

JAN 13 1992

National Radiobiology Archives Distributed Access Programmer's Guide

**J. C. Prather
S. K. Smith
C. R. Watson**

December 1991

**Prepared for the U.S. Department of Energy
under Contract DE-AC06-76RLO 1830**

**Pacific Northwest Laboratory
Operated for the U.S. Department of Energy
by Battelle Memorial Institute**



PNL-7899

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor Battelle Memorial Institute, nor any of their employees, makes **any warranty, expressed or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights.** Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof, or Battelle Memorial Institute. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

PACIFIC NORTHWEST LABORATORY
operated by
BATTELLE MEMORIAL INSTITUTE
for the
UNITED STATES DEPARTMENT OF ENERGY
under Contract DE-AC06-76RLO 1830

Printed in the United States of America

Available to DOE and DOE contractors from the
Office of Scientific and Technical Information, P.O. Box 62, Oak Ridge, TN 37831;
prices available from (615) 576-8401. FTS 626-8401.

Available to the public from the National Technical Information Service,
U.S. Department of Commerce, 5285 Port Royal Rd., Springfield, VA 22161.

PNL--7899

DE92 005449

NATIONAL RADIOBIOLOGY ARCHIVES
DISTRIBUTED ACCESS PROGRAMMER'S GUIDE

J.C. Prather*
S.K. Smith
C.R. Watson

December 1991

Prepared for
the U.S. Department of Energy
under Contract DE-AC06-76RLO 1830

Pacific Northwest Laboratory
Richland, Washington 99352

*Linfield College
McMinnville, Oregon 97128

MASTER

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED *So*

PREFACE

Executive Summary

This document describes the database access software, NRADEMO, and the subset loading script NRADEMO\MAINT\MAINTAIN, which comprise the National Radiobiology Archives Distributed Access Package. The guide is intended for use by an experienced database management specialist. It contains information about the physical and logical organization of the software and data files. It also contains printouts of all the scripts and associated batch processing files. It is part of a suite of documents published by the National Radiobiology Archives.

<i>Document Title</i>	<i>Audience</i>
National Radiobiology Archives Database Documentation: A. Overview	General
National Radiobiology Archives Database Documentation: B. Design	General
National Radiobiology Archives Database Documentation: C. Table Structures	NRA staff, Advisory Committee, Intensive Users
Distributed Access User's Guide	General, Novice Users
Distributed Access Programmer's Guide	NRA staff
Distributed Access Data Loading Guide	NRA staff
MAKEMENU Application Generator User's Guide	NRA staff

ACKNOWLEDGEMENTS

People who made it happen

We would like to thank several people who contributed to the success of this project. Anna Martinez worked with the National Radiobiology Archives (NRA) staff in the summer of 1991. She made many valuable suggestions to the software development team, and is preparing to make major enhancements next summer. Becky Kennedy served as peer reviewer for the implementation and this document. Earleen Ligothe is the NRA project clerk. Earleen coordinates data entry for the Inventory and Bibliography Databases, and proof-read the many revisions of this document.

How we assembled the document

The document was written using WordPerfect® 5.1. Extensive use was made of the Master and Subdocument feature. Screen illustrations were captured and formatted with the Demo II software.

Trademarks

The IBM® PS/2® and AT®, and DOS are products of International Business Machines Corp, P.O. Box 1328, Boca Raton, Florida 33429-1328.

Paradox® is a registered trademark of Borland International, 1800 Greenhills Road, P.O. Box 660001, Scotts Valley, California 95066-0001.

WordPerfect® is a product of WordPerfect® Corporation, 1555 N. Technology Way, Orem, Utah 84057

Dan Bricklin's™ Demo II is a product of Software Garden™, Inc, PO Box 373, Newton Highlands, Massachusetts 02161.

TABLE OF CONTENTS

PREFACE	iii
Executive Summary	iii
ACKNOWLEDGEMENTS	v
People who made it happen	v
How we assembled the document	v
Trademarks	v
CHAPTER 1 - INTRODUCTION	1-1
What is the NRA?	1-1
Information Distribution Philosophy	1-2
Distributed Subset	1-2
Other Subsets	1-2
Access Software	1-3
NRADEMO vs NRA Distributed Access	1-3
Paradox®	1-3
Starting from Paradox®	1-4
ZOO Utility	1-4
Hardware	1-4
Operating System	1-5
Documentation Suite	1-5
Scope and Purpose of Programmer's Guide	1-6
Scope and Purpose of Data Loading Guide	1-6
Scope and Purpose of User's Guide	1-6
Directory Tree	1-6
Version 1.0	1-8
Plans for Enhancement	1-8

CHAPTER 2 - \NRADEMO	2-1
NRADEMO Directory Tree	2-1
NRA.BAT	2-1

CHAPTER 3 - \UTIL	3-1
UTIL Directory Tree	3-1
Directory of \NRADEMO\UTIL	3-2
DOS Batch Processors	3-2
UTIL\NRA.BAT	3-3
UTIL\OLDPATH.BAT\	3-3
Paradox® DATABASES TABLES FOR BUILDING MENU SCRIPTS	3-4
MAKEMENU MENU GENERATOR, v 2.0	3-4
UTIL\MENULEVE.DB	3-7
UTIL\MENUCHOI.DB	3-7
Paradox® SCRIPTS	3-8
UTIL\MAIN.SC	3-8
UTIL\DEMOUTIL.SC	3-9
MasterView	3-11
MasterViewSetUp	3-13
PrintClinical	3-15
PrintScreen	3-16
PrintSum	3-19
FileSum	3-20
PrintAni	3-21
FPrintAni	3-24
PrintTeffect	3-27
FPrintTeffect	3-29
PrintTDose	3-31
FPrintTDose	3-33
PrintInv	3-35
FileInv	3-36
PrintDetail	3-37
FPrintDetail	3-38
PrintIAni	3-39
FPrintIAni	3-41
PrintMaster	3-44
FPrintMaster	3-45
PrintLocation	3-46
FPrintLocation	3-47
PrintBib	3-48
FileBib	3-49
PrintBook	3-50

FPrintBook	3-53
PrintAuthor	3-56
FPrintAuthor	3-58
InitPrintVars	3-60
StartOfPage	3-61
FStartOfPage	3-62
EndOfPage	3-63
FEndOfPage	3-64
AddLineNumber	3-65
FAddLineNumber	3-66
WordWrap	3-67
HelpKey	3-69
TitleScreen	3-70
License1	3-73
License2	3-75
ErrorHandler	3-77
ErrorPrint	3-79
SetPopUp	3-82
PopUp	3-84
PrintProc	3-90
TextEditor	3-93
SearchBox	3-96
AuthorSearch	3-98
TitleSearch	3-102
YearSearch	3-105
SumAnimalSearch	3-108
SumAssignedIdSearch	3-111
InvAnimalSearch	3-114
InvAssignedIdSearch	3-118
TEffectSearch	3-122
TDoseSearch	3-125
SumStudySearch	3-129
InvStudySearch	3-134
UTIL\MENUMADE.SC	3-139
INIT	3-139
BACKONE	3-139
FIRSTONE	3-139
QUITPROC	3-140
MENU0	3-140
MENU2	3-143
MENU3	3-146
MENU4	3-149

MENU5	3-152
MENU6	3-155
MENU23	3-158
MENU33	3-161
MENU43	3-164

CHAPTER 4 - \MAINT	4-1
Directory of E:\NRADEMO\MAINT	4-1
DOS BATCH PROCESSORS	4-3
COMPRESS.BAT	4-3
INSTALL.BAT	4-3
MAKEDISK.BAT	4-5
NRA.BAT	4-6
Paradox® SCRIPTS	4-7
MAINT\MAINTAIN.SC	4-7
MAINT\MAINTUTL.SC	4-8
ADDTOSUM	4-9
ADDTOBIB	4-19
SEARCHBOX	4-23
EMPTYSUM	4-25
EMPTYINV	4-27
EMPTYBIB	4-28
COMPRESSDEMO	4-29
MAKEDISK3	4-30
MAKEDISK5	4-32

