

```

```

' & " (解析ケース = " & nmCase$ & "));"

RIDP = KFP("参照id")
CASP = KFP("解析ケース")
TGTP = KFP(FName$)

For i = 0 To nKanriT - 1
  If KanriT(i, RIDP) = nmID And
    KanriT(i, CASP) = nmCase$ Then
    KanriT(i, TGTP) = FVal
  End If
Next i

Else
  ' き裂・試験条件・温度・形状idの設定
  ' SQL$ = "SELECT " & FName$ & " FROM 管理データテーブル WHERE " &
  ' & "(" & "参照id = " & nmID & "));"

  RIDP = KFP("参照id")
  TGTP = KFP(FName$)

  For i = 0 To nKanriT - 1
    If KanriT(i, RIDP) = nmID Then
      KanriT(i, TGTP) = FVal
    End If
  Next i

End If

Call KanriWrite(IFERR)

' Set dtKANRI = dbSTAR5.OpenResultset(SQL$, rdOpenDynamic, rdConcurLock)
' Set dtKANRI = xOpenResultset(SQL$, rdOpenDynamic, rdConcurLock, hKANRI)
' If hKANRI = -1 Then GoTo Err1

' Do Until dtKANRI.EOF
'   Debug.Print dtKANRI.AbsolutePosition
'   dtKANRI.Edit
'   dtKANRI.rdoColumns(FName$).Value = FVal
'   dtKANRI.Update
'   dtKANRI.Bookmark = dtKANRI.LastModified
'   dtKANRI.MoveNext
' Loop
' If dtKANRI.EOF = False Then
'   dtKANRI.MoveLast
'   Do Until dtKANRI.BOF
'     Debug.Print dtKANRI.AbsolutePosition
'     dtKANRI.Edit
'     dtKANRI.rdoColumns(FName$).Value = FVal
'     dtKANRI.Update
'     dtKANRI.Bookmark = dtKANRI.LastModified
'     dtKANRI.MovePrevious
'   Loop
' End If
' dtKANRI.Close
' xCloseResultset dtKANRI, hKANRI

Exit Sub

Err1:
MsgBox "Error in modSTAR/SetKanriXID6"
Call ShowErrorMessage
On Error Resume Next
' dtKANRI.Close
' xCloseResultset dtKANRI, hKANRI

End Sub

Public Function BH2US(Strg$) As String
Dim i As Integer
Dim SLen As Integer

BH2US = Strg$
SLen = Len(Strg$)
For i = 1 To SLen
  If Mid(Strg$, i, 1) = " " Or _
    Mid(Strg$, i, 1) = "-" Then
    Mid(BH2US, i, 1) = "_"
  End If
Next i

End Function

Public Function GetDBData(DBData)

If IsNull(DBData) = True Then
  GetDBData = 0
Else
  GetDBData = DBData
End If

' Debug.Print "GetDBData: "; GetDBData

```


End Function

```
Public Sub StfToCsv(FName$, IFERR As Integer)
  Dim Buffer$, Buf10$, BufNew$, cTitle$
  Dim i As Long
  Dim j As Long
  Dim Row As Long
  Dim Col As Long
  Dim StfFiles$, CsvFile$
  Dim Sheet(200, 100) As Double
  Dim StfName$(100)

  On Error GoTo Err1

  IFERR = 0
  StfFile$ = FName$
  CsvFile$ = Left$(StfFile$, Len(StfFile$) - 3) & "CSV"
```

```
'--- STFファイルの読み込み -----
```

```
Debug.Print StfFile$, CsvFile$

Row = 0
Col = 0

Open StfFile$ For Input As #1

Do While Not EOF(1)
  Line Input #1, Buffer$

  If Left$(Buffer$, 1) = "$" Then
    'コメント行はスキップ

  ElseIf Left$(Buffer$, 1) = "P" Then
    'PTITLE名を表示

  ElseIf Mid$(Buffer$, 2, 5) = "TITLE" Then
    'XTITLE名を表示
    cTitle$ = Mid$(Buffer$, 11, 10)
    Row = 0
    Col = Col + 1
    i = InStr(1, cTitle$, " ", 0)
    If i = 0 Then
      StfName$(Col - 1) = cTitle$
    Else
      StfName$(Col - 1) = Left$(cTitle$, i - 1)
    End If

  Else
    For i = 1 To 6
      Buf10$ = Mid$(Buffer$, i * 10 + 11, 10)
      If Buf10$ = "" Or Buf10$ = " " Then
        Exit For
      End If

      Call AddExponent(Buf10$, BufNew$)
      Sheet(Row).Value(Col) = Val(BufNew$)
      Row = Row + 1
      Sheet(Row - 1, Col - 1) = Val(BufNew$)
    Next i

  End If

Loop

Close #1
```

```
'--- CSVファイルへ書き込み -----
```

```
Open CsvFile$ For Output As #1

Print #1, "DATE";
For j = 0 To Col - 1
  Print #1, ","; StfName$(j);
Next j
Print #1, Chr$(13); Chr$(10);

For i = 0 To Row - 1
  Print #1, Format(Date$, "yy/mm/dd");
  For j = 0 To Col - 1
    Print #1, ","; Sheet(i, j);
  Next j
  Print #1, Chr$(13); Chr$(10);
Next i

Close #1

Exit Sub
```

```
Err1:
MsgBox "Error in modSTAR/StfToCsv"
On Error GoTo 0
Close #1
```

```

IFERR = 1
End Sub

Public Sub KanriSort(InsPos As Long, IFERR As Integer)
    Dim dtKANRI As rdoResultset
    Dim No As Long
    Dim SQL$
    Dim hKANRI As Integer

    On Error GoTo Err1

    IFERR = 0
    No = InsPos

    'SQL$ = "SELECT id FROM 管理テーブル WHERE " _
    & "(id >= " & No & ")";
    SQL$ = "SELECT id FROM 管理テーブル;"

    'Set dtKANRI = dbSTAR5.OpenResultset(SQL$, rdOpenDynamic, rdConcurLock)
    Set dtKANRI = xOpenResultset(SQL$, rdOpenDynamic, rdConcurLock, hKANRI)
    If hKANRI = -1 Then GoTo Err1

    Do Until dtKANRI.EOF
        If dtKANRI![id] >= No Then
            No = No + 1
            dtKANRI.Edit
            dtKANRI![id] = No
            dtKANRI.Update
            dtKANRI.Bookmark = dtKANRI.LastModified
        End If
        dtKANRI.MoveNext
    Loop

    'dtKANRI.Close
    xCloseResultset dtKANRI, hKANRI

    Exit Sub

Err1:
    MsgBox "Error in modSTAR/KanriSort"
    Call ShowErrorMessage
    On Error Resume Next
    'dtKANRI.Close
    xCloseResultset dtKANRI, hKANRI
    IFERR = 1

End Sub

Public Sub KanriSequence(IFERR As Integer)
    Dim dtKANRI As rdoResultset
    Dim SQL$
    Dim hKANRI As Integer
    Dim i As Long

    On Error GoTo Err1

    IFERR = 0

    'SQL$ = "SELECT id FROM 管理テーブル;"
    'Set dtKANRI = dbSTAR5.OpenResultset(SQL$, rdOpenDynamic, rdConcurLock)
    'Set dtKANRI = xOpenResultset(SQL$, rdOpenDynamic, rdConcurLock, hKANRI)
    'If hKANRI = -1 Then GoTo Err1

    'If dtKANRI.EOF = False Then
    '    dtKANRI.MoveLast
    '    Do Until dtKANRI.BOF
    '        dtKANRI.Edit
    '        dtKANRI![id] = dtKANRI.AbsolutePosition - 1
    '        dtKANRI.Update
    '        dtKANRI.Bookmark = dtKANRI.LastModified
    '        dtKANRI.MovePrevious
    '    Loop
    'End If

    'dtKANRI.Close
    'xCloseResultset dtKANRI, hKANRI

    For i = 0 To nKanriT - 1
        KanriT(i, 0) = i
    Next i
    Call KanriWrite(IFERR)
    If IFERR = 1 Then GoTo Err1

    Exit Sub

Err1:
    MsgBox "Error in modSTAR/KanriSequence"
    Call ShowErrorMessage

```

```

On Error Resume Next
dtKANRI.Close
xCloseResultset dtKANRI, hKANRI
IFERR = 1

End Sub

Public Sub KanriAdd(InsPos As Long)
Dim dtKANRI As rdoResultset
Dim SQL$

SQL$ = "SELECT * FROM 管理テーブル;"

Set dtKANRI = dbSTAR5.OpenResultset(SQL$, rdOpenDynamic, rdConcurLock)

dtKANRI.AddNew
dtKANRI![id] = InsPos
dtKANRI![試験装置] = "NewRecord"
dtKANRI![評価件数] = 0
dtKANRI![参照id] = 0
dtKANRI![解析ケース] = "-"
dtKANRI![評価法id] = 0
dtKANRI![き裂id] = 0
dtKANRI![試験条件id] = 0
dtKANRI![温度id] = 0
dtKANRI![形状id] = 0
dtKANRI![湿度id] = 0
dtKANRI![解析id] = 0
dtKANRI![表計算id] = 0
dtKANRI![損傷値id] = 0
dtKANRI![き裂Sid] = 0
dtKANRI![解析Sid] = 0
dtKANRI![表計算Sid] = 0
dtKANRI![損傷値Sid] = 0
dtKANRI.Update

dtKANRI.Close

End Sub

Public Sub KanriRead(IFERR As Integer)
Dim dtKANRI As rdoResultset
Dim SQL$
Dim i As Long
Dim hKANRI As Integer

IFERR = 0

On Error GoTo Err1

'テーブルを配列に読み込む
SQL$ = "管理テーブル"
Set dtKANRI = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hKANRI)
If hKANRI = -1 Then GoTo Err1

cKanriT = dtKANRI.rdoColumns.Count '管理テーブルのフィールド数の取得
For i = 0 To dtKANRI.rdoColumns.Count - 1
KanriFN$(i) = dtKANRI.rdoColumns(i).Name
Next i

nKanriT = 0 '管理テーブルのレコード数の取得
Do Until dtKANRI.EOF
For i = 0 To dtKANRI.rdoColumns.Count - 1
KanriT(nKanriT, i) = dtKANRI.rdoColumns(i).Value
Debug.Print nKanriT; i; KanriT(nKanriT, i)
Next i
nKanriT = nKanriT + 1
dtKANRI.MoveNext
Loop
Debug.Print "管理テーブルの行数と列数 = "; nKanriT; cKanriT

xCloseResultset dtKANRI, hKANRI

Exit Sub

Err1:
MsgBox "Error in modSTAR/KanriRead"
Call ShowErrorMessage
IFERR = 1

End Sub

Public Sub KanriWrite(IFERR As Integer)
Dim dtKANRI As rdoResultset
Dim SQL$
Dim i As Long
Dim j As Long
Dim hKANRI As Integer

IFERR = 0

```

```

On Error GoTo Err1

'管理テーブルを削除
SQL$ = "DELETE 管理テーブル;"
dbSTAR5.Execute SQL$

'配列のデータをテーブルに保存
SQL$ = "管理テーブル"
Set dtKANRI = xOpenResultset(SQL$, rdOpenDynamic, rdConcurLock, hKANRI)

For i = 0 To nKanriT - 1
    dtKANRI.AddNew
    For j = 0 To cKanriT - 1
        dtKANRI.rdoColumns(j).Value = KanriT(i, j)
    Next j
    dtKANRI.Update
Next i
Debug.Print "管理テーブルの行数と列数 = "; nKanriT; cKanriT

xCloseResultset dtKANRI, hKANRI

Exit Sub

Err1:
MsgBox "Error in modSTAR/KanriWrite"
Call ShowErrorMessage
IFERR = 1

End Sub

Public Sub CopyTable(FrTbl$, ToTbl$, IFERR As Integer)
    Dim i As Long
    Dim SQL$

    On Error GoTo Err1

    IFERR = 0

    Screen.MousePointer = 11

    'To側のテーブルを削除しておく
    For i = 0 To dbSTAR5.rdoTables.Count - 1
        Debug.Print dbSTAR5.rdoTables(i).Name
        If dbSTAR5.rdoTables(i).Name = ToTbl$ Then
            SQL$ = "DROP TABLE " & ToTbl$ & ";"
            dbSTAR5.Execute SQL$
        End If
    Next i

    'FrTbl$をToTbl$へコピー
    SQL$ = "SELECT * INTO " & ToTbl$ & " FROM " & FrTbl$ & ";"
    dbSTAR5.Execute SQL$
    dbSTAR5.rdoTables.Refresh

    Screen.MousePointer = 0

    Exit Sub

Err1:
    Screen.MousePointer = 0
    MsgBox "Error in modSTAR/CopyTable"
    Call ShowErrorMessage
    IFERR = 1

End Sub

Public Function GetDBData2(DBData)
    Dim xVal As Variant

    xVal = DBData
    GetDBData2 = xVal

    'Debug.Print "GetDBData2: "; xVal

End Function

```

Attribute VB_Name = "modSTARC"
Option Explicit

```

Public Const C_id = 0      ' id
Public Const C_Rid = 1    ' 参照id
Public Const C_DATANO = 2 ' データ番号
Public Const C_DATE = 3  ' 日付
Public Const C_LINENO = 4 ' ライン番号
Public Const C_r1 = 5    ' 起点r1
Public Const C_z1 = 6    ' 起点z1
Public Const C_θ1 = 7    ' 起点θ1
Public Const C_r2 = 8    ' 終点r2
Public Const C_z2 = 9    ' 終点z2
Public Const C_θ2 = 10   ' 終点θ2
Public Const C_LEN = 11  ' き裂長さ
Public Const C_BYH = 12  ' 母・溶・HAZ
Public Const C_WELD = 13 ' 溶接法
Public Const C_YOMORI = 14 ' 余盛有無
Public Const C_YKEIJO = 15 ' 余盛形状
Public Const C_WBIKO = 16 ' 溶接備考
Public Const C_RYU = 17  ' 粒内・粒界
Public Const C_CKEIJO = 18 ' き裂形状
Public Const C_CYCLE = 19 ' き裂発生サイクル
Public Const C_STRIATE = 20 ' ストライエーション
Public Const C_BMARK = 21 ' ビーマーク
Public Const C_ACTIVE = 22 ' 統計処理アクティブ
Public Const C_CRYSTAL = 23 ' 結晶粒度
Public Const C_BIKO = 24 ' 備考
Public Const C_L = 25    ' L
Public Const C_ALL = 26  ' フィールド総数