CHAPTER 5 - \PDOXRUN	5-1
PDOXRUN directory tree	5-1
Directory of E:\NRADEMO\PDOXRUN	5-1
What is Paradox® Runtime?	5-2
Distribution Diskette Labeling Requirement	5-2
Paradox® Runtime Disclaimer	5-3

CHAPTER 6 - DATABASE DIRECTORIES	5-1
Database directory trees	6-1
\NRADEMO\B	6-1
\NRADEMO\B\D	6-2
\NRADEMO\B\V	6-3
\NRADEMO\I	6-4
\NRADEMO\I\D	6-4
\NRADEMO\I\V	6-4

\NRADEMO\S	.6-6
\NRADEMO\S\D	.6-6
\NRADEMO\S\V	.6-7
\NRADEMO\W	.6-8

CHAPTER 7 - TEXT FILE DIRECTORIES	7-1
HELP and TEXT directory trees	7-1
\NRADEMO\TEXT	7-2
\NRADEMO\HELP	7-3

CHAPTER 8 - \DOCUMENT	8-1
DOCUMENT directory tree	8-1
Document File Naming Convention	8-2
\DOCUMENT	8-3
\DOCUMENT\MENUPIX	8-3
\DOCUMENT\PIX	8-4
\DOCUMENT\RPTS	8-4
\DOCUMENT\MENU	8-5
\DOCUMENT\DATALOAD	8-5
\DOCUMENT\PROGRAM	8-5
\DOCUMENT\PROGRAM\CHAPTERS	8-6
\DOCUMENT\PROGRAM\DRAFT_1	8-6
\DOCUMENT\PROGRAM\DIRS	8-7

CHAPTER 1 - INTRODUCTION

This Programmer's Guide describes the software associated with the National Radiobiology Archives (NRA) Distributed Access package. The package consists of a **distributed subset** of information representative of the NRA databases and database access software which provide an introduction to the scope and style of the NRA Information Systems. The system is implemented on an IBM® PS2/80 with Paradox® 3.5 database management software.

What is the NRA?

The National Radiobiology Archives is a comprehensive effort to gather, organize, and catalog original data, representative specimens, and supporting materials related to significant radiobiology studies. This provides researchers with information for analyses which compare or combine results of these and other studies and with materials for analysis by advanced molecular biology techniques. The NRA concentrated initially on studies of beagle dogs exposed to ionizing radiation at Argonne National Laboratory (ANL), the Inhalation Toxicology Research Institute (ITRI), the Pacific Northwest Laboratory (PNL), the University of California at Davis (DAVIS), and the University of Utah (UTAH). Rodent studies, primarily those conducted at Oak Ridge National Laboratory (ORNL) and at ANL are being added. The NRA will be receiving a donation of information from the life-span study of beagle dogs conducted at Colorado State University (CSU).

The NRA uses a three task approach to the challenge of organizing material from diverse sources. The microcomputer based NRA **Information Systems** include: experimental design documentation, animal dose-effect summaries, laboratory specific detailed databases, bibliographic citations, and an inventory of the documents and specimens. The NRA works closely with database managers at the participating laboratories to insure that electronic information is accurately translated into the NRA format. The initial donation to the NRA **Document Archives** is the collection of reprints and other materials supporting his book *Radioactivity and Health, A History* by J. Newell Stannard. The NRA **Specimen Archives** has tissue specimens and histopathology blocks from more than 1000 dogs donated by DAVIS. The specimens are organized and housed in a protected environment. These databases and archives are maintained through bar code readable labels.

Computer database technology is essential to integrating such a broad and diverse collection of information. The NRA Information Systems has developed several interrelated databases, each of which follows the relational model. There are three major databases: the Dose-effects Summary, the Collection Inventory, and the Bibliography. These are commonly referred to as the SUMMARY, INVENTORY, and BIBLIOGRAPHY databases and are illustrated in the NRA Distributed Access Package. In addition, the information systems contains many detailed database tables. These are copies of laboratory specific files translated from various computers and information management systems to the microcomputer environment and a common database management system. These DETAILED databases are not included in the NRA Distributed Access Package.

Information Distribution Philosophy

Information in the NRA Summary, Inventory, and Bibliography database is available on request. Printed reports have been provided in the past. The completion of the NRA Distributed Access Package is the realization of a long standing goal of the NRA staff and advisory committee. Now, information may be easily distributed to the user in an electronic form which preserves the relationships between the various database tables.

In the future, a subset of requested information will be extracted by the NRA staff and placed on diskette along with the access software. The User's Guide and the diskette(s) will be sent to the information requester.

Distributed Subset

The NRA Distributed Access Package includes a representative subset of the extensive NRA Information Systems database maintained at the Pacific Northwest Laboratory. The introductory subset consists of summary records for 100 control animals from various laboratories. It also contains 100 typical records from the bibliographic citation database. Inventory records showing the location of animal specimens and cited documents are also included. The introductory subset is described in the User's Guide. Instructions for download of this subset are given in the "Data Loading Guide".

Other Subsets

Other subsets, or the full NRA Information Systems database, may be accessed through menus provided with the NRA Distributed Access Package. The NRA will select and distribute subsets of information upon request. Instructions for preparation of a subset and moving it to the Distributed Access package are given in the Data Loading Guide.

Access Software

The Distributed Access menus, forms, and reports are independent of the data set. They comprise a relational database management system configured especially for the National Radiobiology Archives. The Distributed Access software has been designed to show the breadth of the NRA Information Systems by allowing the user to browse records with the databases' associated main forms. Several simple searches have been included to illustrate potential data extractions. Printed reports may be produced. The Distributed Access system is menu driven, and on-screen help is available at all times.

NRADEMO vs NRA Distributed Access

In the summer of 1991, the working version of the access software was called NRADEMO and was stored in subdirectory \NRADEMO. Our early evaluations of the package convinced us that it was more than "just a demonstration". The menu system has the ability to work with any subset of the NRA database. It is not limited to just a few records. Therefore, we changed the name of the package from NRADEMO to "NRA Distributed Access". Thus, this guide is titled "Distributed Access" while it describes software called "NRADEMO".

The access software, and the distributed database subset, is stored at the National Radiobiology Archives under the directory name "NRADEMO". Although the master data set is stored in a directory named "NRADEMO", to be consistent with the application's name the maintenance scripts create and use a directory structure called "NRA" on the NRA Distributed Access Package diskettes. The internal copy will continue to be stored in E:\NRADEMO since that path is embedded in the WordPerfect® subdocument names.

Paradox®

The NRADEMO system was developed in Paradox®, a database management product of Borland International, for which the NRA project holds a license. A working knowledge of Paradox® and PAL are necessary for understanding this Programmer's Guide and the NRA Distributed Access software.

The system will run under Paradox® 3.5 or it may be run using the run time application version of Paradox®. The latter method allows running the system on computers which do not have the Paradox® software. This feature makes NRADEMO portable.

Starting from Paradox®

The NRA maintenance programmer should run the application directly from Paradox®, then test any modifications using the runtime software. To run invoke the NRA Distributed Access system from Paradox®, enter:

**F10
TOOLS
MORE
DIRECTORY
E:\NRADEMO
OK**

**F10
SCRIPTS
PLAY
UTIL\MAIN**

ZOO Utility

To facilitate the portability, we have provided the public domain file compression/expansion utility program, ZOO Version 1.1, written by Rahul Dhesi in 1986. After the NRADEMO system is archived from hard disk to a floppy using ZOO (a process which is automated by MAINT\MAKEDISK.BAT), the system can be imported onto another computer and used under the run time Paradox® software. A copy of ZOO is included on the distribution diskettes. See the file ZOO.HLP for usage instructions.

Hardware

The NRA Distributed Access Package may be implemented on any IBM® PS/2®, AT® or fully-compatible computer configured with either a 5-1/4 inch or 3-1/2 inch high density floppy disk drive. A minimum of 3.5 megabytes of available on-line disk storage is necessary to install the software. A minimum of 512 kilobytes of RAM should be available. If printed reports are desired, a printer must be available to the computer.

Operating System

This software will execute under DOS 3.3 or later version.

Documentation Suite

This document is part of a suite of documents describing the NRA Information Systems and the Distributed Access Package.

<i>Document Title</i>	<i>Audience</i>	<i>Publication Date</i>
National Radiobiology Archives Database Documentation: A. Overview	General	
National Radiobiology Archives Database Documentation: B. Design	General	
National Radiobiology Archives Database Documentation: C. Table Structures	NRA, NRAAC	Apr '91
Distributed Access User's Guide	General	Oct '91
Distributed Access Programmer's Guide	NRA	Oct '91
Distributed Access Data Loading Guide	NRA	Oct '91
MAKEMENU Application Generator User's Guide	NRA	Sep '91

Scope and Purpose of Programmer's Guide

The Programmer's Guide is designed for the maintenance programmer. It includes an overview of the NRA distributed access concept, a discussion of hardware and software environments, copies of all PAL scripts, and directories of all associated database tables. The Programmer's Guide is mostly devoted to the PAL scripts. Details of database organization will be found in the Table Structures document. The E: drive is the name of the physical drive where the software and documentation reside. The N: drive is the name of the physical drive where the complete NRA database resides. Since WordPerfect® requires full path names for the retrieval of subdocuments and Paradox® requires drive letters for linking tables across drives, the drive letters E: and N: are hardcoded in the documentation. If it is desirable to move the system to a different drive, the DOS SUBST command can be used to handle the generation of the WordPerfect® document while search and replace command will handle any hard codes which may exist in the Paradox® scripts.

Scope and Purpose of Data Loading Guide

The Data Loading Guide is designed for the database administrator. The guide includes an overview of the NRA distributed access concept, a discussion of hardware and software environments, and instructions for extracting a subset of the larger NRA database and moving it to the Distributed Access Package.

Scope and Purpose of User's Guide

The User's Guide is designed for the user. It includes an overview of the NRA distributed access concept, a discussion of hardware requirements, instructions for installing and executing the software on the user's computer, descriptions and examples of menus, instructions for printing reports of database contents, a description of the database tables and alphabetized reference sections.

Directory Tree

The system is stored in several directories on the E: disk drive. This Programmer's Guide is organized around the subdirectories. The tree diagram below shows the name of the subdirectory, the chapter in this guide where the subdirectory is discussed, and the logical purpose of the structure. Subdirectories containing the application software discussed in this guide are shaded.

<i>directory tree node</i>	<i>chapter</i>	<i>purpose of node</i>
E:\		
└─NRADEMO	2	Root, startup batch files
└─B	6	Bibliography root
└─D	6	Bibliography data tables
└─V	6	Bibliography validation tables
└─DOCUMENT	8	Documentation WP files
└─PROGRAM	8	WP file for this guide
└─CHAPTERS	8	Subdocuments
└─DRAFT_1	8	first draft
└─DIRS	8	Subdocuments
└─PIX	8	Subdocuments for User's Guide
└─MENUPIX	8	Subdocuments for User's Guide
└─RPTS	8	Subdocuments for User's Guide
└─HELP	7	Help Text files
└─I	6	Inventory root
└─D	6	Inventory data tables
└─V	6	Inventory validation tables
└─MAINT	4	PAL scripts for data maintenance
└─PDOXRUN	5	Paradox [®] runtime software
└─S	6	Summary database root
└─D	6	Summary data tables
└─V	6	Summary validation tables
└─TEXT	7	Individual animal text files
└─UTIL	3	PAL scripts for the application
└─W	6	Work tables

The bulk of this document is devoted PAL scripts. In the course of this project, we expanded our general purpose menu maker. We describe it in a companion document, "MAKEMENU Application Generator User's Guide". We used menu maker to develop this application. The tables for menu generation and the generated menu script are described in Chapter 3, which documents subdirectory \UTIL; it exceeds 150 pages. Chapter 4, devoted to subdirectory \MAINT is over 30 pages.

The database directories, B, I, and S, are briefly described in chapter 6; they are discussed in detail in "National Radiobiology Archives Database Documentation - C. Table Structures".

Version 1.0

Version 1.0 was released on 10/17/91 under the name, "NRA Distributed Access". The User's Guide and diskettes were distributed to the NRA user community for evaluation and comment. The Introductory Subset of Data includes records associated with 100 animals and 100 books.

Backup of Version 1.0 is on the NRA optical disk: NRA-6, and on diskettes stored with the master copy of this document.

Plans for Enhancement

The application needs several enhancements to become a robust and useful application. The menu system needs to be redesigned to reduce the number of global variables since these are likely to cause side effects leading to maintenance problems and use RAM which slows down the application. The help system should be redone using a C program to read in and display text files. The application needs the ability to query, manipulate and export the data. A Query-by-Form technique of querying the data will allow the user to use the Query-by-Example method of data retrieval. The ability to manipulate the data and narrow down the extraction sets is possible if one is given the ability to query the "answer" table generated by previous queries and not just the tables supplied with the application. The export capabilities of Paradox® which are part of the Paradox® Tools will create delimited ASCII records while the report capabilities will handle the export of fixed-width records. The system should mature into a useful data manipulation tool with these enhancement which are detailed in the following paragraphs of this section.

The menu system can be designed with minimal use of global variables which will speed up the application and also free up valuable RAM. Currently the menu system keeps track of the titles, colors, dimensions, and screen locations of all the previous menus in global variables. The title, colors, dimensions, and screen locations are known for each menu level before runtime thus the code can be written without these global variables. Two or more procedures will be required for each menu level. One procedure will be required to draw and refresh the screen as necessary, and the other will be required to call various procedures including one to highlight the current menu choice. The implementation of this new system will require some effort but will result in a robust system with few global variables.

The help system currently uses Paradox® to read in text files and parse them into various sized lines with a popup procedure which is slow and has few options. The system should adapt a C program which can read in text files into an adjustable window which can be placed anywhere on the screen. The program should be able to print the file, search the file, and save a copy of the file onto a diskette. This program could be used both for displaying

the help files of the system and the clinical text files.

The querying capability of the system is currently limited to prepackaged search routines. The system should adapt a Query-by-Form or Query-by-Table system to use a Query-by-Example (QBE) method for retrieving data. This is a good ad hoc method of querying which is powerful and flexible. The Paradox® Runtime does not support the WAIT feature with query images, thus one must create tables with identical field names with alphanumeric field types which will allow the entry of QBE statements in all the fields. Paradox® must then transfer the data entered in all the fields to query images which are then processed. The resulting answer table can then be queried again if necessary to further filter the data for analysis.

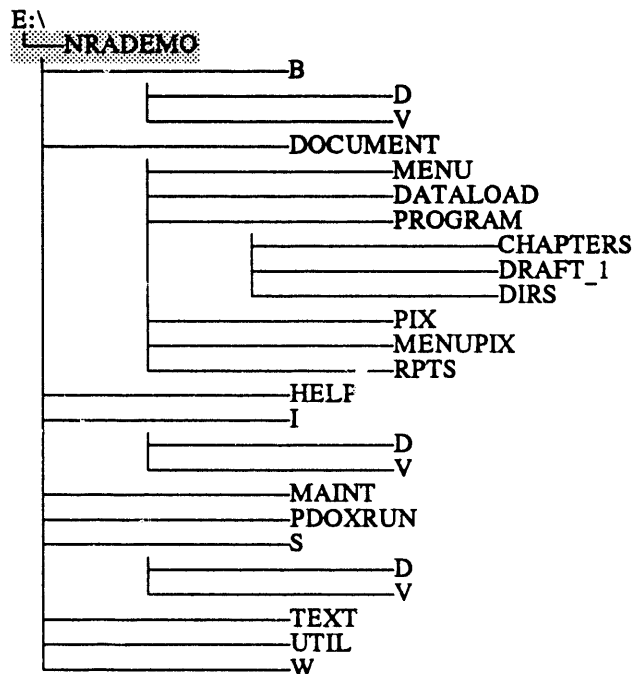
The data export is currently limited to reports intended for reading rather than data analysis. Paradox® supports export to delimited ASCII files though the Tools menu and one can create fixed width reports using the Paradox® report writer. The fixed width report may have problems with numeric data containing decimals since numbers default in the report writer to integers. The problems can be resolved with additional code.