Public Const S_id = 0      ' id
Public Const S_Rid = 1    ' 参照id
Public Const S_Sochi = 2  ' 試験装置
Public Const S_Tai = 3    ' 試験体
Public Const S_MATid = 4  ' 材料id
Public Const S_Bui = 5    ' 評価部位
Public Const S_TCOLD = 6  ' COLD温度
Public Const S_THOT = 7  ' HOT温度
Public Const S_HCOLD = 8  ' COLD保持時間
Public Const S_HHOT = 9  ' HOT保持時間
Public Const S_LOAD = 10  ' 機械荷重
Public Const S_CYCLE = 11 ' サイクル数
Public Const S_NODATA = 12 ' データ数
Public Const S_RefName = 13 ' 関連資料名
Public Const S_RefNo = 14 ' 関連資料番号
Public Const S_RefDS = 15 ' 関連データセット
Public Const S_SMAT = 16  ' SMAT TAG NO
Public Const S_MEMO = 17  ' メモ
Public Const S_ALL = 18  ' フィールド総数

Public Const A_id = 0      ' id
Public Const A_Rid = 1    ' 参照id
Public Const A_Aid = 2    ' 解析id
Public Const A_DATE = 3  ' 日付
Public Const A_CASE = 4  ' 解析ケース
Public Const A_LINENO = 5 ' ライン番号
Public Const A_L = 6     ' L
Public Const A_Q = 7     ' q
Public Const A_r1 = 8    ' 起点r1
Public Const A_z1 = 9    ' 起点z1
Public Const A_θ1 = 10   ' 起点θ1
Public Const A_r2 = 11   ' 終点r2
Public Const A_z2 = 12   ' 終点z2
Public Const A_θ2 = 13   ' 終点θ2
Public Const A_TEMPMAX = 14 ' TEMPMAX
Public Const A_J1id = 15  ' 時点1id
Public Const A_J2id = 16  ' 時点2id
Public Const A_Cid = 17  ' 試験条件id
Public Const A_ALL = 18  ' フィールド総数

Public Const J_id = 0      ' id
Public Const J_Rid = 1    ' 参照id
Public Const J_Aid = 2    ' 解析id
Public Const J_CASE = 3  ' 解析ケース
Public Const J_TIME = 4  ' 評価時点
Public Const J_TEMP = 5  ' TEMP (表面温度)
Public Const J_LTEMP = 6 ' LTEMP (等価線形温度)
Public Const J_MTEMP = 7 ' MTEMP (平均温度)
Public Const J_σr = 8    ' σr (表面応力)
Public Const J_σz = 9    ' σz (表面応力)
Public Const J_σθ = 10   ' σθ (表面応力)
Public Const J_τrz = 11  ' τrz (表面応力)
Public Const J_Lσr = 12  ' Lσr (等価線形応力)
Public Const J_Lσz = 13  ' Lσz (等価線形応力)
Public Const J_Lσθ = 14  ' Lσθ (等価線形応力)
Public Const J_Lτrz = 15 ' Lτrz (等価線形応力)
Public Const J_Mσr = 16  ' Mσr (膜応力)
Public Const J_Mσz = 17  ' Mσz (膜応力)
Public Const J_Mσθ = 18  ' Mσθ (膜応力)
Public Const J_Mτrz = 19 ' Mτrz (膜応力)
Public Const J_VMS = 20  ' VMS (ミス)
Public Const J_TRES = 21 ' TRES (トレス)
Public Const J_εr = 22  ' εr (弾性歪み)
Public Const J_εz = 23  ' εz (弾性歪み)
Public Const J_εθ = 24  ' εθ (弾性歪み)
Public Const J_γrz = 25  ' γrz (弾性歪み)
Public Const J_εpr = 26  ' εpr (塑性歪み)

```



```
Public Const AJ_ALL = A_ALL + J_ALL
Public Const AJJ_ALL = A_ALL + J_ALL * 2
```

```
Public Sub InputFieldCheck(nType As Long, dtFld As Recordset, IsNotExist As Integer)
```

```
Dim dtHYOKA As rdoResultset
Dim i As Long, j As Long, nSize As Long, ifld As Long
Dim tmp$, fld$, FN1$(100), FN2$(100), STFS$(100)
Dim SQL$
Dim hHYOKA As Integer
```

```
On Error GoTo Err1
```

```
For i = 0 To 99
```

```
FN1$(i) = ""
FN2$(i) = ""
STFS$(i) = ""
```

```
Next i
```

```
If nType = 12 Then 'き裂(Excelシート)
```

```
FN1$(0) = "日付"
FN1$(1) = "クォ番号"
FN1$(2) = "起点r1"
FN1$(3) = "起点z1"
FN1$(4) = "起点θ1"
FN1$(5) = "終点r2"
FN1$(6) = "終点z2"
FN1$(7) = "き裂長さ"
FN1$(8) = "母・溶・HAZ"
FN1$(9) = "溶接法"
FN1$(10) = "余盛有無"
FN1$(11) = "余盛形状"
FN1$(12) = "溶接備考"
FN1$(13) = "粒内・粒界"
FN1$(14) = "き裂形状"
FN1$(15) = "き裂発生部位"
FN1$(16) = "ストライション"
FN1$(17) = "ヒーマーク"
FN1$(18) = "統計処理クォイア"
FN1$(19) = "結晶粒度"
FN1$(20) = "備考"
FN1$(21) = "終点θ2"
nSize = 21
```

```
'FN2$(2) = "r1"
'FN2$(3) = "z1"
'FN2$(4) = "θ1"
'FN2$(5) = "r2"
'FN2$(6) = "z2"
```

```
ElseIf nType = 21 Or _
nType = 22 Or _
nType = 23 Then '温度(21:XLS/22:MDB/23:CSV)
FN1$(0) = "TIME1"
FN1$(1) = "TEMP1"
FN1$(2) = "CONV1"
nSize = 3
```

```
ElseIf nType = 31 Or _
nType = 32 Or _
nType = 33 Then '形状(31:XLS/32:MDB/33:CSV)
FN1$(0) = "MODEL-R"
FN1$(1) = "MODEL-Z"
nSize = 2
```

```
ElseIf nType = 41 Or _
nType = 42 Or _
nType = 43 Then '解析(41:XLS/42:MDB/43:CSV)
FN1$(0) = "起点r1"
FN1$(1) = "起点z1"
FN1$(2) = "起点θ1"
FN1$(3) = "終点r2"
FN1$(4) = "終点z2"
FN1$(5) = "終点θ2"
FN1$(6) = "TEMPMAX"
FN1$(7) = "TIME1"
FN1$(8) = "TEMP1"
FN1$(9) = "LTEMP1"
FN1$(10) = "MTEMP1"
FN1$(11) = "σr1"
FN1$(12) = "σz1"
FN1$(13) = "σθ1"
FN1$(14) = "τr21"
FN1$(15) = "Lσr1"
FN1$(16) = "Lσz1"
FN1$(17) = "Lσθ1"
FN1$(18) = "Lτr21"
FN1$(19) = "Mσr1"
FN1$(20) = "Mσz1"
FN1$(21) = "Mσθ1"
FN1$(22) = "Mτr21"
FN1$(23) = "εr1"
FN1$(24) = "εz1"
```

```

FN1$(25) = "εθ1"
FN1$(26) = "γrz1"
FN1$(27) = "εpr1"
FN1$(28) = "εpz1"
FN1$(29) = "εpθ1"
FN1$(30) = "γprz1"
FN1$(31) = "εcr1"
FN1$(32) = "εcz1"
FN1$(33) = "εcθ1"
FN1$(34) = "γcrz1"
' FN1$(0) = "日付"
' FN1$(1) = "ライン番号"
nSize = 35

```

```

FN2$(0) = "r1"
FN2$(1) = "z1"
FN2$(2) = "θ1"
FN2$(3) = "r2"
FN2$(4) = "z2"
FN2$(5) = "θ2"
FN2$(6) = "最高金属温度"
FN2$(7) = "評価時点1"
FN2$(8) = "表面温度1"
FN2$(9) = "等価線形温度1"
FN2$(10) = "平均温度1"
FN2$(11) = "表面応力σr1"
FN2$(12) = "表面応力σz1"
FN2$(13) = "表面応力σθ1"
FN2$(14) = "表面応力τrz1"
FN2$(15) = "等価線形応力σr1"
FN2$(16) = "等価線形応力σz1"
FN2$(17) = "等価線形応力σθ1"
FN2$(18) = "等価線形応力τrz1"
FN2$(19) = "膜応力σr1"
FN2$(20) = "膜応力σz1"
FN2$(21) = "膜応力σθ1"
FN2$(22) = "膜応力τrz1"
FN2$(23) = "弾性歪εr1"
FN2$(24) = "弾性歪εz1"
FN2$(25) = "弾性歪εθ1"
FN2$(26) = "弾性歪γrz1"
FN2$(27) = "塑性歪εr1"
FN2$(28) = "塑性歪εz1"
FN2$(29) = "塑性歪εθ1"
FN2$(30) = "塑性歪γrz1"
FN2$(31) = "クリープ歪εr1"
FN2$(32) = "クリープ歪εz1"
FN2$(33) = "クリープ歪εθ1"
FN2$(34) = "クリープ歪γrz1"

```

```

STF$(0) = "R1"
STF$(1) = "Z1"
STF$(2) = "H1"
STF$(3) = "R2"
STF$(4) = "Z2"
STF$(5) = "H2"
STF$(6) = "TEMPMAX"
STF$(7) = "TIME1"
STF$(8) = "TEMP1"
STF$(9) = "LTEMP1"
STF$(10) = "MTEMP1"
STF$(11) = "STRS11"
STF$(12) = "STRS21"
STF$(13) = "STRS31"
STF$(14) = "STRS41"
STF$(15) = "LSTRS11"
STF$(16) = "LSTRS21"
STF$(17) = "LSTRS31"
STF$(18) = "LSTRS41"
STF$(19) = "MSTRS11"
STF$(20) = "MSTRS21"
STF$(21) = "MSTRS31"
STF$(22) = "MSTRS41"
STF$(23) = "EE11"
STF$(24) = "EE21"
STF$(25) = "EE31"
STF$(26) = "EE41"
STF$(27) = "EP11"
STF$(28) = "EP21"
STF$(29) = "EP31"
STF$(30) = "EP41"
STF$(31) = "EC11"
STF$(32) = "EC21"
STF$(33) = "EC31"
STF$(34) = "EC41"

```

```
ElseIf nType = 90 Then ' 損傷評価法(90)
```

```

' 評価法7-F#から損傷値のF#-M#名を取得
SQL$ = "SELECT * FROM 評価法7-F#;"
' Set dtHYOKA = dbSTAR5.OpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly)
Set dtHYOKA = xOpenResultset(SQL$, rdOpenStatic, rdConcurReadOnly, hHYOKA)
On Error GoTo Err1

```

```

For i = 4 To dtHYOKA.rdoColumns.Count - 1
  FN1$(i - 4) = dtHYOKA.rdoColumns(i).Name
Next i
nSize = dtHYOKA.rdoColumns.Count - 4

```



```

'dtHYOKA.Close
xCloseResultset dtHYOKA, hHYOKA

'FN1$(0) = "SP"
'FN1$(1) = "Sn"
'FN1$(2) = "Sj"
'FN1$(3) = "K"
'FN1$(4) = "Ke d"
'FN1$(5) = "Ke"
'FN1$(6) = "Ke d g"
'FN1$(7) = "Ke d l"
'FN1$(8) = "Keep"
'FN1$(9) = "Ef"
'FN1$(10) = "Et"
'FN1$(11) = "Qn"
'FN1$(12) = "Qw"
'FN1$(13) = "Ql"
'FN1$(14) = "Qnu"
'FN1$(15) = "Qe f f"
'FN1$(16) = "Qf"
'FN1$(17) = "Qbm"
'FN1$(18) = "Neuber"
'FN1$(19) = "Df"
'FN1$(20) = "Dc"
'FN1$(21) = "D"
'FN1$(22) = "Dcr"
'nSize = 23

End If

tmp$ = ""
IsNotExist = 0

If nType <> 90 Then      '評価法以外るとき
  For i = 0 To nSize - 1
    For j = 0 To dtFld.Fields.Count - 1
      'MsgBox "(" & dtFld.Fields(j).Name & ")"
      If FN1$(i) = dtFld.Fields(j).Name Then Exit For
      If FN2$(i) = dtFld.Fields(j).Name Then Exit For
      If STF$(i) = dtFld.Fields(j).Name Then Exit For
    Next j
    If j >= dtFld.Fields.Count Then tmp$ = tmp$ & " " & FN1$(i)
  Next i

  If tmp$ <> "" Then
    IsNotExist = 1
    MsgBox "指定された入力フィールドには" & tmp$ & " フィールドがありません。処理を中止します。"
  End If

Else      '評価法るとき
  For i = 0 To dtFld.Fields.Count - 1
    ifld = dtFld.Fields(i).Value
    fld$ = dtFld.Fields(i).Name
    If ifld >= nSize Then      'フィールド番号が過大
      tmp$ = tmp$ & " " & fld$
    ElseIf ifld >= 0 Then      'マイナスのときはスワフ*
      For j = 0 To nSize - 1
        If fld$ = FN1$(j) Then Exit For
      Next j
      If j >= nSize Then tmp$ = tmp$ & " " & fld$
    End If
  Next i

  If tmp$ <> "" Then
    IsNotExist = 1
    MsgBox "指定された入力フィールドには未知のフィールド(" & tmp$ & ")がありません。処理を中止します。"
  End If

End If

Exit Sub

Err1:
MsgBox "Error in modSTARC/InputFieldCheck"
Call ShowErrorMessage
On Error Resume Next
dtHYOKA.Close
IsNotExist = 2

End Sub

Public Sub ShowErrorMessage()
Dim Msg As String

If Err.Number <> 0 Then
  'modSTAR.ErrNo = Err.Number
  Msg = "エラー番号" & Str(Err.Number) & "; " & Err.Source & _
    " でエラーが発生しました。" & Chr(13) & Err.Description
  MsgBox Msg, , "エラー", Err.HelpFile, Err.HelpContext
End If

```

End Sub