With the above changes to the application implemented, the NRA Distributed Access System will be on its way to becoming robust and useful tool for analyzing the data of the National Radiobiology Archives Information Systems.

CHAPTER 2 - \NRADEMO

NRADEMO Directory Tree

This chapter describes the NRADEMO subdirectory which contains only one batch file, NRA.BAT. This directory is called NRA in the distribution copies.



NRA.BAT

A batch file, NRA.BAT which starts the NRA Distributed Access application is the only file stored in the root directory, \NRADEMO. The user types in NRA to invoke the runtime version of the application. The NRA maintenance programmer will usually run the application directly from Paradox®. The runtime version is slightly limited--the F1 lookup

and fill is not available.

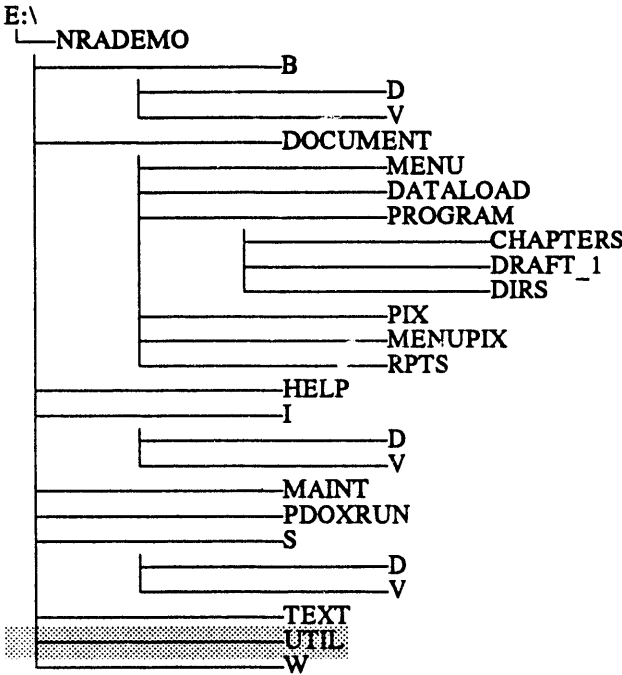
A copy of this file is also stored in the \UTIL subdirectory. It is copied to \NRADEMO when the system is archived from the hard disk to a floppy using MAKEDISK.BAT. That process erases all files in \NRADEMO before it copies the batch file. This insures that the user does not clutter up this directory.

```
echo off
echo One moment please...
path >util\oldpath.bat
path = pdoxrun
pdoxrun util\main
util\oldpath
cls
```

CHAPTER 3 - \UTIL

UTIL Directory Tree

The \UTIL directory contains software and utilities necessary for NRADEMO system functionality:



Directory of \NRADEMO\UTIL

Directory of E:\NRADEMO\UTIL

.	<DIR>	8-25-91	3:47p
.	<DIR>	8-25-91	3:47p
DEMOUTIL LIB	223893	8-25-91	1:20p
DEMOUTIL SC	105441	8-25-91	1:20p
MAIN SC	1141	8-24-91	9:39a
MAKEMENU SC	29280	8-24-91	10:45a
MENUCHOI DB	36864	8-24-91	1:20p
MENUCHOI F1	738	8-24-91	1:21p
MENUCHOI F2	946	8-24-91	1:21p
MENUCHOI F3	1488	8-24-91	1:22p
MENUCHOI PX	4096	8-24-91	1:20p
MENUCHOI R	4063	8-24-91	1:21p
MENUCHOI SET	200	8-24-91	1:20p
MENULEVE DB	4096	8-24-91	1:56a
MENULEVE F	968	8-24-91	1:56a
MENULEVE F1	1901	8-24-91	12:06p
MENULEVE F2	1253	8-24-91	1:56a
MENULEVE PX	4096	8-24-91	1:56a
MENULEVE R	3055	8-24-91	1:56a
MENULEVE SET	146	8-24-91	1:56a
MENULEVE VAL	115	8-24-91	1:56a
MENUMADE LIB	146229	8-24-91	10:47a
MENUMADE SC	52472	8-24-91	10:47a
MENUUTIL LIB	38767	8-25-91	1:12p
MENUUTIL SC	16909	8-25-91	1:12p
NRA BAT	116	7-30-91	2:20p
OLDPATH BAT	100	8-22-91	10:19p

27 File(s)

Generated by DEMOUTIL.SC
Application specific code
Script to start the demonstration
See MENU MAKER document
Table containing the menu choices
menu development screen
menu development screen
menu development screen
database table index
menu development report
database table settings
Table containing the menu levels
menu development screen
menu development screen
menu development screen
database table index
menu development report
database table settings
database table validations
Product of MAKEMENU.SC
Product of MAKEMENU.SC
See MENU MAKER document
Product of MENUUTIL.SC
Invokes application
Restores previous path

The DOS batch processors, the database tables containing the menus, and the Paradox[®] scripts are printed below. All 27 files are necessary for operation of NRADEMO.

DOS Batch Processors

Two DOS batch files are stored in \UTIL. NRA.BAT invokes the application, while OLDPATH.BAT restores the previous path.

UTIL\NRA.BAT

```
echo off
echo One moment, please...
path >util\oldpath.bat
path = pdoxrun
pdoxrun util\main
util\oldpath
cls
```

UTIL\OLDPATH.BAT\

```
PATH=C:\NETWORK;C:\DOS;C:\PDOX35;C:\PARADOX3;E:\WP51;S:\LIBRARY;S:\C
MAIL;C:\ST240;C:\;C:\FCTOOLS;
```

Paradox® DATABASES TABLES FOR BUILDING MENU SCRIPTS

The menu system is based on a general purpose menu maker described in "National Radiobiology Archives MENU MAKER Programmer's Manual". The menu stored in a table UTIL\MENULEVE and UTIL\MENUCHOI and is processed by a script UTIL\MAKEMENU. The result is automatically placed in a script UTIL\MENUMADE. The completed application is invoked by a script UTIL\MAIN.

The following screens show the data entry process for building a menu system :

```
MAKEMENU MENU GENERATOR, v 2.0
Epidemiology and Biometry Dept., Battelle, PNL, 1991
How it works:
The programmer enters the data required (to make the menu library) using
the form provided on the next screen. The data is saved in two tables:
    UTIL\MENULEVE = holds menu level data, ie title name, colors,...
    UTIL\MENUCHOI = menu choice data, ie choice name, action, library...
When the data is entered and saved, the programmer can either create the menu
library as "UTIL\MENUMADE" by pressing [F6]... or change the library name by
pressing [F5], typing in the new library name, and then [F6] to create the
library.
For the programmer to tie the new menu system into the application, in the
script that calls the menu system he/she must:
    1) Set the autolib: AUTOLIB = "UTIL\MENUUTIL,libname", where libname is
        the name of the menu library created.
    2) Call the first menu of the menu system: MENUO()
A few rules to remember:
    1) To call another menu, type "menu" + the menu level desired + "(" in the
        action field. ie MENU157()...
    2) If you desire a main title for your application, enter it on level "-1".
    3) When typing in a path name inside quotes, you must use double backslashes.
    4) The first menu level MUST be numbered "0" for the menus to work.
    5) For best results, use color1 and color2 with the same color background.
    6) The background color can be changed under "bkcolor" in "INIT" procedure.
        Hit any key to continue...
```

F1-HELP CTRL(Z)-ZOOM ESC-QUIT MENUMAKER VIEWING RECORD 2 OF 10
 F9-EDIT F3-UPIMAGE F4-DNIMAGE F7-FORM TOGGLE F5-CHANGE LIBNAME F6-MAKE MENU LIB
 MAKEMENU ENTRY FORM -CONDENSED

LEVEL	TITLE	CLR1	CLR2
0	NRADEMO MAIN MENU	23	30

#	CHOICE	DESCRIPTION	ACTION	LIBRARY	HELP
1	Information	Info about the Natio	TEXTEDITOR(	"UTIL\\DEMO	Information
2	Summary	Access the NRA Summa	MENU2()		Summary wil
3	Inventory	Access the NRA Inven	MENU3()		Inventory w
4	Bibliograph	Access the NRA Bibli	MENU4()		Bibliograph
5	Detail	Access the NRA Detai	MENU5()		Detail will
6	HELP	Information about th	MENU6()		HELP will d
7	Quit	Quit NRADEMO, return	QUITPROC()		Quit is use