```
Public Function xOpenResultset(xName$, xType, xLockType, xHandle) As rdoResultset
  On Error GoTo Err1
  xHandle = 0
  dbSTAR5.QueryTimeout = 180
  Set xOpenResultset = dbSTAR5.OpenResultset(xName$, xType, xLockType)
  xHandle = 1
  Exit Function
```

```
Err1:
  MsgBox "Error in modSTARC/xOpenResultset"
  Call ShowErrorMessage
  xHandle = -1
```

End Function

```
Public Sub xCloseResultset(xObject As rdoResultset, xHandle As Integer)
  On Error GoTo Err1
  If xHandle = 1 Then
    xObject.Close
  End If
  xHandle = 0
  Exit Sub
```

```
Err1:
  MsgBox "Error in modSTARC/xCloseResultset"
  Call ShowErrorMessage
  xHandle = -1
```

End Sub

```
Public Function FindFieldPosition(dtTMP As rdoResultset, FName$) As Long
  Dim i As Long
  FindFieldPosition = -1
  For i = 0 To dtTMP.rdoColumns.Count - 1
    If dtTMP.rdoColumns(i).Name = FName$ Then
      FindFieldPosition = i
      Exit For
    End If
  Next i
  End Function
```

```
Public Function KFP(FName$) As Long
  Dim i As Long
  KFP = 50 ' 配列の最後
  For i = 0 To cKanriT - 1
    If FName$ = KanriFNS(i) Then
      KFP = i
      Exit For
    End If
  Next i
  End Function
```

```

Attribute VB_Name = "modDXFREAD"
' Private Lev As Integer      '現在のレベル
Private VTCount As Integer  'POLYLINEのVERTEXのカウンタ
Private IF70 As Integer     'POLYLINEの閉合フラグ
Private MaxID2 As Long      'DBに書込むID
Private X0 As Double        'X方向のオフセット量
Private Y0 As Double        'Y方向のオフセット量
Private X1 As Double        '起点のX座標
Private Y1 As Double        '起点のY座標
Private X2 As Double        '終点のX座標
Private Y2 As Double        '終点のY座標
Private X9 As Double        'POLYLINEの開始点のX座標
Private Y9 As Double        'POLYLINEの開始点のY座標
Private S1Type$             '第1レベルのタイプ
Private S2Type$             '第2レベルのタイプ
Private S3Type$             '第3レベルのタイプ
Private S4Type$             '第4レベルのタイプ
Private S2Name$             '第2レベルの名前
Private S3Name$             '第3レベルの名前

Private Sub ShapeDxfBlocks(No As Integer, SData$, IER As Integer)
    IER = 0
    On Error GoTo ERROREXIT
    If No = 0 Then
        If S3Type$ <> "" Then
            '未処理のコメントの実行
            If S3Type$ = "LINE" Then
                Print #2, X1; ", "; Y1; ", "; X2; ", "; Y2
                X1 = 0#
                Y1 = 0#
                X2 = 0#
                Y2 = 0#
                S3Type$ = ""
                S4Type$ = ""
            ElseIf S3Type$ = "POLYLINE" Then
                'POLYLINEは"SEQEND"で処理
            ElseIf S3Type$ = "INSERT" Then
                X0 = X1
                Y0 = Y1
                Debug.Print "Open " & S3Name$ & " For Input"
                'Open "E:\TMP\STAR#" & S3Name$ For Input As #3
                Open modSTAR.WorkDir$ & S3Name$ For Input As #3
                'ファイルを読み込み、別のファイルに格納
                Debug.Print S3Name$ & "ファイルを読み込み、" & S2Name$ & "ファイルに格納"
                Do While Not EOF(3)
                    Input #3, X1, Y1, X2, Y2
                    Print #2, X1 + X0; ", "; Y1 + Y0; ", "; X2 + X0; ", "; Y2 + Y0
                Loop
                Debug.Print "Close " & S3Name$
                Close #3
                X0 = 0#
                Y0 = 0#
                X1 = 0#
                Y1 = 0#
                X2 = 0#
                Y2 = 0#
                S3Type$ = ""
                S4Type$ = ""
                S3Name$ = ""
            End If
        End If
    End If

    If SData$ = "ENDSEC" Then      'セクションの終り
        ' Lev = 0
        S1Type$ = ""
        S2Type$ = ""
        S3Type$ = ""
        S4Type$ = ""
    ElseIf SData$ = "ENDBLK" Then
        Close #2
        ' Lev = 1
        S2Type$ = ""
        S3Type$ = ""
        S4Type$ = ""
    ElseIf SData$ = "BLOCK" Then
        ' Lev = 2
        S2Type$ = "BLOCK"
        S3Type$ = ""
    End If

```

```

S4Type$ = ""
ElseIf SData$ = "LINE" Then
    ' Lev = 3
    S3Type$ = "LINE"
    S4Type$ = ""

ElseIf SData$ = "INSERT" Then
    ' Lev = 3
    S3Type$ = "INSERT"
    S4Type$ = ""

ElseIf SData$ = "POLYLINE" Then
    ' Lev = 3
    S3Type$ = "POLYLINE"
    S4Type$ = ""
    VTCOUNT = 0
    IF70 = 0

ElseIf SData$ = "SEQEND" Then
    'ファイルのデータを保存
    If S3Type$ = "POLYLINE" And S4Type$ = "VERTEX" Then
        Print #2, X0; ", "; Y0; ", "; X1; ", "; Y1
        If (IF70 And 1) = 1 Then
            Print #2, X1; ", "; Y1; ", "; X9; ", "; Y9
        End If
    End If
    ' Lev = 2
    S3Type$ = ""
    S4Type$ = ""
    VTCOUNT = 0

ElseIf SData$ = "VERTEX" Then
    VTCOUNT = VTCOUNT + 1

    If VTCOUNT = 1 Then      '1個目
        '何もしない
    ElseIf VTCOUNT = 2 Then '2個目
        X0 = X1
        Y0 = Y1
        X9 = X1
        Y9 = Y1
    ElseIf VTCOUNT >= 3 Then '3個目以降
        'ファイルにデータを保存
        Print #2, X0; ", "; Y0; ", "; X1; ", "; Y1
        X0 = X1
        Y0 = Y1
    End If

    ' Lev = 4
    S4Type$ = "VERTEX"

Else
    MsgBox "対応できないコメントカードです。"
    GoTo ERROREXIT

End If

ElseIf No = 2 Then
    ' If Lev = 2 Then
    If S3Type$ <> "INSERT" Then
        S2Name$ = SData$ 'ex. "SHAPE"
        Debug.Print "Open " & S2Name$ & " For Output"
        ' Open "E:\TMP\STAR*" & S2Name$ For Output As #2
        Open modSTAR.WorkDir$ & S2Name$ For Output As #2

    ' ElseIf Lev = 3 Then
    Else
        S3Name$ = SData$

    End If

ElseIf No = 5 Then
    ' 無視

ElseIf No = 6 Then
    ' 無視

ElseIf No = 8 Then
    ' 無視

ElseIf No = 10 Then
    X1 = Val(SData$)

ElseIf No = 11 Then

```

```

X2 = Val(SData$)
ElseIf No = 20 Then
  Y1 = Val(SData$)
ElseIf No = 21 Then
  Y2 = Val(SData$)
ElseIf No = 30 Then
  '無視
ElseIf No = 62 Then
  '無視
ElseIf No = 66 Then
  '無視
ElseIf No = 70 Then
  IF70 = Val(SData$)
Else
  MsgBox "対応できないカト番号です。" & NoX$
  GoTo ERROREXIT
End If

GoTo NORMALEXIT

-----
ERROREXIT:
MsgBox "Error in modDXFREAD/ShapeDxfBlocks"
Call ShowErrorMessage
Close #2
Close #3
IER = 1

NORMALEXIT:
On Error GoTo 0
End Sub

Private Sub ShapeDxfEntities(No As Integer, SData$, dtSHAPE As rdoResultset, IER As Integer)
  IER = 0
On Error GoTo ERROREXIT
  If No = 0 Then
    'コメントカトの処理
    '未処理のコメントがあれば、実行
    If S2Type$ <> "" Then
      If S2Type$ = "LINE" Then
        'LINEコメントの処理
        Debug.Print MaxID2; ", "; X1; ", "; Y1; ", "; X2; ", "; Y2
        dtSHAPE.AddNew
        dtSHAPE![id] = MaxID2
        dtSHAPE![参照id] = modSTAR.nmID
        dtSHAPE![起点r1] = X1
        dtSHAPE![起点z1] = Y1
        dtSHAPE![終点r2] = X2
        dtSHAPE![終点z2] = Y2
        dtSHAPE![起点r1] = Y1
        dtSHAPE![起点z1] = X1
        dtSHAPE![終点r2] = Y2
        dtSHAPE![終点z2] = X2
        dtSHAPE.Update
        MaxID2 = MaxID2 + 1
        X1 = 0#
        Y1 = 0#
        X2 = 0#
        Y2 = 0#
        S2Type$ = ""
        S3Type$ = ""
      ElseIf S2Type$ = "POLYLINE" Then
        'POLYLINEコメントの処理
        'POLYLINEは"SEQEND"で処理
      ElseIf S2Type$ = "INSERT" Then
        'INSERTコメントの処理
        X0 = X1
        Y0 = Y1
        Debug.Print "Open " & S2Name$ & " For Input"

```

```

'Open "E:\TMP\STAR#" & S2Name$ For Input As #2
Open modSTAR.WorkDir$ & S2Name$ For Input As #2
'ファイルを読み込み、DBに登録
Debug.Print S2Name$ & "ファイルを読み込み、DBに登録"
Do While Not EOF(2)
  Input #2, X1, Y1, X2, Y2
  Debug.Print MaxID2; ", "; X1 + X0; ", "; Y1 + Y0; ", "; X2 + X0; ", "; Y2 + Y0
  dtSHAPE.AddNew
  dtSHAPE![id] = MaxID2
  dtSHAPE![参照id] = modSTAR.nmID
  dtSHAPE![起点r1] = X1 + X0
  dtSHAPE![起点z1] = Y1 + Y0
  dtSHAPE![終点r2] = X2 + X0
  dtSHAPE![終点z2] = Y2 + Y0
  dtSHAPE![起点r1] = Y1 + Y0
  dtSHAPE![起点z1] = X1 + X0
  dtSHAPE![終点r2] = Y2 + Y0
  dtSHAPE![終点z2] = X2 + X0
  dtSHAPE.Update
  MaxID2 = MaxID2 + 1
Loop
Debug.Print "Close " & S2Name$
Close #2
X0 = 0#
Y0 = 0#
X1 = 0#
Y1 = 0#
X2 = 0#
Y2 = 0#
S2Type$ = ""
S3Type$ = ""
S2Name$ = ""

End If

End If

'コメントカードを読んだときの処理
If SData$ = "ENDSEC" Then      'セクションの終り

  'Lev = 0
  S1Type$ = ""
  S2Type$ = ""
  S3Type$ = ""

ElseIf SData$ = "LINE" Then   'LINEカード

  'Lev = 2
  S2Type$ = "LINE"
  S3Type$ = ""

ElseIf SData$ = "INSERT" Then 'INSERTカード

  'Lev = 2
  S2Type$ = "INSERT"
  S3Type$ = ""

ElseIf SData$ = "POLYLINE" Then 'POLYLINEカード

  'Lev = 2
  S2Type$ = "POLYLINE"      'POLYLINEには名前カードがないため
  S3Type$ = ""
  VTCOUNT = 0
  IF70 = 0

ElseIf SData$ = "SEQEND" Then 'SEQENDカード

  'DBにデータを登録
  If S2Type$ = "POLYLINE" And S3Type$ = "VERTEX" Then
    Debug.Print MaxID2; ", "; X0; ", "; Y0; ", "; X1; ", "; Y1
    dtSHAPE.AddNew
    dtSHAPE![id] = MaxID2
    dtSHAPE![参照id] = modSTAR.nmID
    dtSHAPE![起点r1] = X0
    dtSHAPE![起点z1] = Y0
    dtSHAPE![終点r2] = X1
    dtSHAPE![終点z2] = Y1
    dtSHAPE![起点r1] = Y0
    dtSHAPE![起点z1] = X0
    dtSHAPE![終点r2] = Y1
    dtSHAPE![終点z2] = X1
    dtSHAPE.Update
    MaxID2 = MaxID2 + 1
  If (IF70 And 1) = 1 Then    '閉じたポリラインの処理
    Debug.Print MaxID2; ", "; X1; ", "; Y1; ", "; X9; ", "; Y9 'L-フ'状のとき
    dtSHAPE.AddNew
    dtSHAPE![id] = MaxID2
    dtSHAPE![参照id] = modSTAR.nmID
    dtSHAPE![起点r1] = X1
    dtSHAPE![起点z1] = Y1
    dtSHAPE![終点r2] = X9
    dtSHAPE![終点z2] = Y9
    dtSHAPE![起点r1] = Y1
    dtSHAPE![起点z1] = X1
    dtSHAPE![終点r2] = Y9
    dtSHAPE![終点z2] = X9
    dtSHAPE.Update

```

```

        MaxID2 = MaxID2 + 1
    End If
End If
Lev = 1
S2Type$ = ""
S3Type$ = ""
VTCOUNT = 0

ElseIf SData$ = "VERTEX" Then      ' VERTEXカ-ト*
    VTCOUNT = VTCOUNT + 1

    If VTCOUNT = 1 Then            ' 1個目
        ' 何もしない
    ElseIf VTCOUNT = 2 Then        ' 2個目
        X0 = X1
        Y0 = Y1
        X9 = X1
        Y9 = Y1
    ElseIf VTCOUNT >= 3 Then      ' 3個目以降
        ' DBにデータを登録
        Debug.Print MaxID2; ", "; X0; ", "; Y0; ", "; X1; ", "; Y1
        dtSHAPE.AddNew
        dtSHAPE![id] = MaxID2
        dtSHAPE![参照id] = modSTAR.nmID
        dtSHAPE![起点r1] = X0
        dtSHAPE![起点z1] = Y0
        dtSHAPE![終点r2] = X1
        dtSHAPE![終点z2] = Y1
        dtSHAPE![起点r1] = Y0
        dtSHAPE![起点z1] = X0
        dtSHAPE![終点r2] = Y1
        dtSHAPE![終点z2] = X1
        dtSHAPE.Update
        MaxID2 = MaxID2 + 1
        X0 = X1
        Y0 = Y1
    End If

    S3Type$ = "VERTEX"