F1-HELP CTRL(Z)-ZOOM ESC-QUIT MENUMAKER VIEWING RECORD 2 OF 10
 F9-EDIT F3-UPIMAGE F4-DNIMAGE F7-FORM TOGGLE F5-CHANGE LIBNAME F6-MAKE MENU LIB
 MAKEMENU ENTRY FORM -EXPANDED

LEVEL	TITLE	CLR1	CLR2
0	NRADEMO MAIN MENU	23	30

#: 1	NAME:Information DESCRIPTION:Info about the National Radiobiology Archives. ACTION:TEXTEDITOR("TEXT\\NRA.TXT","NATIONAL RADIOBIOLOGY ARCHIVES") LIBRARY:"UTIL\\DEMOUTIL" HELP:Information will pop up a window which describes the National Radiobiology Archives.
#: 2	NAME:Summary DESCRIPTION:Access the NRA Summary Database. ACTION:MENUE2() LIBRARY: HELP:Summary will pop up the Summary menu. The Summary database contains information about individual animals.

UTIL\MENULEVE.DB

There are 4 levels in the maintenance menu

10/18/91

Menu Level Table

Page 1

Menu level	Menutitle	Menu color1	Menu color2
-1	NRADEMO MAINTENANCE PROGRAM	14	7
0	MAIN MENU	7	14
1	EMPTY TABLES	7	14
2	ADD RECORDS	7	14
4	MAKE DISKETTES	7	14

UTIL\MENUCHOI.DB

10/18/91

Menu choice Table

Page 1

Menu level	Menu choice	Choicename	Choicedescript	Action	Autolib
0	1	Empty tables	Empty Summary, inventory, or Bibliography tables.	MENU1()	
0	2	Add records	Add records to Summary or Bibliography (& inv).	MENU2()	
0	3	Compress	Compress NRADEMO.	COMPRESSDEMO()	"MAINT\MAINTUTL"
0	4	Make disks	Make distribution disks from compressed NRADEMO.	MENU4()	
0	5	Quit	Quit NRADEMO maintenance program.	QUITPROC()	
1	1	Summary	Empty NRADEMO Summary tables.	EMPTYSUM()	"MAINT\MAINTUTL"
1	2	Inventory	Empty NRADEMO inventory tables.	EMPTYINV()	"MAINT\MAINTUTL"
1	3	Bibliography	Empty NRADEMO Bibliography tables.	EMPTYBIB()	"MAINT\MAINTUTL"
1	4	Return	Return to previous menu.	BACKONE()	
1	5	Quit	Quit NRADEMO maintenance program.	QUITPROC()	
2	1	Summary	Add records to the NRADEMO Summary (& inv).	ADDTOSUM()	"MAINT\MAINTUTL"
2	2	Bibliography	Add records to the NRADEMO Bibliography (& inv).	ADDTOBIB()	"MAINT\MAINTUTL"
2	3	Return	Return to previous menu.	BACKONE()	
2	4	Quit	Quit NRADEMO maintenance program.	QUITPROC()	
4	1	3 1/2	Send compressed NRADEMO to 3 1/2 inch diskettes.	MAKEDISK3()	"MAINT\MAINTUTL"
4	2	5 1/4	Send compressed NRADEMO to 5 1/4 inch diskettes.	MAKEDISK5()	"MAINT\MAINTUTL"
4	3	Return	Return to previous screen.	BACKONE()	
4	4	Quit	Quit NRADEMO maintenance program.	QUITPROC()	

Paradox® SCRIPTS

UTIL\MAIN.SC

Script MAIN calls the title screen and the first menu.

```
;/*****  
;* FILE: UTIL\MAIN.SC  
;* MODULE: 0.0  
;* PURPOSE: Calls the title screen and first menu of the NRADEMO application.  
;* CALLED BY:  
;* CALLS:  
;* LIBRARY CREATED: None  
;* GLOBAL VARIABLES:  
;* PROCEDURES IN MODULE: None  
;*  
;*  
;* AUTHOR: Jonathan C. Prather  
;* DATE: 8/11/91  
;*  
;* REVISION: Sean K. Smith added calls to LICENSE1 and LICENSE 2 on 9/26/91  
;*****/  
  
;free memory  
CLEAR  
CURSOR OFF  
RELEASE VARS ALL  
RELEASE PROCS ALL  
  
;set autolibrary to run menu system  
AUTOLIB = "UTIL\MENUUTIL,UTIL\MENUMADE"  
  
;define error procedure from util\errorutl library  
READLIB "UTIL\DEMOUTIL" ERRORHANDLER  
ERRORPROC = "ERRORHANDLER"  
  
; Call procedure to initialize global variables  
INIT()  
  
; Call the title screen of the application  
READLIB "UTIL\DEMOUTIL" TITLESCEEN  
TITLESCEEN()  
  
; Call the DOE license agreement  
READLIB "UTIL\DEMOUTIL" LICENSE1  
LICENSE1()  
  
; If using RUNTIME, call the Paradox disclaimer  
IF ISRUNTIME()  
  THEN  
    READLIB "UTIL\DEMOUTIL" LICENSE2  
    LICENSE2()  
  ENDF  
; Call the first menu of the application  
MENUO()  
  
;/*****
```

```

; * END MODULE: 0.0 MAIN
; *****/

```

UTIL\DEMOUTIL.SC

```

; *****
; * FILE: DEMOUTIL.SC
; * MODULE: x.x
; * PURPOSE:
; * CALLED BY: Procedures in library created by MAKEMENU
; * CALLS:
; * LIBRARY CREATED: util\demoutil
; * GLOBAL VARIABLES: highlight, bkcolor
; * PROCEDURES IN MODULE: MasterView
; *                               MasterViewSetUp
; *                               PrintClinical
; *                               PrintScreen
; *                               PrintSum
; *                               FileSum
; *                               PrintAni
; *                               FPrintAni
; *                               PrintTeffect
; *                               FPrintTeffect
; *                               PrintTDose
; *                               FPrintTDose
; *                               PrintInv
; *                               FileInv
; *                               PrintDetail
; *                               FPrintDetail
; *                               PrintIAni
; *                               FPrintIAni
; *                               PrintMaster
; *                               FPrintMaster
; *                               PrintLocation
; *                               FPrintLocation
; *                               PrintBib
; *                               FileBib
; *                               PrintBook
; *                               FPrintBook
; *                               PrintAuthor
; *                               FPrintAuthor
; *                               InitPrintVars
; *                               StartOfPage
; *                               FStartOfPage
; *                               EndOfPage
; *                               FEndOfPage
; *                               AddLineNumber
; *                               FAddLineNumber
; *                               WordWrap
; *                               HelpKey
; *                               TitleScreen
; *                               License1
; *                               License2
; *                               ErrorHandler
; *                               ErrorPrint
; *                               SetPopUp
; *                               PopUp
; *                               PrintProc
; *                               TextEditor
; *                               SearchBox
; *                               AuthorSearch
; *                               TitleSearch
; *                               YearSearch
; *

```

```

/*
/*      SumAnimalSearch
/*      SumAssignedIdSearch
/*      InvAnimalSearch
/*      InvAssignedIdSearch
/*      TEffectSearch
/*      TDoseSearch
/*      SumStudySearch
/*      InvStudySearch
/*
/*  AUTHOR: Jonathan C. Prather
/*  DATE: 8/22/91
/*  REVISED 9/18/91 BY Chuck Watson
/*      MasterView - case "F5" - changed location of text file
/*      HelpKey - changed location of help files
/*  REVISED 9/26/91 BY Sean K Smith
/*      YEARESEACH() - changed PopUp call title from year to
/*                      year of publication.
/*      MasterView() - Removed sections of code to MasterSetUp()
/*                      and PrintClinical() to slim down. Also
/*                      modified added Otherwise to switch to
/*                      handle common key presses.
/*      MasterViewSetUp() - added this proc to slim down
/*                          MasterView
/*      PrintClinical() - added to slim down MasterView
/*      PrintScreen() - Added procedures From PrintScreen()
/*                      through WordWrap() in order to print
/*                      the current record.
/*
/******
LIBNAME = "UTIL\\DEMOUTIL"
@ 24,0 CLEAR EOL ?? "CREATING "+LIBNAME+" LIBRARY"
STYLE BLINK ?? "..."
STYLE
CREATLIB LIBNAME SIZE 75 ;/* create library to hold procedures in this file */