Else                                ' 処理できないコマ-ト*カ-ト*のとき
    MsgBox "対応できないコマ-ト*カ-ト*です。"
    GoTo ERROREXIT

End If

ElseIf No = 2 Then                  ' 名前カ-ト*の処理
    S2Name$ = SData$                ' ex. "SHAPE"

ElseIf No = 5 Then
    ' 無視

ElseIf No = 6 Then
    ' 無視

ElseIf No = 8 Then
    ' 無視

ElseIf No = 10 Then
    X1 = Val(SData$)

ElseIf No = 11 Then
    X2 = Val(SData$)

ElseIf No = 20 Then
    Y1 = Val(SData$)

ElseIf No = 21 Then
    Y2 = Val(SData$)

ElseIf No = 30 Then
    ' 無視

ElseIf No = 62 Then
    ' 無視

ElseIf No = 66 Then
    ' 無視

ElseIf No = 70 Then
    IF70 = Val(SData$)

```

```

Else                                '処理できないカート番号のとき
    MsgBox "対応できないカート番号です。" & NoXS
    GoTo ERROREXIT
End If

GoTo NORMALEXIT

'-----
ERROREXIT:
MsgBox "Error in modDXFREAD/ShapeDxfEntities"
Call ShowErrorMessage
Close #2
IER = 1

NORMALEXIT:
On Error GoTo 0
End Sub

Private Sub ShapeDxfHeader(No As Integer, SData$, IER As Integer)
    '何もしないですべてのカートを無視
End Sub

Private Sub ShapeDxfTables(No As Integer, SData$, IER As Integer)
    '何もしないですべてのカートを無視
End Sub

Public Sub ShapeDxfRead(FName$, MaxID As Long, dtSHAPE As rdoResultset, RCount As Long, IER As Integer)
    Dim No As Integer
    Dim CC As Long
    Dim SData$
    Dim NoXS$
    On Error GoTo ERROREXIT

    ' Open "E:\TMP\STAR\SHAPE_1.DXF" For Input As #1
    Open FName$ For Input As #1

    IER = 0
    MaxID2 = MaxID
    CC = 0
    No = 0
    SData$ = ""
    S1Type$ = ""
    S2Type$ = ""
    S3Type$ = ""
    Lev = 0

ABC:
    Line Input #1, NoXS
    Line Input #1, SData$
    No = Val(NoXS)
    CC = CC + 2
    ' Debug.Print No; "("; SData$; ")"

    If S1Type$ <> "" Then                'セクション内のデータの処理

        If S1Type$ = "HEADER" Then Call ShapeDxfHeader(No, SData$, IER)
        If S1Type$ = "TABLES" Then Call ShapeDxfTables(No, SData$, IER)
        If S1Type$ = "BLOCKS" Then Call ShapeDxfBlocks(No, SData$, IER)
        If S1Type$ = "ENTITIES" Then Call ShapeDxfEntities(No, SData$, dtSHAPE, IER)

        If IER = 1 Then GoTo ERROREXIT

        If SData$ = "ENDSEC" Then
            S1Type$ = ""
            S2Type$ = ""
            S3Type$ = ""
            Lev = 0
        End If

    Else                                'セクション外のデータの処理

        If No = 0 Then

            If SData$ = "SECTION" Then
                S1Type$ = ""

            ElseIf SData$ = "EOF" Then
                Debug.Print "reach EOF (" & CC & "行)"
                GoTo NORMALEXIT
            End If
        End If
    End If
End Sub

```



```

Else
  MsgBox CC & "行目でエラー: SECTION/EOF"
  GoTo ERROREXIT
End If

ElseIf No = 2 Then          'セクションのタイプの設定
  If SData$ = "HEADER" Then
    S1Type$ = "HEADER"
    S2Type$ = ""
    S3Type$ = ""
    'Lev = 1

  ElseIf SData$ = "TABLES" Then
    S1Type$ = "TABLES"
    S2Type$ = ""
    S3Type$ = ""
    'Lev = 1

  ElseIf SData$ = "BLOCKS" Then
    S1Type$ = "BLOCKS"
    S2Type$ = ""
    S3Type$ = ""
    'Lev = 1

  ElseIf SData$ = "ENTITIES" Then
    S1Type$ = "ENTITIES"
    S2Type$ = ""
    S3Type$ = ""
    'Lev = 1

  Else          'セクション名がHEADER, TABLES, BLOCKS, ENTITIES以外はエラー
    MsgBox CC & "行目でエラー: SECTION名"
    GoTo ERROREXIT
  End If

Else          'Noが0と2以外の場合はエラー
  MsgBox "無効なカット番号です。"
  GoTo ERROREXIT
End If

End If

GoTo ABC          '次のカットの読み込みへ

-----
ERROREXIT:
  MsgBox "Error in modDXFREAD/ShapeDxfRead"
  IER = 1

NORMALEXIT:
  RCount = MaxID2 - MaxID
  On Error GoTo 0
  Close #1

End Sub

```

```

VERSION 5.00
Object = "{00028C01-0000-0000-0000-000000000046}#1.0#0"; "DBGRID32.CCX"
Begin VB.Form frmResDBG
    Caption = "検索結果(DBGridコントロール)"
    ClientHeight = 3990
    ClientLeft = 990
    ClientTop = 975
    ClientWidth = 5775
    Icon = "X_RESDBG.frx":0000
    LinkTopic = "Form1"
    MDIChild = -1 'True
    PaletteMode = 1 'Z-Order'
    ScaleHeight = 3990
    ScaleWidth = 5775
    Begin VB.Data dtaRes
        Caption = "dtaRes"
        Connect = "Access"
        DatabaseName = ""
        DefaultCursorType = 0 '既定値のカリ'
        DefaultType = 2 'ODBC'
        Exclusive = 0 'False'
        BeginProperty Font
            Name = "MS Pゴシック"
            Size = 11.25
            Charset = 128
            Weight = 400
            Underline = 0 'False'
            Italic = 0 'False'
            Strikethrough = 0 'False'
        EndProperty
        Height = 375
        Left = 480
        Options = 0
        ReadOnly = -1 'True'
        RecordsetType = 2 'スナップショット'
        RecordSource = ""
        Top = 3360
        Visible = 0 'False'
        Width = 3495
    End
    Begin MSDBGrid.DBGrid dbgBun
        Bindings = "X_RESDBG.frx":0442
        Height = 2775
        Left = 0
        OleObjectBlob = "X_RESDBG.frx":0453
        TabIndex = 0
        Top = 0
        Width = 5295
    End
    Begin VB.Menu mnuResFile
        Caption = "ファイル(&F)"
        NegotiatePosition = 1 '左'
        Begin VB.Menu mnuResClose
            Caption = "閉じる(&C)"
        End
    End
End
Attribute VB_Name = "frmResDBG"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Option Explicit

Private Sub Form_Load()
    Caption = modSTAR.tmpFormName$
    'デ-タコントロールとDBグリッドコントロールを使用して検索結果を表示
    dtaRes.DatabaseName = modSTAR.QueryDB$
    dtaRes.RecordSource = modSTAR.CurQuerySQL$
End Sub

Private Sub Form_Resize()
    dbgBun.Width = ScaleWidth
    dbgBun.Height = ScaleHeight
End Sub

Private Sub mnuResClose_Click()
    'Unload Me
    Unload frmStarMain.ActiveForm
End Sub

```

付 録

B. 表計算プログラムソースリスト

T C A L C S V . E X E

```

VERSION 1.0 CLASS
BEGIN
  MultiUse = -1 'True
END
Attribute VB_Name = "TCALCSV"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = True
Attribute VB_PredeclaredId = False
Attribute VB_Exposed = True

Private tAnalRows As Long
Private tAnalCols As Long
Private tJitenRows As Long
Private tJitenCols As Long
Private tTableRows As Long
Private tTableCols As Long

Public Sub Exec(AD() As Variant, J1D() As Variant, J2D() As Variant, TD() As Variant)
  ' Excelを使わないでVBで表計算を行う

  Dim i As Integer
  Dim dσr, dσz, dσθ, dτrz As Double
  Dim dLσr, dLσz, dLσθ, dLτrz As Double
  Dim dMσr, dMσz, dMσθ, dMτrz As Double
  Dim dσ1, dσ2, dσ3 As Double
  Dim dLσ1, dLσ2, dLσ3 As Double
  Dim dMσ1, dMσ2, dMσ3 As Double
  Dim Δσ1, Δσ2, Δσ3 As Double
  Dim ΔLσ1, ΔLσ2, ΔLσ3 As Double
  Dim ΔMσ1, ΔMσ2, ΔMσ3 As Double
  Dim dεr, dεz, dεθ, dγrz As Double
  Dim dεpr, dεpz, dεpθ, dγprz As Double
  Dim dεcr, dεcz, dεcθ, dγcrz As Double
  Dim wrk As Double

  For i = 0 To tAnalRows - 1
    TD(i, T_id) = AD(i, A_id)
    TD(i, T_Rid) = AD(i, A_Rid)
    TD(i, T_Aid) = AD(i, A_Aid)
    TD(i, T_DATE) = AD(i, A_DATE)
    TD(i, T_CASE) = AD(i, A_CASE)
    TD(i, T_LINENO) = AD(i, A_LINENO)
    TD(i, T_L) = AD(i, A_L)
    TD(i, T_r1) = AD(i, A_r1)
    TD(i, T_z1) = AD(i, A_z1)
    TD(i, T_θ1) = AD(i, A_θ1)

    TD(i, T_dσr) = J1D(i, J_σr) - J2D(i, J_σr)
    TD(i, T_dσz) = J1D(i, J_σz) - J2D(i, J_σz)
    TD(i, T_dσθ) = J1D(i, J_σθ) - J2D(i, J_σθ)
    TD(i, T_dτrz) = J1D(i, J_τrz) - J2D(i, J_τrz)

    TD(i, T_dLσr) = J1D(i, J_Lσr) - J2D(i, J_Lσr)
    TD(i, T_dLσz) = J1D(i, J_Lσz) - J2D(i, J_Lσz)
    TD(i, T_dLσθ) = J1D(i, J_Lσθ) - J2D(i, J_Lσθ)
    TD(i, T_dLτrz) = J1D(i, J_Lτrz) - J2D(i, J_Lτrz)

    TD(i, T_dMσr) = J1D(i, J_Mσr) - J2D(i, J_Mσr)
    TD(i, T_dMσz) = J1D(i, J_Mσz) - J2D(i, J_Mσz)
    TD(i, T_dMσθ) = J1D(i, J_Mσθ) - J2D(i, J_Mσθ)
    TD(i, T_dMτrz) = J1D(i, J_Mτrz) - J2D(i, J_Mτrz)

    dσr = TD(i, T_dσr)
    dσz = TD(i, T_dσz)
    dσθ = TD(i, T_dσθ)
    dτrz = TD(i, T_dτrz)
    wrk = Sqr(((dσr - dσz) / 2) ^ 2 + dτrz ^ 2)
    TD(i, T_dσ1) = (dσr + dσz) / 2 + wrk
    TD(i, T_dσ2) = (dσr + dσz) / 2 - wrk
    TD(i, T_dσ3) = dσθ

    dLσr = TD(i, T_dLσr)
    dLσz = TD(i, T_dLσz)
    dLσθ = TD(i, T_dLσθ)
    dLτrz = TD(i, T_dLτrz)
    wrk = Sqr(((dLσr - dLσz) / 2) ^ 2 + dLτrz ^ 2)
    TD(i, T_dLσ1) = (dLσr + dLσz) / 2 + wrk
    TD(i, T_dLσ2) = (dLσr + dLσz) / 2 - wrk
    TD(i, T_dLσ3) = dLσθ

    dMσr = TD(i, T_dMσr)
    dMσz = TD(i, T_dMσz)
    dMσθ = TD(i, T_dMσθ)
    dMτrz = TD(i, T_dMτrz)
    wrk = Sqr(((dMσr - dMσz) / 2) ^ 2 + dMτrz ^ 2)
    TD(i, T_dMσ1) = (dMσr + dMσz) / 2 + wrk
    TD(i, T_dMσ2) = (dMσr + dMσz) / 2 - wrk
    TD(i, T_dMσ3) = dMσθ

    TD(i, T_Δσr) = Abs(TD(i, T_dσr))
    TD(i, T_Δσz) = Abs(TD(i, T_dσz))
    TD(i, T_Δσθ) = Abs(TD(i, T_dσθ))
    TD(i, T_Δτrz) = Abs(TD(i, T_dτrz))
  
```

```

TD(i, T_ΔLσr) = Abs(TD(i, T_dLσr))
TD(i, T_ΔLσz) = Abs(TD(i, T_dLσz))
TD(i, T_ΔLσθ) = Abs(TD(i, T_dLσθ))
TD(i, T_ΔLτrz) = Abs(TD(i, T_dLτrz))

TD(i, T_ΔMσr) = Abs(TD(i, T_dMσr))
TD(i, T_ΔMσz) = Abs(TD(i, T_dMσz))
TD(i, T_ΔMσθ) = Abs(TD(i, T_dMσθ))
TD(i, T_ΔMτrz) = Abs(TD(i, T_dMτrz))

TD(i, T_Δσ1) = Abs(TD(i, T_dσ1))
TD(i, T_Δσ2) = Abs(TD(i, T_dσ2))
TD(i, T_Δσ3) = Abs(TD(i, T_dσ3))

TD(i, T_Δlσ1) = Abs(TD(i, T_dLσ1))
TD(i, T_Δlσ2) = Abs(TD(i, T_dLσ2))
TD(i, T_Δlσ3) = Abs(TD(i, T_dLσ3))