```

```

;*****
; * PROCEDURE: MasterView()
; * MODULE:
; * PURPOSE:
; * CALLED BY:
; * CALLS:
; * INPUT ASSERTION:
; * OUTPUT ASSERTION:
; * GLOBAL VARIABLES:
; * AUTHOR: J. C. Prather
; * DATE: August 1991
; * REVISED 9/18/91 BY: Chuck Watson
;                      Clinical text file now located in directory \text
;                      rather than \text\clinical. See Case "F5"
;*****/
PROC MasterView(Tbl, frm)

    MasterViewSetUp()

    While TRUE
        if imageno()=1 then
            therecordno = strval(recno())
        endif
        PRMPTSPC=FILL(" ",14-(LEN(STRVAL(NRECORDS(TBL)))+LEN(THERECORDNO)))

        Wait TABLE
        PROMPT "F1-HELP      [CTRL][Z]-ZOOM  ESC-RETURN TO MENU "+PRMPTSPC+"VIEWING RECORD
"+therecordno+" OF "+STRVAL(NRECORDS(tbl)),PRMPT2
        Until "F1","F3","F4","F5",-81,-73,-72,-80,-71,-79,-75,-77,27,"F6"
        key = RetVal

        SWITCH
            Case key = "F1":
                HelpKey()

            Case key = "F3":      ;/* allow UpImage with animal form */
                if (inMultiForm and inFormView) then
                    UpImage
                    IF FRM="12" THEN
                        WHILE IMAGENO(>)3
                            UPIMAGE
                        ENDWHILE
                    ENDIF
                else
                    Beep
                endif

            Case key = "F4":      ;/* allow DownImage with animal form */
                if (inMultiForm and inFormView) then
                    DownImage
                    IF FRM="12" THEN
                        WHILE IMAGENO(>)3
                            DOWNIMAGE
                        ENDWHILE
                    ENDIF
                else
                    Beep
                endif

            Case key = "F5":      ;/* Print clinical text */

```

```

PrintClinical()

Case Key = "F6":          /* print the current record and linked records */
  IMAGEMEM = IMAGEN()
  FIELDMEM = FIELD()
  RECORDMEM = RECNO()

  MOVETO 1

  PrintScreen()

  MOVETO IMAGEMEM
  MOVETO RECORD RECORDMEM
  MOVETO FIELD FIELDMEM
  PICKFORM frm

Case key = 27: /* Esc key */
  STYLE ATTRIBUTE HIGHLIGHT
  @24,59 ?? " ONE MOMENT PLEASE"
  STYLE ATTRIBUTE HIGHLIGHT+128
  @24,77 ?? "...
  CLEARALL
  return FALSE

OTHERWISE:
  KEYPRESS key

endswitch

endwhile
clearall
ENDPROC
WRITELIB LIBNAME MasterView
RELEASE PROCS MasterView
;*****
/* END MODULE: x.x MasterView()
;*****

```

```

;*****
;* PROCEDURE: MasterViewSetUp()
;* MODULE:
;* PURPOSE:
;* CALLED BY:
;* CALLS:
;* INPUT ASSERTION:
;* OUTPUT ASSERTION:
;* GLOBAL VARIABLES:
;* AUTHOR:
;* DATE:
;*****/
PROC MasterViewSetUp()

; put up working message
STYLE ATTRIBUTE HIGHLIGHT
@24,59 ?? " ONE MOMENT PLEASE"
STYLE ATTRIBUTE HIGHLIGHT+128
@24,77 ?? "... "
CANVAS ON
CANVAS OFF
;*****

IF ISEMPY(Tbl) THEN BEEP
;DRAW BOX FOR MESSAGE
STYLE ATTRIBUTE HIGHLIGHT
PAINTCANVAS ATTRIBUTE BKCOLOR 24,0,24,79
@ 6,5 ?? " "+FILL("-",68)+" "
@ 7,5 ?? " "+FILL(" ",68)+" "
@ 8,5 ?? " "+FILL(" ",68)+" "
@ 9,5 ?? " "+FILL(" ",68)+" "
@ 10,5 ?? " "+FILL(" ",68)+" "
@ 11,5 ?? " "+FILL("-",68)+" "

@ 8,6 ?? FORMAT("W68,AC","TABLE IS EMPTY")
@ 9,6 ?? FORMAT("W68,AC"," Press any key to continue...")
CANVAS ON
CANVAS OFF
GC = GETCHAR()
RETURN
ENDIF

clearall
VIEW Tbl;
CURSOR OFF
IF (frm = "") then
inFormView = FALSE
inMultiForm = FALSE
ELSE
RequiredCheck Off
Pickform frm
RequiredCheck On
inMultiForm = IsMultiForm(Tbl, frm)
inFormView = TRUE
ENDIF

; /* set prompt depending if more than one table is present */
IF inMultiForm then
prmt2 = "F3-UPIMAGE F4-DNIMAGE F6-PRINT
HOME,END,"+chr(24)+" "+chr(25)+" PGUP,PGDN-SCAN RECORDS"

```

```

ELSE
    prmp2 = "                HOME,END,"+chr(24)+",""+chr(25)+",PGUP,PGDN-SCAN RECORDS
ENDIF

IF TBL = "S\\D\\ANI" OR FRM = "12" THEN
    prmp2 = "F3-UPIMAGE F4-DNIMAGE F5-CLN_TXT F6-PRINT
HOME,END,"+CHR(24)+",""+CHR(25)+",PGUP,PGDN-SCAN RECORDS"
ENDIF

canvas on

ENDPROC
WRITELIB LIBNAME MasterViewSetUp
RELEASE PROCS MasterViewSetUp
;*****
;*/
;* END MODULE: x.x MasterViewSetUp()
;*****/

```

```

; /*****
; * PROCEDURE: PrintClinical()
; * MODULE:
; * PURPOSE:
; * CALLED BY:
; * CALLS:
; * INPUT ASSERTION:
; * OUTPUT ASSERTION:
; * GLOBAL VARIABLES:
; * AUTHOR:
; * DATE:
; *****/
PROC PrintClinical()
    IF TBL = "S\D\ANI" OR FRM = "12" THEN
        IMAGEMEM= IMAGENO()
        RECORDMEM= RECNO()
        FIELDMEM= FIELD()
        IF IMAGENO() <> 1 THEN MOVETO 1
        ENDIF
        ANIMALNUM= [ANIMAL_ID]
        filename = "TEXT\\"+[ANIMAL_ID]+".txt"
        if isfile(filename) then
            CLINICALFLAG = "Y"
            TextEditor(filename,"CLINICAL TEXT OF ANIMAL: "+[ANIMAL_ID])
            RELEASE VARS CLINICALFLAG
            FORMKEY
            MOVETO IMAGEMEM
            MOVETO RECORD RECORDMEM
            MOVETO FIELD FIELDMEM
        else message "No text file on this animal included, press any key to continue"
            gc = getchar()
            MOVETO IMAGEMEM
            MOVETO RECORD RECORDMEM
            MOVETO FIELD FIELDMEM
            RELEASE VARS ANIMALNUM,RECORDMEM,IMAGEMEM,FIELDMEM
        endif
        ELSE BEEP
    ENDIF

ENDPROC
WRITE! LIBNAME PrintClinical
RELEASE PROCS PrintClinical
; /*****
; * END MODULE: x.x PrintClinical()
; *****/