TD(i, T_ΔMσ1) = Abs(TD(i, T_dMσ1))
TD(i, T_ΔMσ2) = Abs(TD(i, T_dMσ2))
TD(i, T_ΔMσ3) = Abs(TD(i, T_dMσ3))

dσ1 = TD(i, T_dσ1)
dσ2 = TD(i, T_dσ2)
dσ3 = TD(i, T_dσ3)
Δσ1 = TD(i, T_Δσ1)
Δσ2 = TD(i, T_Δσ2)
Δσ3 = TD(i, T_Δσ3)
TD(i, T_ΔVMS) = Sqr((1 / 2) * ((dσr - dσz) ^ 2 +
                                (dσz - dσθ) ^ 2 +
                                (dσθ - dσr) ^ 2 +
                                6 * dτrz ^ 2))
TD(i, T_ΔTRES) = Max3(Abs(dσ1 - dσ2),
                       Abs(dσ2 - dσ3),
                       Abs(dσ3 - dσ1))
TD(i, T_ΔRANK) = Max3(Δσ1, Δσ2, Δσ3)

dLσ1 = TD(i, T_dLσ1)
dLσ2 = TD(i, T_dLσ2)
dLσ3 = TD(i, T_dLσ3)
ΔLσ1 = TD(i, T_ΔLσ1)
ΔLσ2 = TD(i, T_ΔLσ2)
ΔLσ3 = TD(i, T_ΔLσ3)
TD(i, T_ΔLVMS) = Sqr((1 / 2) * ((dLσr - dLσz) ^ 2 +
                                (dLσz - dLσθ) ^ 2 +
                                (dLσθ - dLσr) ^ 2 +
                                6 * dLτrz ^ 2))
TD(i, T_ΔLTRES) = Max3(Abs(dLσ1 - dLσ2),
                       Abs(dLσ2 - dLσ3),
                       Abs(dLσ3 - dLσ1))
TD(i, T_ΔLRANK) = Max3(ΔLσ1, ΔLσ2, ΔLσ3)

dMσ1 = TD(i, T_dMσ1)
dMσ2 = TD(i, T_dMσ2)
dMσ3 = TD(i, T_dMσ3)
ΔMσ1 = TD(i, T_ΔMσ1)
ΔMσ2 = TD(i, T_ΔMσ2)
ΔMσ3 = TD(i, T_ΔMσ3)
TD(i, T_ΔMVMS) = Sqr((1 / 2) * ((dMσr - dMσz) ^ 2 +
                                (dMσz - dMσθ) ^ 2 +
                                (dMσθ - dMσr) ^ 2 +
                                6 * dMτrz ^ 2))
TD(i, T_ΔMTRES) = Max3(Abs(dMσ1 - dMσ2),
                       Abs(dMσ2 - dMσ3),
                       Abs(dMσ3 - dMσ1))
TD(i, T_ΔMRANK) = Max3(ΔMσ1, ΔMσ2, ΔMσ3)

TD(i, T_der) = J1D(i, J_εr) - J2D(i, J_εr)
TD(i, T_dez) = J1D(i, J_εz) - J2D(i, J_εz)
TD(i, T_dεθ) = J1D(i, J_εθ) - J2D(i, J_εθ)
TD(i, T_dγrz) = J1D(i, J_γrz) - J2D(i, J_γrz)

TD(i, T_dεpr) = J1D(i, J_εpr) - J2D(i, J_εpr)
TD(i, T_dεpz) = J1D(i, J_εpz) - J2D(i, J_εpz)
TD(i, T_dεpθ) = J1D(i, J_εpθ) - J2D(i, J_εpθ)
TD(i, T_dγprz) = J1D(i, J_γprz) - J2D(i, J_γprz)

TD(i, T_dεcr) = J1D(i, J_εcr) - J2D(i, J_εcr)
TD(i, T_dεcz) = J1D(i, J_εcz) - J2D(i, J_εcz)
TD(i, T_dεcθ) = J1D(i, J_εcθ) - J2D(i, J_εcθ)
TD(i, T_dγcrz) = J1D(i, J_γcrz) - J2D(i, J_γcrz)

der = TD(i, T_der)
dez = TD(i, T_dez)
dεθ = TD(i, T_dεθ)
dγrz = TD(i, T_dγrz)
dεpr = TD(i, T_dεpr)
dεpz = TD(i, T_dεpz)
dεpθ = TD(i, T_dεpθ)
dγprz = TD(i, T_dγprz)
dεcr = TD(i, T_dεcr)
dεcz = TD(i, T_dεcz)
dεcθ = TD(i, T_dεcθ)
dγcrz = TD(i, T_dγcrz)
TD(i, T_ΔEPCQ) = (1 / 3) * Sqr(2 * (((dεr + dεpr + dεcr) - (dez + dεpz + dεcz)) ^ 2 +
                                     ((dεz + dεpz + dεcz) - (dεθ + dεpθ + dεcθ)) ^ 2 +
                                     ((dεθ + dεpθ + dεcθ) - (dεr + dεpr + dεcr)) ^ 2 +
                                     ))

```

$$TD(i, T_ΔEPEQ) = (1 / 3) * \text{Sqr}(2 * ((\frac{3}{2}) * (d\gamma rz + d\gamma prz + d\gamma crz) ^ 2))$$

$$TD(i, T_ΔECEQ) = (1 / 3) * \text{Sqr}(2 * ((\frac{3}{2}) * d\gamma prz ^ 2 + \frac{(d\epsilon pr - d\epsilon pz)^2 + (d\epsilon p\theta - d\epsilon pr)^2 + (d\epsilon cr - d\epsilon cz)^2 + (d\epsilon c\theta - d\epsilon cr)^2}{(3/2) * d\gamma crz ^ 2}))$$

Next i

End Sub

'----- 表計算終了 -----'

Public Property Get AnalRows() As Long
AnalRows = tAnalRows
End Property

Public Property Let AnalRows(vNewValue As Long)
tAnalRows = vNewValue
End Property

Public Property Get AnalCols() As Long
AnalCols = tAnalCols
End Property

Public Property Let AnalCols(vNewValue As Long)
tAnalCols = vNewValue
End Property

Public Property Get JitenRows() As Long
JitenRows = tJitenRows
End Property

Public Property Let JitenRows(vNewValue As Long)
tJitenRows = vNewValue
End Property

Public Property Get JitenCols() As Long
JitenCols = tJitenCols
End Property

Public Property Let JitenCols(vNewValue As Long)
tJitenCols = vNewValue
End Property

Public Property Get TableRows() As Long
TableRows = tTableRows
End Property

Public Property Let TableRows(vNewValue As Long)
tTableCols = vNewValue
End Property

Public Property Get TableCols() As Long
TableCols = tTableCols
End Property

Public Property Let TableCols(vNewValue As Long)
tTableCols = vNewValue
End Property

Public Function Max3(Var1, Var2, Var3)
Max3 = Var1
If Var2 > Max3 Then Max3 = Var2
If Var3 > Max3 Then Max3 = Var3
End Function

Public Function Min3(Var1, Var2, Var3)
Min3 = Var1
If Var2 < Min3 Then Min3 = Var2
If Var3 < Min3 Then Min3 = Var3
End Function

Attribute VB_Name = "modSTARC"
Option Explicit

```

Public Const C_id = 0      ' id
Public Const C_Rid = 1    ' 参照id
Public Const C_DATANO = 2 ' データ番号
Public Const C_DATE = 3  ' 日付
Public Const C_LINENO = 4 ' ライン番号
Public Const C_r1 = 5     ' 起点r1
Public Const C_z1 = 6     ' 起点z1
Public Const C_θ1 = 7     ' 起点θ1
Public Const C_r2 = 8     ' 終点r2
Public Const C_z2 = 9     ' 終点z2
Public Const C_θ2 = 10    ' 終点θ2
Public Const C_LEN = 11   ' き裂長さ
Public Const C_BYH = 12   ' 母-溶-HAZ
Public Const C_WELD = 13  ' 溶接法
Public Const C_YOMORI = 14 ' 余盛有無
Public Const C_YKEIJO = 15 ' 余盛形状
Public Const C_WBIKO = 16 ' 溶接備考
Public Const C_RYU = 17   ' 粒内・粒界
Public Const C_CKEIJO = 18 ' き裂形状
Public Const C_CYCLE = 19 ' き裂発生サイクル
Public Const C_STRIATE = 20 ' ストライション
Public Const C_BMARK = 21 ' ヒーマーク
Public Const C_ACTIVE = 22 ' 統計処理アクティブ
Public Const C_CRYSTAL = 23 ' 結晶粒度
Public Const C_BIKO = 24  ' 備考
Public Const C_L = 25     ' L
Public Const C_ALL = 26   ' フィールド総数

Public Const S_id = 0      ' id
Public Const S_Rid = 1    ' 参照id
Public Const S_Sochi = 2  ' 試験装置
Public Const S_Tai = 3    ' 試験体
Public Const S_MATid = 4  ' 材料id
Public Const S_Bui = 5    ' 評価部位
Public Const S_TCOLD = 6  ' COLD温度
Public Const S_THOT = 7   ' HOT温度
Public Const S_HCOLD = 8  ' COLD保持時間
Public Const S_HHOT = 9   ' HOT保持時間
Public Const S_LOAD = 10  ' 機械荷重
Public Const S_CYCLE = 11 ' サイクル数
Public Const S_NODATA = 12 ' データ数
Public Const S_RefName = 13 ' 関連資料名
Public Const S_RefNo = 14 ' 関連資料番号
Public Const S_RefDS = 15 ' 関連データセット
Public Const S_SMAT = 16  ' SMAT TAG NO
Public Const S_MEMO = 17  ' メモ
Public Const S_ALL = 18   ' フィールド総数

Public Const A_id = 0      ' id
Public Const A_Rid = 1    ' 参照id
Public Const A_Aid = 2    ' 解析id
Public Const A_DATE = 3  ' 日付
Public Const A_CASE = 4  ' 解析ケース
Public Const A_LINENO = 5 ' ライン番号
Public Const A_L = 6     ' L
Public Const A_Q = 7     ' q
Public Const A_r1 = 8     ' 起点r1
Public Const A_z1 = 9     ' 起点z1
Public Const A_θ1 = 10    ' 起点θ1
Public Const A_r2 = 11    ' 終点r2
Public Const A_z2 = 12    ' 終点z2
Public Const A_θ2 = 13    ' 終点θ2
Public Const A_TEMPMAX = 14 ' TEMPMAX
Public Const A_J1id = 15  ' 時点1id
Public Const A_J2id = 16  ' 時点2id
Public Const A_Cid = 17   ' 試験条件id
Public Const A_ALL = 18   ' フィールド総数

Public Const J_id = 0      ' id
Public Const J_Rid = 1    ' 参照id
Public Const J_Aid = 2    ' 解析id
Public Const J_CASE = 3  ' 解析ケース
Public Const J_TIME = 4  ' 評価時点
Public Const J_TEMP = 5  ' TEMP (表面温度)
Public Const J_LTEMP = 6 ' LTEMP (等価線形温度)
Public Const J_MTEMP = 7 ' MTEMP (平均温度)
Public Const J_σr = 8    ' σr (表面応力)
Public Const J_σz = 9    ' σz (表面応力)
Public Const J_σθ = 10   ' σθ (表面応力)
Public Const J_τrz = 11  ' τrz (表面応力)
Public Const J_Lσr = 12  ' Lσr (等価線形応力)
Public Const J_Lσz = 13  ' Lσz (等価線形応力)
Public Const J_Lσθ = 14  ' Lσθ (等価線形応力)
Public Const J_Lτrz = 15 ' Lτrz (等価線形応力)
Public Const J_Mσr = 16  ' Mσr (膜応力)
Public Const J_Mσz = 17  ' Mσz (膜応力)
Public Const J_Mσθ = 18  ' Mσθ (膜応力)
Public Const J_Mτrz = 19 ' Mτrz (膜応力)
Public Const J_VMS = 20  ' VMS (ミーゼス)
Public Const J_TRES = 21 ' TRES (トレス)
Public Const J_εr = 22  ' εr (弾性歪み)
Public Const J_εz = 23  ' εz (弾性歪み)
Public Const J_εθ = 24  ' εθ (弾性歪み)
Public Const J_γrz = 25 ' γrz (弾性歪み)
Public Const J_εpr = 26  ' εpr (塑性歪み)

```


Public Const AJ_ALL = A_ALL + J_ALL
Public Const AJJ_ALL = A_ALL + J_ALL * 2

Attribute VB_Name = "modTC"
Option Explicit

Public InpDir\$
Public InpFName\$
Public InpTName\$
Public FormCloseType As Integer

Public Anal(300, 100) As Variant
Public Jiten1(300, 100) As Variant
Public Jiten2(300, 100) As Variant
Public Table(300, 100) As Variant

Public TField\$(100)

Public Const A_id = 0	'id
Public Const A_Rid = 1	'参照id
Public Const A_DATE = 2	'日付
Public Const A_CASE = 3	'解析ケース
Public Const A_LINENO = 4	'ライン番号
Public Const A_L = 5	'L
Public Const A_Q = 6	'q
Public Const A_r1 = 7	'起点r1
Public Const A_z1 = 8	'起点z1
Public Const A_θ1 = 9	'起点θ1
Public Const A_r2 = 10	'終点r2
Public Const A_z2 = 11	'終点z2
Public Const A_θ2 = 12	'終点θ2
Public Const A_TEMPMAX = 13	'TEMPMAX
Public Const A_J1id = 14	'時点1id
Public Const A_J2id = 15	'時点2id
Public Const A_Cid = 16	'試験条件id
Public Const A_ALL = 17	'フィールド総数

Public Const J_id = 0	'id
Public Const J_TIME = 1	'評価時点
Public Const J_TEMP = 2	'TEMP (表面温度)
Public Const J_LTEMP = 3	'LTEMP (等価線形温度)
Public Const J_MTEMP = 4	'MTEMP (平均温度)
Public Const J_σr = 5	'σr (表面応力)
Public Const J_σz = 6	'σz (表面応力)
Public Const J_σθ = 7	'σθ (表面応力)
Public Const J_τrz = 8	'τrz (表面応力)
Public Const J_Lσr = 9	'Lσr (等価線形応力)
Public Const J_Lσz = 10	'Lσz (等価線形応力)
Public Const J_Lσθ = 11	'Lσθ (等価線形応力)
Public Const J_Lτrz = 12	'Lτrz (等価線形応力)
Public Const J_Mσr = 13	'Mσr (膜応力)
Public Const J_Mσz = 14	'Mσz (膜応力)
Public Const J_Mσθ = 15	'Mσθ (膜応力)
Public Const J_Mτrz = 16	'Mτrz (膜応力)
Public Const J_VMS = 17	'VMS (ミ-ε'ス)
Public Const J_TRES = 18	'TRES (トレス)
Public Const J_εr = 19	'εr (弾性歪み)
Public Const J_εz = 20	'εz (弾性歪み)
Public Const J_εθ = 21	'εθ (弾性歪み)
Public Const J_γrz = 22	'γrz (弾性歪み)
Public Const J_εpr = 23	'εpr (塑性歪み)
Public Const J_εpz = 24	'εpz (塑性歪み)
Public Const J_εpθ = 25	'εpθ (塑性歪み)
Public Const J_γprz = 26	'γprz (塑性歪み)
Public Const J_εcr = 27	'εcr (ｸﾘ-ﾌﾞ歪み)
Public Const J_εcz = 28	'εcz (ｸﾘ-ﾌﾞ歪み)
Public Const J_εcθ = 29	'εcθ (ｸﾘ-ﾌﾞ歪み)
Public Const J_γcrz = 30	'γcrz (ｸﾘ-ﾌﾞ歪み)
Public Const J_ALL = 31	'フィールド総数

Public Const T_id = 0	'id
Public Const T_Rid = 1	'参照id
Public Const T_DATE = 2	'日付
Public Const T_CASE = 3	'解析ケース
Public Const T_LINENO = 4	'ライン番号
Public Const T_L = 5	'L
Public Const T_r1 = 6	'起点r1
Public Const T_z1 = 7	'起点z1
Public Const T_θ1 = 8	'起点θ1
Public Const T_dσr = 9	'
Public Const T_dσz = 10	'
Public Const T_dσθ = 11	'
Public Const T_dτrz = 12	'
Public Const T_dLσr = 13	'
Public Const T_dLσz = 14	'
Public Const T_dLσθ = 15	'
Public Const T_dLτrz = 16	'
Public Const T_dMσr = 17	'
Public Const T_dMσz = 18	'
Public Const T_dMσθ = 19	'
Public Const T_dMτrz = 20	'
Public Const T_dσ1 = 21	'
Public Const T_dσ2 = 22	'
Public Const T_dσ3 = 23	'
Public Const T_dLσ1 = 24	'
Public Const T_dLσ2 = 25	'
Public Const T_dLσ3 = 26	'
Public Const T_dMσ1 = 27	'
Public Const T_dMσ2 = 28	'
Public Const T_dMσ3 = 29	'