```

```

;*****
;* PROCEDURE: PrintScreen()
;* MODULE:
;* PURPOSE: This is the main driver of the record printing routine and prints
;*           the current record on the screen and the record also linked to it
;*           on the current form.
;* CALLED BY: MasterView()
;* CALLS:
;* INPUT ASSERTION:
;* OUTPUT ASSERTION:
;* GLOBAL VARIABLES:
;* AUTHOR: Sean K. Smith
;* DATE: 9/26/91
;*****/
PROC CLOSED PrintScreen()
    USEVARS Tbl, Frm, AutoLib,HIGHLIGHT
    PRIVATE ANS,FILENAM

;init variables
ANS = ""

SHOWMENU "FILE":"Send Report to file...",
          "PRINTER":"Send Report to printer...",
          "EXIT":"Return to previous screen..."
TO ANS

SWITCH
CASE ANS = "FILE":

    WHILE (TRUE)

        FILENAM = ""
        STYLE ATTRIBUTE HIGHLIGHT
        @ 0,0 ?? FORMAT("W80,AL","Enter an 8 character file name: ")
        @ 1,0 ?? FORMAT("W80,AL","File with .RPT will be placed in " + DIRECTORY())
        STYLE ATTRIBUTE 31
        CURSOR NORMAL

        @ 0,31
        ACCEPT "A8" TO FILENAM

        IF ((RetVal = FALSE) OR (ISBLANK(FileNam))) THEN
            CLEAR
            QUITLOOP
        ENDIF

;/* check for illegal characters */
IF ((SEARCH(".",FileNam) <> 0) OR
    (SEARCH("\",FileNam) <> 0) OR
    (SEARCH("/",FileNam) <> 0) OR
    (SEARCH(":",FileNam) <> 0)) THEN

    WHILE (CHARWAITING())
        x = GETCHAR()
    ENDWHILE

    MESSAGE "ILLEGAL FILENAME -- TYPE ANY CHARACTER TO CONTINUE"
    x = GETCHAR()
    CLEAR
    LOOP

```

```

ENDIF

/* check to see if file exists */
IF (ISFILE(filenam + ".RPT")) THEN

    WHILE (CHARWAITING())
        x = GETCHAR()
    ENDWHILE

    MESSAGE "FILE EXISTS! -- TYPE ANY CHARACTER TO CONTINUE"
    x = GETCHAR()
    CLEAR
    LOOP
ENDIF

QUITLOOP
ENDWHILE

FileNam = FileNam + ".RPT"

PAINTCANVAS FILL " " ATTRIBUTE 79 0,0,1,79
STYLE ATTRIBUTE 78
@ 0,0?? FORMAT("W80,AC","Sending report to file: " +DIRECTORY()+FILENAM)

SWITCH
CASE ((Tbl = "S\D\ANI") OR (Tbl = "TANI")):
    report_title = "SUMMARY DATABASE REPORT"
    FileSum()

CASE ((Tbl = "I\D\DETAIL") OR (Tbl = "TDETAIL")):
    report_title = "INVENTORY DATABASE REPORT"
    FileInv()

CASE ((Tbl = "B\D\BOOK") OR (Tbl = "TBOOK")):
    report_title = "BIBLIOGRAPHY DATABASE REPORT"
    FileBib()

OTHERWISE:
    WHILE (CHARWAITING())
        x = GETCHAR()
    ENDWHILE

    MESSAGE "REPORT IS NOT CURRENTLY AVAILABLE - TYPE ANY KEY"
    x = GETCHAR()
    CLEAR
ENDSWITCH

RETURN

CASE ANS = "PRINTER":

PAINTCANVAS FILL " " ATTRIBUTE 79 0,0,1,79
STYLE ATTRIBUTE 78
@ 0,0?? FORMAT("W80,AC","Sending report to the printer")

SWITCH
CASE ((Tbl = "S\D\ANI") OR (Tbl = "TANI")):
    report_title = "SUMMARY DATABASE REPORT"
    PrintSum()
CASE ((Tbl = "I\D\DETAIL") OR (Tbl = "TDETAIL")):

```

```

        report_title = "INVENTORY DATABASE REPORT"
        PrintInv()
    CASE ((Tbl = "B\D\BOOK") OR (Tbl = "TBOOK")):
        report_title = "BIBLIOGRAPHY DATABASE REPORT"
        PrintBib()

    OTHERWISE:
        WHILE (CHARWAITING())
            > = GETCHAR()
        ENDWHILE

        MESSAGE "REPORT IS NOT CURRENTLY AVAILABLE - TYPE ANY KEY"
        x = GETCHAR()
        CLEAR
    ENDSWITCH

    RETURN

CASE ANS = "EXIT":
    RETURN

CASE ANS = "Esc":
    RETURN

OTHERWISE:
    BEEP

ENDSWITCH

RETURN

ENDPROC
WRITELIB LIBNAME PrintScreen
RELEASE PROCS PrintScreen
;*****
;* END MODULE: x.x PrintScreen()
;*****

```

```

;*****
;* PROCEDURE: PrintSum()
;* MODULE:
;* PURPOSE: Prints the Animal Summary form record and linked records.
;* CALLED BY: PrintScreen()
;* CALLS:
;* INPUT ASSERTION:
;* OUTPUT ASSERTION:
;* GLOBAL VARIABLES:
;* AUTHOR: Sean K. Smith
;* DATE: 9/26/91
;*****/
PROC PrintSum()

    InitPrintVars()

    OPEN PRINTER

        StartOfPage()
        PrintAni()           ;/* print the animal record */
        PrintTeffect()       ;/* print the TEffect records which are linked */
        PrintTDose()         ;/* print the TDose records which are linked */
        EndOfPage()

    CLOSE PRINTER

    CLEAR
ENDPROC
WRITELIB LIBNAME PrintSum
RELEASE PROCS PrintSum
;*****
;* END MODULE: x.x PrintSum()
;*****/

```

```

; /*****
; * PROCEDURE: FileSum()
; * MODULE:
; * PURPOSE: This prints the Animal Summary record and linked records to a file.
; * CALLED BY: PrintScreen()
; * CALLS:
; * INPUT ASSERTION:
; * OUTPUT ASSERTION:
; * GLOBAL VARIABLES:
; * AUTHOR: Sean K. Smith
; * DATE: 9/26/91
; *****/
PROC FileSum()

    InitPrintVars()

    FStartOfPage()
    FPrintAni()
    FPrintTeffect()
    FPrintTDose()
    FEndOfPage()

    CLEAR
ENDPROC
WRITELIB LIBNAME FileSum
RELEASE PROCS FileSum
; /*****
; * END MODULE: x.x FileSum()
; *****/

```

```

;*****
;* PROCEDURE: PrintAni()
;* MODULE:
;* PURPOSE: This is the routine which formats and prints the ANI record to
;*           the printer.
;* CALLED BY: PrintSum()
;* CALLS:
;* INPUT ASSERTION:
;* OUTPUT ASSERTION:
;* GLOBAL VARIABLES:
;* AUTHOR: Sean K. Smith
;* DATE: 9/26/91
;*****/
PROC PrintAni()
  PRIVATE x,          ;/* number of total lines required on the page */
          i,          ;/* counter of FOR loop */
          table_title, ;/* holds the table title */
          no_chars,    ;/* temp to hold number of pad chars in title */
          cod_lines,   ;/* number of lines COD wrapped onto */
          cod_ln_add,  ;/* number of additional required on page */
          finding_lines, ;/* number of lines FINDINGS wrapped onto */
          find_ln_add ;/* number of additional required on page */

  ;/* word wrap the COD field and return number of lines */
  cod_lines = WordWrap([CAUSE_OF_DEATH], "cod_array")

  ;/* calculate the number of additional lines required on the page */
  IF (cod_lines > 1) THEN
    cod_ln_add = cod_lines - 1
  ELSE
    cod_ln_add = 0
  ENDIF

  ;/* word wrap the FINDINGS field and return the number of lines */
  findings_lines = WordWrap([FINDINGS], "findings_array")

  ;/* calculate the number of additional lines required on the page */
  IF (findings_lines > 1) THEN
    find_ln_add = findings_lines - 1
  ELSE
    find_ln_add = 0
  ENDIF

  x = cod_ln_add + find_ln_add + 30 ;/* lines required to print record */

  AddLineNumber(x + 2) ;/* call to line counter proc */

  table_title = " Animal Summary " ;/* note spaces on either side */
  no_chars = INT((80 - LEN(table_title))/2) ;/* no chars to fill */
  table_title = FILL(" ", no_chars) + table_title + FILL(" ", no_chars)

  PRINT "\n\n" ;/* insert two blank lines */
  PRINT boldon, table_title, boldoff, "\n\n" ;/* print title */

  ;/* print the field of the record on separate lines */
  PRINT boldon, "ANIMAL_ID:      ", boldoff, [ANIMAL_ID], "\n"
  PRINT boldon, "LAB_ID:         ", boldoff, [LAB_ID], gap,
    [S\D\LAB -> LAB_NAME], "\n"
  PRINT boldon, "STUDY_ID:         ", boldoff, [STUDY_ID], gap,
    [S\D\STU -> NRA_STUDY_NAME], "\n"