```

Public Const T_Δσr = 30
Public Const T_Δσz = 31
Public Const T_Δσθ = 32
Public Const T_Δτrz = 33
Public Const T_ΔLσr = 34
Public Const T_ΔLσz = 35
Public Const T_ΔLσθ = 36
Public Const T_ΔLτrz = 37
Public Const T_ΔMσr = 38
Public Const T_ΔMσz = 39
Public Const T_ΔMσθ = 40
Public Const T_ΔMτrz = 41
Public Const T_Δσ1 = 42
Public Const T_Δσ2 = 43
Public Const T_Δσ3 = 44
Public Const T_ΔLσ1 = 45
Public Const T_ΔLσ2 = 46
Public Const T_ΔLσ3 = 47
Public Const T_ΔMσ1 = 48
Public Const T_ΔMσ2 = 49
Public Const T_ΔMσ3 = 50
Public Const T_ΔVMS = 51
Public Const T_ΔTRES = 52
Public Const T_ΔRANK = 53
Public Const T_ΔLVMS = 54
Public Const T_ΔLTRES = 55
Public Const T_ΔLRANK = 56
Public Const T_ΔMVMS = 57
Public Const T_ΔMTRES = 58
Public Const T_ΔMRANK = 59
Public Const T_dεr = 60
Public Const T_dεz = 61
Public Const T_dεθ = 62
Public Const T_dγrz = 63
Public Const T_dεpr = 64
Public Const T_dεpz = 65
Public Const T_dεpθ = 66
Public Const T_dγprz = 67
Public Const T_dεcr = 68
Public Const T_dεcz = 69
Public Const T_dεcθ = 70
Public Const T_dγcrz = 71
Public Const T_ΔEPECEQ = 72
Public Const T_ΔEPEQ = 73
Public Const T_ΔECEQ = 74
Public Const T_ALL = 75 'フォーム総数

```

```

Public Const AJ_ALL = A_ALL + J_ALL
Public Const AJJ_ALL = A_ALL + J_ALL * 2

```

```

Public Sub ExecTCalc()
    Dim dbTEMP As Database
    Dim dtTEMP As Recordset
    Dim frmRES As New frmResult
    Dim FName$, TName$
    Dim DatabaseName$, RecordSource$, Connect$
    Dim i As Integer
    Dim j As Integer
    Dim k As Integer
    Dim ARows As Integer
    Dim TCols As Integer
    Dim obj As Object
    Dim AFG As Object

    frmExcel.Show 1

    If modTC.FormCloseType = vbCancel Then Exit Sub

    FName$ = modTC.InpFName$
    TName$ = modTC.InpTName$

    If StrComp(Right$(FName$, 3), "XLS", 1) = 0 Then 'Excelシート
        'Excelシートを一時的に表示
        Connect$ = "Excel 5.0;"
        DatabaseName$ = FName$
        RecordSource$ = TName$

    ElseIf StrComp(Right$(FName$, 3), "MDB", 1) = 0 Then 'Access
        'Accessデータベースを一時的に表示
        Connect$ = "Access 2.0;"
        DatabaseName$ = FName$
        RecordSource$ = TName$

    Else
        MsgBox "認識できないファイル(拡張子)です。"
        Exit Sub
    End If
End Sub

```

```
Screen.MousePointer = 11
```

```
'Excelのトのオフ'>
```

```
Set dbTEMP = OpenDatabase(DatabaseName$, False, False, Connect$)
```

```
Set dtTEMP = dbTEMP.OpenRecordset(RecordSource$, dbOpenSnapshot)
```

```
If dtTEMP.RecordCount > 0 Then
```

```
k = 0
```

```
Do Until dtTEMP.EOF
```

```
' Anal(k, A_id) = GetDBData(dtTEMP![Id])
' Anal(k, A_Rid) = GetDBData(dtTEMP![参照id])
' Anal(k, A_DATE) = GetDBData(dtTEMP![日付])
' Anal(k, A_CASE) = GetDBData(dtTEMP![解析ケース])
' Anal(k, A_LINENO) = GetDBData(dtTEMP![ライン番号])
' Anal(k, A_L) = GetDBData(dtTEMP![L])
' Anal(k, A_Q) = GetDBData(dtTEMP![Q])
Anal(k, A_r1) = GetDBData(dtTEMP![起点r1])
Anal(k, A_z1) = GetDBData(dtTEMP![起点z1])
Anal(k, A_θ1) = GetDBData(dtTEMP![起点θ1])
Anal(k, A_r2) = GetDBData(dtTEMP![終点r2])
Anal(k, A_z2) = GetDBData(dtTEMP![終点z2])
Anal(k, A_θ2) = GetDBData(dtTEMP![終点θ2])
Anal(k, A_TEMPMAX) = GetDBData(dtTEMP![TEMPMAX])
' Anal(k, A_J1id) = GetDBData(dtTEMP![時点1id])
' Anal(k, A_J2id) = GetDBData(dtTEMP![時点2id])
' Anal(k, A_Cid) = GetDBData(dtTEMP![試験条件id])
```

```
' Jiten1(k, J_id) = GetDBData(dtTEMP![Id])
Jiten1(k, J_TIME) = GetDBData(dtTEMP![TIME1])
Jiten1(k, J_TEMP) = GetDBData(dtTEMP![TEMP1])
Jiten1(k, J_LTEMP) = GetDBData(dtTEMP![LTEMP1])
Jiten1(k, J_MTEMP) = GetDBData(dtTEMP![MTEMP1])
Jiten1(k, J_σr) = GetDBData(dtTEMP![σr1])
Jiten1(k, J_σz) = GetDBData(dtTEMP![σz1])
Jiten1(k, J_σθ) = GetDBData(dtTEMP![σθ1])
Jiten1(k, J_τrz) = GetDBData(dtTEMP![τrz1])
Jiten1(k, J_Lσr) = GetDBData(dtTEMP![Lσr1])
Jiten1(k, J_Lσz) = GetDBData(dtTEMP![Lσz1])
Jiten1(k, J_Lσθ) = GetDBData(dtTEMP![Lσθ1])
Jiten1(k, J_Lτrz) = GetDBData(dtTEMP![Lτrz1])
Jiten1(k, J_Mσr) = GetDBData(dtTEMP![Mσr1])
Jiten1(k, J_Mσz) = GetDBData(dtTEMP![Mσz1])
Jiten1(k, J_Mσθ) = GetDBData(dtTEMP![Mσθ1])
Jiten1(k, J_Mτrz) = GetDBData(dtTEMP![Mτrz1])
' Jiten1(k, J_VMS) = GetDBData(dtTEMP![VMS1])
' Jiten1(k, J_TRES) = GetDBData(dtTEMP![TRES1])
Jiten1(k, J_εr) = GetDBData(dtTEMP![εr1])
Jiten1(k, J_εz) = GetDBData(dtTEMP![εz1])
Jiten1(k, J_εθ) = GetDBData(dtTEMP![εθ1])
Jiten1(k, J_γrz) = GetDBData(dtTEMP![γrz1])
Jiten1(k, J_εpr) = GetDBData(dtTEMP![εpr1])
Jiten1(k, J_εpz) = GetDBData(dtTEMP![εpz1])
Jiten1(k, J_εpθ) = GetDBData(dtTEMP![εpθ1])
Jiten1(k, J_γprz) = GetDBData(dtTEMP![γprz1])
Jiten1(k, J_εcr) = GetDBData(dtTEMP![εcr1])
Jiten1(k, J_εcz) = GetDBData(dtTEMP![εcz1])
Jiten1(k, J_εcθ) = GetDBData(dtTEMP![εcθ1])
Jiten1(k, J_γcrz) = GetDBData(dtTEMP![γcrz1])
```

```
' Jiten2(k, J_id) = GetDBData(dtTEMP![Id])
Jiten2(k, J_TIME) = GetDBData(dtTEMP![TIME2])
Jiten2(k, J_TEMP) = GetDBData(dtTEMP![TEMP2])
Jiten2(k, J_LTEMP) = GetDBData(dtTEMP![LTEMP2])
Jiten2(k, J_MTEMP) = GetDBData(dtTEMP![MTEMP2])
Jiten2(k, J_σr) = GetDBData(dtTEMP![σr2])
Jiten2(k, J_σz) = GetDBData(dtTEMP![σz2])
Jiten2(k, J_σθ) = GetDBData(dtTEMP![σθ2])
Jiten2(k, J_τrz) = GetDBData(dtTEMP![τrz2])
Jiten2(k, J_Lσr) = GetDBData(dtTEMP![Lσr2])
Jiten2(k, J_Lσz) = GetDBData(dtTEMP![Lσz2])
Jiten2(k, J_Lσθ) = GetDBData(dtTEMP![Lσθ2])
Jiten2(k, J_Lτrz) = GetDBData(dtTEMP![Lτrz2])
Jiten2(k, J_Mσr) = GetDBData(dtTEMP![Mσr2])
Jiten2(k, J_Mσz) = GetDBData(dtTEMP![Mσz2])
Jiten2(k, J_Mσθ) = GetDBData(dtTEMP![Mσθ2])
Jiten2(k, J_Mτrz) = GetDBData(dtTEMP![Mτrz2])
' Jiten2(k, J_VMS) = GetDBData(dtTEMP![VMS2])
' Jiten2(k, J_TRES) = GetDBData(dtTEMP![TRES2])
Jiten2(k, J_εr) = GetDBData(dtTEMP![εr2])
Jiten2(k, J_εz) = GetDBData(dtTEMP![εz2])
Jiten2(k, J_εθ) = GetDBData(dtTEMP![εθ2])
Jiten2(k, J_γrz) = GetDBData(dtTEMP![γrz2])
Jiten2(k, J_εpr) = GetDBData(dtTEMP![εpr2])
Jiten2(k, J_εpz) = GetDBData(dtTEMP![εpz2])
Jiten2(k, J_εpθ) = GetDBData(dtTEMP![εpθ2])
Jiten2(k, J_γprz) = GetDBData(dtTEMP![γprz2])
Jiten2(k, J_εcr) = GetDBData(dtTEMP![εcr2])
Jiten2(k, J_εcz) = GetDBData(dtTEMP![εcz2])
Jiten2(k, J_εcθ) = GetDBData(dtTEMP![εcθ2])
Jiten2(k, J_γcrz) = GetDBData(dtTEMP![γcrz2])
```

```
k = k + 1
```

```
dtTEMP.MoveNext
```

```
Loop
```

```

ARows = k
TCols = T_ALL

Set obj = CreateObject("TCALCSV.TCALC")

obj.AnalRows = k
obj.Exec Anal, Jiten1, Jiten2, Table

Set obj = Nothing

End If

dtTEMP.Close 'Excelシートのカース
dbTEMP.Close 'Excelシートのカース

'結果表示ウインドウの作成
frmRES.Show

'ウインドウタイトルの設定
frmTC_MDI.ActiveForm.Caption = FName$

'結果をグリッドに表示
Set AFG = frmTC_MDI.ActiveForm.grdResult

AFG.Rows = ARows + 2
AFG.Cols = TCols + 1

'フィールド名の転記
For j = 0 To T_ALL - 1
    AFG.ColWidth(j + 1) = 1000          'デフォルトは630twip
    AFG.Col = j + 1
    AFG.Row = 0
    AFG.Text = TField$(j)
    AFG.Row = 1
    AFG.Text = TField$(j)
Next j

'データ部の転記
For i = 0 To ARows - 1
    AFG.Row = i + 2
    AFG.Col = 0
    AFG.Text = i + 1
    For j = 0 To T_ALL - 1
        AFG.Col = j + 1
        AFG.Text = Table(i, j)
    Next j
Next i

Screen.MousePointer = 0

End Sub

Public Function GetDBData(JVal)

    If IsNull(JVal) = True Then
        GetDBData = 0
    Else
        GetDBData = JVal
    End If

End Function

```

```

Attribute VB_Name = "frmExcel"
Attribute VB_Creatable = False
Attribute VB_Exposed = False
Option Explicit

Private Sub cmbFileType_Click()
    Dim tmp As String

    tmp = Mid(cmbFileType, InStr(cmbFileType, "(") + 1)
    filFileList.Pattern = Left(tmp, Len(tmp) - 1)
End Sub

Private Sub cmdCancel_Click()
    modTC.FormCloseType = vbCancel
    Unload Me
End Sub

Private Sub cmdOK_Click()
    Dim Ext As String

    ' 入力データのチェック
    If lblCurFile.Caption = "" Then
        MsgBox "ファイルが選択されていません。"
        Exit Sub
    End If
    Ext = UCase(Right(lblCurFile.Caption, 3))
    If (Ext = "MDB" Or Ext = "XLS") And _
        lstRange.ListIndex < 0 Then
        MsgBox "データ範囲が選択されていません。"
        Exit Sub
    End If

    ' ファイル名とテーブル名を設定
    If Right(dirDirList.Path, 1) <> "*" Then
        modTC.InpFileName$ = dirDirList.Path & "*" & lblCurFile.Caption
    Else
        modTC.InpFileName$ = dirDirList.Path & lblCurFile.Caption
    End If

    If Ext = "MDB" Or Ext = "XLS" Then
        modTC.InpTableName$ = lstRange.List(lstRange.ListIndex)
    Else
        modTC.InpTableName$ = ""
    End If

    modTC.InpDir$ = dirDirList.Path
    modTC.FormCloseType = vbOK
    Unload Me
End Sub

Private Sub dirDirList_Change()
    lblCurDir.Caption = dirDirList.Path
    filFileList.Path = dirDirList.Path
End Sub

Private Sub drvDrive_Change()
    Dim oldPath As String

    oldPath = dirDirList.Path
    On Error GoTo trapDrv:
    dirDirList.Path = drvDrive.Drive
    On Error GoTo 0
    Exit Sub

trapDrv:
    Beep
    MsgBox "ドライブの準備ができていません。", 48, "エラー"
    dirDirList.Path = oldPath
    drvDrive.Drive = Left(oldPath, 2)
End Sub

Private Sub filFileList_Click()
    lblCurFile.Caption = filFileList.filename
End Sub

Private Sub Form_Load()
    Dim FileType$
    Dim m As Long
    Dim m2 As Long

    lblCurDir.Caption = dirDirList.Path
    FileType$ = "Excel 5.0 (*.xls)|Access (*.mdb)|"
    m = 1
    Do While 1
        m2 = InStr(m, FileType$, "|", 1)

```

```

    If m2 = 0 Then Exit Do
    cmbFileType.AddItem Mid$(FileType$, m, m2 - m)
    m = m2 + 1
Loop
cmbFileType.ListIndex = 0

dirDirList.Path = modTC.InpDir$
drvDrive.Drive = Left(modTC.InpDir$, 2)

modTC.FormCloseType = vbCancel
End Sub

Private Sub lblCurFile_Change()
    Dim dbTMP As Database
    Dim i As Long
    Dim LCount As Long
    Dim tmpFilename As String

    Screen.MousePointer = 11

    'ファイル名の取得
    If Right(dirDirList.Path, 1) <> "\" Then
        tmpFilename = dirDirList.Path & "\" & lblCurFile.Caption
    Else
        tmpFilename = dirDirList.Path & lblCurFile.Caption
    End If

    'テーブル範囲リストボックスをクリア
    LCount = lstRange.ListCount
    For i = LCount - 1 To 0 Step -1
        lstRange.RemoveItem i
    Next i

    'テーブル範囲リストボックスの更新
    Select Case UCase(Right(tmpFilename, 3))
    Case "MDB"
        Set dbTMP = Workspaces(0).OpenDatabase(tmpFilename, False, True)
        For i = 0 To dbTMP.TableDefs.Count - 1
            lstRange.AddItem dbTMP.TableDefs(i).Name
        Next i
        dbTMP.Close
    Case "XLS"
        Set dbTMP = Workspaces(0).OpenDatabase(tmpFilename, False, True, "Excel 5.0;")
        For i = 0 To dbTMP.TableDefs.Count - 1
            lstRange.AddItem dbTMP.TableDefs(i).Name
        Next i
        dbTMP.Close
    End Select

    Screen.MousePointer = 0
End Sub

```

Attribute VB_Name = "frmTC_MDI"
 Attribute VB_Creatable = False
 Attribute VB_Exposed = False
 Option Explicit

Private Sub MDIForm_Load()

modTC.InpDir\$ = CurDir

TField\$(T_id) = "id"
 TField\$(T_Rid) = "参照id"
 TField\$(T_DATE) = "日付"
 TField\$(T_CASE) = "解析ケース"
 TField\$(T_LINENO) = "ライン番号"
 TField\$(T_L) = "L"
 TField\$(T_r1) = "起点r1"
 TField\$(T_z1) = "起点z1"
 TField\$(T_θ1) = "起点θ1"
 TField\$(T_dσr) = "dσr"
 TField\$(T_dσz) = "dσz"
 TField\$(T_dσθ) = "dσθ"
 TField\$(T_dτrz) = "dτrz"
 TField\$(T_dLσr) = "dLσr"
 TField\$(T_dLσz) = "dLσz"
 TField\$(T_dLσθ) = "dLσθ"
 TField\$(T_dLτrz) = "dLτrz"
 TField\$(T_dMσr) = "dMσr"
 TField\$(T_dMσz) = "dMσz"
 TField\$(T_dMσθ) = "dMσθ"
 TField\$(T_dMτrz) = "dMτrz"
 TField\$(T_dσ1) = "dσ1"
 TField\$(T_dσ2) = "dσ2"
 TField\$(T_dσ3) = "dσ3"
 TField\$(T_dLσ1) = "dLσ1"
 TField\$(T_dLσ2) = "dLσ2"
 TField\$(T_dLσ3) = "dLσ3"
 TField\$(T_dMσ1) = "dMσ1"
 TField\$(T_dMσ2) = "dMσ2"
 TField\$(T_dMσ3) = "dMσ3"
 TField\$(T_Δσr) = "Δσr"
 TField\$(T_Δσz) = "Δσz"
 TField\$(T_Δσθ) = "Δσθ"
 TField\$(T_Δτrz) = "Δτrz"
 TField\$(T_ΔLσr) = "ΔLσr"
 TField\$(T_ΔLσz) = "ΔLσz"
 TField\$(T_ΔLσθ) = "ΔLσθ"
 TField\$(T_ΔLτrz) = "ΔLτrz"
 TField\$(T_ΔMσr) = "ΔMσr"
 TField\$(T_ΔMσz) = "ΔMσz"
 TField\$(T_ΔMσθ) = "ΔMσθ"
 TField\$(T_ΔMτrz) = "ΔMτrz"
 TField\$(T_Δσ1) = "Δσ1"
 TField\$(T_Δσ2) = "Δσ2"
 TField\$(T_Δσ3) = "Δσ3"
 TField\$(T_ΔLσ1) = "ΔLσ1"
 TField\$(T_ΔLσ2) = "ΔLσ2"
 TField\$(T_ΔLσ3) = "ΔLσ3"
 TField\$(T_ΔMσ1) = "ΔMσ1"
 TField\$(T_ΔMσ2) = "ΔMσ2"
 TField\$(T_ΔMσ3) = "ΔMσ3"
 TField\$(T_ΔVMS) = "ΔVMS"
 TField\$(T_ΔTRES) = "ΔTRES"
 TField\$(T_ΔRANK) = "ΔRANK"
 TField\$(T_ΔLVMS) = "ΔLVMS"
 TField\$(T_ΔLTRES) = "ΔLTRES"
 TField\$(T_ΔLRANK) = "ΔLRANK"
 TField\$(T_ΔMVMS) = "ΔMVMS"
 TField\$(T_ΔMTRES) = "ΔMTRES"
 TField\$(T_ΔMRANK) = "ΔMRANK"
 TField\$(T_dεr) = "dεr"
 TField\$(T_dεz) = "dεz"
 TField\$(T_dεθ) = "dεθ"
 TField\$(T_dγrz) = "dγrz"
 TField\$(T_dεpr) = "dεpr"
 TField\$(T_dεpz) = "dεpz"
 TField\$(T_dεpθ) = "dεpθ"
 TField\$(T_dγprz) = "dγprz"
 TField\$(T_dεcr) = "dεcr"
 TField\$(T_dεcz) = "dεcz"
 TField\$(T_dεcθ) = "dεcθ"
 TField\$(T_dγcrz) = "dγcrz"
 TField\$(T_ΔEEPCEQ) = "ΔEEPCE"
 TField\$(T_ΔEPEQ) = "ΔEPEQ"
 TField\$(T_ΔECEQ) = "ΔECEQ"

End Sub

Private Sub mnuExcel_Click()
 Call ExecTCalc
 End Sub

Private Sub mnuExit_Click()
 End
 End Sub


```
VERSION 5.00
Begin VB.Form Form1
    Caption       = "Form1"
    ClientHeight  = 8445
    ClientLeft    = 1140
    ClientTop     = 1515
    ClientWidth   = 6780
    Icon          = "TCALCSV.frx":0000
    LinkTopic     = "Form1"
    PaletteMode   = 1 'Z オ-ク'-
    ScaleHeight   = 8445
    ScaleWidth    = 6780
    Begin VB.CommandButton Command1
        Caption    = "Command1"
        Height     = 615
        Left       = 960
        TabIndex   = 0
        Top        = 720
        Width      = 1695
    End
End
Attribute VB_Name = "Form1"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
```

```
Attribute VB_Name = "frmResult"  
Attribute VB_Creatable = False  
Attribute VB_Exposed = False  
Option Explicit
```

```
Private Sub Form_Resize()  
    grdResult.Width = ScaleWidth  
    grdResult.Height = ScaleHeight  
End Sub
```

```
Private Sub mnuEditAll_Click()  
    grdResult.SelStartRow = 1  
    grdResult.SelStartCol = 1  
    grdResult.SelEndRow = grdResult.Rows - 1  
    grdResult.SelEndCol = grdResult.Cols - 1  
End Sub
```

```
Private Sub mnuEditCopy_Click()  
    Clipboard.SetText grdResult.Clip, vbCFText  
End Sub
```

```
Private Sub mnuExcel_Click()  
    Call ExecTCalc  
End Sub
```

```
Private Sub mnuFileClose_Click()  
    Unload frmTC_MDI.ActiveForm  
End Sub
```

付 録

C. 補間処理プログラムソースリスト

INTPSV. EXE

```

VERSION 1.0 CLASS
BEGIN
  MultiUse = -1 'True
END
Attribute VB_Name = "INTP"
Attribute VB_Creatable = True
Attribute VB_Exposed = True

Private tCrackSRows As Long
Private tTargetRows As Long
Private tTargetCols As Long

Public Sub Exec(Crks() As Variant, Tgt() As Variant, TgtS() As Variant)
  Dim c, i, j, k As Integer
  Dim LN As Integer
  Dim LL As Double
  Dim Fac, Var1, Var2 As Double
  Dim VType$

  For c = 0 To tCrackSRows - 1
    LN = Crks(c, 0) - 1 'ライク番号
    LL = Crks(c, 1) 'L

    For i = 1 To tTargetRows - 1
      If LL <= Tgt(LN, i, 0) Then
        Fac = (LL - Tgt(LN, i - 1, 0)) / (Tgt(LN, i, 0) - Tgt(LN, i - 1, 0))
        For j = 1 To tTargetCols - 1
          VType$ = TypeName(Tgt(LN, i, j))
          If VType$ = "String" Or VType$ = "Date" Then
            TgtS(c, j) = Tgt(LN, i, j)
            Debug.Print TgtS(c, j)
          Else
            Var1 = Tgt(LN, i - 1, j)
            Var2 = Tgt(LN, i, j)
            TgtS(c, j) = Var1 + Fac * (Var2 - Var1)
            Debug.Print TgtS(c, j)
          End If
        Next j

        TgtS(c, 1) = Crks(c, 2) 'idを転記
        Exit For '次のき裂へ
      End If
    Next i
  Next c
End Sub

```

```

Public Function Max3(Var1, Var2, Var3)
  Max3 = Var1
  If Var2 > Max3 Then Max3 = Var2
  If Var3 > Max3 Then Max3 = Var3
End Function

Public Function Min3(Var1, Var2, Var3)
  Min3 = Var1
  If Var2 < Min3 Then Min3 = Var2
  If Var3 < Min3 Then Min3 = Var3
End Function

Public Property Get CrackSRows() As Integer
  CrackSRows = tCrackSRows
End Property

Public Property Let CrackSRows(vNewValue As Integer)
  tCrackSRows = vNewValue
End Property

Public Property Get TargetRows() As Integer
  TargetRows = tTargetRows
End Property

Public Property Let TargetRows(vNewValue As Integer)
  tTargetRows = vNewValue
End Property

Public Property Get TargetCols() As Integer
  TargetCols = tTargetCols
End Property

Public Property Let TargetCols(vNewValue As Integer)
  tTargetCols = vNewValue
End Property

```

```

Attribute VB_Name = "modIP"
Option Explicit

Public InpCDir$
Public InpADir$
Public InpCFName$
Public InpAFName$
Public InpCTName$
Public InpATName$

Public FormCloseType As Integer

Public Anal(300, 100) As Variant
Public Jiten1(300, 100) As Variant
Public Jiten2(300, 100) As Variant
Public Table(300, 100) As Variant

Public Crk$(600, 3) As Variant      '統計用のき裂データ
Public Tgt(3, 300, 100) As Variant '計算データ(解析,表計算,損傷値)
Public Tgt$(600, 100) As Variant   '統計用の計算データ

Public AField$(100)

Public Sub ExecTCalc()
    Dim dbCRKS As Database
    Dim dbTRG As Database
    Dim dtCRKS As Recordset
    Dim dtTRG As Recordset
    Dim frmRES As New frmResult
    Dim CFName$, CTName$
    Dim AFName$, ATName$
    Dim CDB$, CRS$, CCN$
    Dim ADB$, ARS$, ACN$
    Dim i As Integer
    Dim j As Integer
    Dim k As Integer
    Dim n As Integer
    Dim CSRows As Integer
    Dim TRows As Integer
    Dim ARows As Integer
    Dim TCols As Integer
    Dim obj As Object
    Dim AFG As Object

    frmExcel.Show 1

    If modIP.FormCloseType = vbCancel Then Exit Sub

    CFName$ = modIP.InpCFName$
    CTName$ = modIP.InpCTName$
    AFName$ = modIP.InpAFName$
    ATName$ = modIP.InpATName$

    If StrComp(Right$(CFName$, 3), "XLS", 1) = 0 Then      'Excelシート
        'Excelシートを一時的に表示
        CCN$ = "Excel 5.0;"
        CDB$ = CFName$
        CRS$ = CTName$

    ElseIf StrComp(Right$(CFName$, 3), "MDB", 1) = 0 Then 'Access
        'Accessデータベースを一時的に表示
        CCN$ = "Access 2.0;"
        CDB$ = CFName$
        CRS$ = CTName$

    Else
        MsgBox "認識できないき裂ファイル(拡張子)です。"
        Exit Sub
    End If

    If StrComp(Right$(AFName$, 3), "XLS", 1) = 0 Then      'Excelシート
        'Excelシートを一時的に表示
        ACN$ = "Excel 5.0;"
        ADB$ = AFName$
        ARS$ = ATName$

    ElseIf StrComp(Right$(AFName$, 3), "MDB", 1) = 0 Then 'Access
        'Accessデータベースを一時的に表示
        ACN$ = "Access 2.0;"
        ADB$ = AFName$
        ARS$ = ATName$

    Else
        MsgBox "認識できない計算ファイル(拡張子)です。"
        Exit Sub
    End If

```

End If

```

Screen.MousePointer = 11

' き裂_シートのオープン
Set dbCRKS = OpenDatabase(CDB$, False, False, CCN$)
Set dtCRKS = dbCRKS.OpenRecordset(CRS$, dbOpenSnapshot)

' 計算シートのオープン
Set dbTRG = OpenDatabase(ADB$, False, False, ACN$)
Set dtTRG = dbTRG.OpenRecordset(ARS$, dbOpenSnapshot)

If dtCRKS.RecordCount > 0 And dtTRG.RecordCount > 0 Then

' き裂データの読み込み
dtCRKS.MoveLast
CSRows = dtCRKS.RecordCount
Debug.Print CSRows
i = 0
dtCRKS.MoveFirst
Do Until dtCRKS.EOF
  CrkS(i, 0) = dtCRKS![ライン番号]
  CrkS(i, 1) = dtCRKS![L]
  CrkS(i, 2) = dtCRKS![id]
  Debug.Print i; CrkS(i, 0); CrkS(i, 1); CrkS(i, 2)
  i = i + 1
dtCRKS.MoveNext
Loop

' 計算データの読み込み
dtTRG.MoveLast
TRows = dtTRG.RecordCount
TCols = dtTRG.Fields.Count
Debug.Print "TRows,TCols:"; TRows; TCols
m = -1
i = 0
dtTRG.MoveFirst
Do Until dtTRG.EOF
  If (dtTRG![ライン番号] - 1) <> m Then
    m = m + 1
    i = 0
  End If
  Tgt(m, i, 0) = dtTRG![L]
  Debug.Print m; i; 0; Tgt(m, i, 0)
  For j = 0 To dtTRG.Fields.Count - 1
    If IsNull(dtTRG.Fields(j).Value) = False Then
      Tgt(m, i, j + 1) = dtTRG.Fields(j).Value
    End If
  Next j
  i = i + 1
dtTRG.MoveNext
Loop

' 計算フィールドの北°-
For j = 0 To dtTRG.Fields.Count - 1
  AField$(j) = dtTRG.Fields(j).Name
Next j

' 計算の実行
Set obj = CreateObject("INTPSV.INTP")
obj.CrackSRows = CSRows
obj.TargetRows = TRows
obj.TargetCols = TCols + 1
Debug.Print "Exec Start"
obj.Exec CrkS, Tgt, TgtS
Debug.Print "Exec End"
Set obj = Nothing

End If

dtTRG.Close ' 計算シートのクローズ
dbTRG.Close ' 計算シートのクローズ

dtCRKS.Close ' き裂_シートのクローズ
dbCRKS.Close ' き裂_シートのクローズ

' 結果表示ウィンドウの作成
frmRES.Show

' ウィンドウタイトルの設定
frmIP_MDI.ActiveForm.Caption = CFName$ & " " & AFName$

' 結果をグリッドに表示
Set AFG = frmIP_MDI.ActiveForm.grdResult

AFG.Rows = CSRows + 2
AFG.Cols = TCols + 1

' フィールド名の転記
For j = 0 To TCols - 1

```

```
AFG.ColWidth(j + 1) = 1000      'フィールド幅は630twip
AFG.Col = j + 1
AFG.Row = 0
AFG.Text = AField$(j)
AFG.Row = 1
AFG.Text = AField$(j)
Next j

'データ部の転記
For i = 0 To CSRows - 1
  AFG.Row = i + 2
  AFG.Col = 0
  AFG.Text = i + 1
  For j = 0 To TCols - 1
    AFG.Col = j + 1
    AFG.Text = Tgt$(i, j + 1)
  Next j
Next i

Screen.MousePointer = 0

End Sub

Public Function GetDBData(JVal)
  If IsNull(JVal) = True Then
    GetDBData = 0
  Else
    GetDBData = JVal
  End If
End Function

End Function
```



```

Attribute VB_Name = "frmExcel"
Attribute VB_Creatable = False
Attribute VB_Exposed = False
Option Explicit

Private Sub cmbAFileType_Click()
    Dim tmp As String

    tmp = Mid(cmbAFileType, InStr(cmbAFileType, "(") + 1)
    filAFileList.Pattern = Left(tmp, Len(tmp) - 1)
End Sub

Private Sub cmbCFileType_Click()
    Dim tmp As String

    tmp = Mid(cmbCFileType, InStr(cmbCFileType, "(") + 1)
    filCFileList.Pattern = Left(tmp, Len(tmp) - 1)
End Sub

Private Sub cmdCancel_Click()
    modIP.FormCloseType = vbCancel
    Unload Me
End Sub

Private Sub cmdOK_Click()
    Dim CExt As String
    Dim AExt As String

    '入力データのチェック
    If lblCurCFile.Caption = "" Then
        MsgBox "き裂ファイルが選択されていません。"
        Exit Sub
    End If
    If lblCurAFile.Caption = "" Then
        MsgBox "計算ファイルが選択されていません。"
        Exit Sub
    End If
    CExt = UCase(Right(lblCurCFile.Caption, 3))
    If (CExt = "MDB" Or CExt = "XLS") And _
        lstCRange.ListIndex < 0 Then
        MsgBox "き裂データの範囲が選択されていません。"
        Exit Sub
    End If
    AExt = UCase(Right(lblCurAFile.Caption, 3))
    If (AExt = "MDB" Or AExt = "XLS") And _
        lstARange.ListIndex < 0 Then
        MsgBox "計算データの範囲が選択されていません。"
        Exit Sub
    End If

    'ファイル名とテーブル名を設定
    If Right(dirCDirList.Path, 1) <> "/" Then
        modIP.InpCFName$ = dirCDirList.Path & "/" & lblCurCFile.Caption
    Else
        modIP.InpCFName$ = dirCDirList.Path & lblCurCFile.Caption
    End If

    If Right(dirADirList.Path, 1) <> "/" Then
        modIP.InpAFName$ = dirADirList.Path & "/" & lblCurAFile.Caption
    Else
        modIP.InpAFName$ = dirADirList.Path & lblCurAFile.Caption
    End If

    If CExt = "MDB" Or CExt = "XLS" Then
        modIP.InpCTName$ = lstCRange.List(lstCRange.ListIndex)
    Else
        modIP.InpCTName$ = ""
    End If

    If AExt = "MDB" Or AExt = "XLS" Then
        modIP.InpATName$ = lstARange.List(lstARange.ListIndex)
    Else
        modIP.InpATName$ = ""
    End If

    modIP.InpCDir$ = dirCDirList.Path
    modIP.InpADir$ = dirADirList.Path

    modIP.FormCloseType = vbOK
    Unload Me
End Sub

Private Sub dirADirList_Change()
    lblCurADir.Caption = dirADirList.Path
    filAFileList.Path = dirADirList.Path
End Sub

Private Sub dirCDirList_Change()
    lblCurCDir.Caption = dirCDirList.Path
    filCFileList.Path = dirCDirList.Path

```

End Sub

```
Private Sub drvADrive_Change()
    Dim oldPath As String

    oldPath = dirADirList.Path
    On Error GoTo trapDrv:
    dirADirList.Path = drvADrive.Drive
    On Error GoTo 0
    Exit Sub
```

```
trapDrv:
    Beep
    MsgBox "ドライブの準備ができていません。", 48, "エラー"
    dirADirList.Path = oldPath
    drvADrive.Drive = Left(oldPath, 2)
```

End Sub

```
Private Sub drvCDrive_Change()
    Dim oldPath As String

    oldPath = dirCDirList.Path
    On Error GoTo trapDrv:
    dirCDirList.Path = drvCDrive.Drive
    On Error GoTo 0
    Exit Sub
```

```
trapDrv:
    Beep
    MsgBox "ドライブの準備ができていません。", 48, "エラー"
    dirCDirList.Path = oldPath
    drvCDrive.Drive = Left(oldPath, 2)
```

End Sub

```
Private Sub filAFileList_Click()
    lblCurAFile.Caption = filAFileList.filename
    lblAFileName.Caption = lblCurADir & "¥" & filAFileList.filename
End Sub
```

```
Private Sub filCFileList_Click()
    lblCurCFile.Caption = filCFileList.filename
    lblCFileName.Caption = lblCurCDir & "¥" & filCFileList.filename
End Sub
```

```
Private Sub Form_Load()
    Dim CFileType$
    Dim AFileType$
    Dim m As Long
    Dim m2 As Long
```

```
    lblCurCDir.Caption = dirCDirList.Path
    lblCurADir.Caption = dirADirList.Path
```

```
    CFileType$ = "Excel 5.0 (*.xls)|Access (*.mdb)|"
    AFileType$ = "Excel 5.0 (*.xls)|Access (*.mdb)|"
```

```
    m = 1
    Do While 1
        m2 = InStr(m, CFileType$, "|", 1)
        If m2 = 0 Then Exit Do
        cmbCFileType.AddItem Mid$(CFileType$, m, m2 - m)
        m = m2 + 1
    Loop
```

```
    cmbCFileType.ListIndex = 0
```

```
    m = 1
    Do While 1
        m2 = InStr(m, AFileType$, "|", 1)
        If m2 = 0 Then Exit Do
        cmbAFileType.AddItem Mid$(AFileType$, m, m2 - m)
        m = m2 + 1
    Loop
```

```
    cmbAFileType.ListIndex = 0
```

```
    dirCDirList.Path = modIP.InpCDir$
    dirADirList.Path = modIP.InpADir$
    drvCDrive.Drive = Left(modIP.InpCDir$, 2)
    drvADrive.Drive = Left(modIP.InpADir$, 2)
```

```
    modIP.FormCloseType = vbCancel
```

End Sub

```
Private Sub lblCurAFile_Change()
    Dim dbTMP As Database
    Dim i As Long
    Dim lCount As Long
    Dim tmpFilename As String

    Screen.MousePointer = 11
```

```

'ファイル名の取得
If Right(dirADirList.Path, 1) <> "*" Then
    tmpFilename = dirADirList.Path & "*" & lblCurAFile.Caption
Else
    tmpFilename = dirADirList.Path & lblCurAFile.Caption
End If

'データ範囲リストボックスをクリア
LCount = lstARange.ListCount
For i = LCount - 1 To 0 Step -1
    lstARange.RemoveItem i
Next i

'データ範囲リストボックスの更新
Select Case UCase(Right(tmpFilename, 3))
Case "MDB"
    Set dbTMP = Workspaces(0).OpenDatabase(tmpFilename, False, True)
    For i = 0 To dbTMP.TableDefs.Count - 1
        lstARange.AddItem dbTMP.TableDefs(i).Name
    Next i
    dbTMP.Close
Case "XLS"
    Set dbTMP = Workspaces(0).OpenDatabase(tmpFilename, False, True, "Excel 5.0;")
    For i = 0 To dbTMP.TableDefs.Count - 1
        lstARange.AddItem dbTMP.TableDefs(i).Name
    Next i
    dbTMP.Close
End Select

Screen.MousePointer = 0
End Sub

Private Sub lblCurCFile_Change()
    Dim dbTMP As Database
    Dim i As Long
    Dim LCount As Long
    Dim tmpFilename As String

    Screen.MousePointer = 11

'ファイル名の取得
If Right(dirCDirList.Path, 1) <> "*" Then
    tmpFilename = dirCDirList.Path & "*" & lblCurCFile.Caption
Else
    tmpFilename = dirCDirList.Path & lblCurCFile.Caption
End If

'データ範囲リストボックスをクリア
LCount = lstCRange.ListCount
For i = LCount - 1 To 0 Step -1
    lstCRange.RemoveItem i
Next i

'データ範囲リストボックスの更新
Select Case UCase(Right(tmpFilename, 3))
Case "MDB"
    Set dbTMP = Workspaces(0).OpenDatabase(tmpFilename, False, True)
    For i = 0 To dbTMP.TableDefs.Count - 1
        lstCRange.AddItem dbTMP.TableDefs(i).Name
    Next i
    dbTMP.Close
Case "XLS"
    Set dbTMP = Workspaces(0).OpenDatabase(tmpFilename, False, True, "Excel 5.0;")
    For i = 0 To dbTMP.TableDefs.Count - 1
        lstCRange.AddItem dbTMP.TableDefs(i).Name
    Next i
    dbTMP.Close
End Select

Screen.MousePointer = 0
End Sub

```

```
Attribute VB_Name = "frmResult"  
Attribute VB_Creatable = False  
Attribute VB_Extensible = False  
Option Explicit
```

```
Private Sub Form_Resize()  
    grdResult.Width = ScaleWidth  
    grdResult.Height = ScaleHeight  
End Sub
```

```
Private Sub mnuEditAll_Click()  
    grdResult.SelStartRow = 1  
    grdResult.SelStartCol = 1  
    grdResult.SelEndRow = grdResult.Rows - 1  
    grdResult.SelEndCol = grdResult.Cols - 1  
End Sub
```

```
Private Sub mnuEditCopy_Click()  
    Clipboard.SetText grdResult.Clip, vbCFText  
End Sub
```

```
Private Sub mnuExcel_Click()  
    Call ExecTCalc  
End Sub
```

```
Private Sub mnuFileClose_Click()  
    Unload frmIP_MDI.ActiveForm  
End Sub
```

```
Attribute VB_Name = "frmIP_MDI"  
Attribute VB_Creatable = False  
Attribute VB_Exposed = False  
Option Explicit
```

```
Private Sub MDIForm_Load()
```

```
    modIP.InpCDir$ = CurDir  
    modIP.InpADir$ = CurDir
```

```
End Sub
```

```
Private Sub mnuExcel_Click()
```

```
    Call ExecTCalc  
End Sub
```

```
Private Sub mnuExit_Click()
```

```
    End  
End Sub
```