```

```

PRINT boldon, "GROUP_ID:      ", boldoff, [GROUP_ID], gap,
               [S\D\GRO -> NRA_GROUP_NAME], "\n"
PRINT boldon, "SPECIES:      ", boldoff, [SPECIES], gap,
               [S\V\SPECIES -> TRANS], "\n"
PRINT boldon, "PRIMARY:      ", boldoff, [PRIMARY], "\n"
PRINT boldon, "CONTROL:      ", boldoff, [CONTROL], "\n"
PRINT boldon, "RESULTS_FINAL: ", boldoff, [RESULTS_FINAL], "\n"
PRINT boldon, "ASSIGNED_ID:   ", boldoff, [ASSIGNED_ID], "\n"
PRINT boldon, "SEX:          ", boldoff, [SEX], "\n"
PRINT boldon, "SIRE_ID:       ", boldoff, [SIRE_ID], "\n"
PRINT boldon, "DAM_ID:        ", boldoff, [DAM_ID], "\n"
PRINT boldon, "LITTER_NO:      ", boldoff, [LITTER_NO], "\n"
PRINT boldon, "BIRTH_DATE:     ", boldoff, [BIRTH_DATE], "\n"
PRINT boldon, "BURDEN_TYPE:      ", boldoff, [BURDEN_TYPE], "\n"
PRINT boldon, "QTY_INSULT_SI:   ", boldoff, [QTY_INSULT_SI], "\n"
PRINT boldon, "UNITS_SI:        ", boldoff, [INSULT_UNITS_SI], "\n"
PRINT boldon, "QTY_INSULT_TD:   ", boldoff, [QTY_INSULT_TRADITIONAL], "\n"
PRINT boldon, "UNITS_TRAD:       ", boldoff, [INSULT_UNITS_TRADITIONAL], "\n"
PRINT boldon, "FIRST_INSULT:      ", boldoff, [FIRST_INSULT], "\n"
PRINT boldon, "FINAL_INSULT:      ", boldoff, [FINAL_INSULT], "\n"
PRINT boldon, "FIRST_INS_WGT:     ", boldoff, [WGT_FIRST_INSULT], " grams", "\n"
PRINT boldon, "REMOVAL_DATE:     ", boldoff, [REMOVAL_DATE], "\n"
PRINT boldon, "DEATH_DATE:        ", boldoff, [DEATH_DATE], "\n"
PRINT boldon, "DISPOSITION:     ", boldoff, [DISPOSITION], gap,
               [S\V\DISP -> DESCRIPTION], "\n"
PRINT boldon, "TISS_ANALYSIS:   ", boldoff, [TISS_QUALITY], "\n"

/* wordwrap the cause of death */
PRINT boldon, "COD: ", boldoff
IF ( cod_lines <> 0 ) THEN
    PRINT SPACES(11), cod_array[1], "\n"

    FOR i FROM 2 TO cod_lines
        PRINT SPACES(16), cod_array[i], "\n"
    ENDFOR
    RELEASE VARS cod_array /* release the array */
ELSE
    PRINT "\n"
ENDIF

/* wordwrap the findings */
PRINT boldon, "FINDINGS: ", boldoff
IF ( findings_lines <> 0 ) THEN
    PRINT SPACES(6), findings_array[1], "\n"

    FOR i FROM 2 TO findings_lines
        PRINT SPACES(16), findings_array[i], "\n"
    ENDFOR
    RELEASE VARS findings_array /* release the array */
ELSE
    PRINT "\n"
ENDIF

PRINT boldon, "ACCESSION_DATE: ", boldoff, [ACCESSION_DATE], "\n"

ENDPROC
WRITELIB LIBNAME PrintAni
RELEASE PROCS PrintAni
;*****
;* END MODULE: x.x PrintAni()
;*****/

```

```

; /*****
; * PROCEDURE: FPrintAni()
; * MODULE:
; * PURPOSE: Prints the ANI record to a File
; * CALLED BY: FileSum()
; * CALLS:
; * INPUT ASSERTION:
; * OUTPUT ASSERTION:
; * GLOBAL VARIABLES:
; * AUTHOR: Sean K. Smith
; * DATE: 9/26/91
; *****/
PROC FPrintAni()
  PRIVATE x,          /* number of total lines required on the page */
         i,          /* counter of FOR loop */
         table_title, /* holds the table title */
         no_chars,    /* temp to hold number of pad chars in title */
         cod_lines,    /* number of lines COD wrapped onto */
         cod_ln_add,   /* number of additional required on page */
         findings_lines, /* number of lines FINDINGS wrapped onto */
         find_ln_add   /* number of additional required on page */

  /* word wrap the COD field and return number of lines */
  cod_lines = WordWrap([CAUSE_OF_DEATH], "cod_array")

  /* calculate the number of additional lines required on the page */
  IF (cod_lines > 1) THEN
    cod_ln_add = cod_lines - 1
  ELSE
    cod_ln_add = 0
  ENDIF

  /* word wrap the FINDINGS field and return the number of lines */
  findings_lines = WordWrap([FINDINGS], "findings_array")

  /* calculate the number of additional lines required on the page */
  IF (findings_lines > 1) THEN
    find_ln_add = findings_lines - 1
  ELSE
    find_ln_add = 0
  ENDIF

  x = cod_ln_add + find_ln_add + 30 /* lines required to print record */

  FAddLineNumber(x + 2) /* call to line counter proc */

  table_title = " Animal Summary " /* note spaces on either side */
  no_chars = INT((80 - LEN(table_title))/2) /* no chars to fill */
  table_title = FILL(" ", no_chars) + table_title + FILL(" ", no_chars)

  PRINT FILE FileNam "\n\n" /* insert two blank lines */
  PRINT FILE FileNam boldon, table_title, boldoff, "\n\n" /* print title */

  /* print the field of the record on separate lines */
  PRINT FILE FileNam boldon, "ANIMAL_ID:      ", boldoff, [ANIMAL_ID], "\n"
  PRINT FILE FileNam boldon, "LAB_ID:        ", boldoff, [LAB_ID], gap,
    [S\D\LAB -> LAB_NAME], "\n"
  PRINT FILE FileNam boldon, "STUDY_ID:      ", boldoff, [STUDY_ID], gap,
    [S\D\STU -> NRA_STUDY_NAME], "\n"
  PRINT FILE FileNam boldon, "GROUP_ID:      ", boldoff, [GROUP_ID], gap,

```

```

        [S\D\GRO -> NRA_GROUP_NAME], "\n"
PRINT FILE FileNam boldon, "SPECIES: " , boldoff, [SPECIES], gap,
        [S\V\SPECIES -> TRANS], "\n"
PRINT FILE FileNam boldon, "PRIMARY: " , boldoff, [PRIMARY], "\n"
PRINT FILE FileNam boldon, "CONTROL: " , boldoff, [CONTROL], "\n"
PRINT FILE FileNam boldon, "RESULTS_FINAL: " , boldoff, [RESULTS_FINAL], "\n"
PRINT FILE FileNam boldon, "ASSIGNED_ID: " , boldoff, [ASSIGNED_ID], "\n"
PRINT FILE FileNam boldon, "SEX: " , boldoff, [SEX], "\n"
PRINT FILE FileNam boldon, "SIRE_ID: " , boldoff, [SIRE_ID], "\n"
PRINT FILE FileNam boldon, "DAM_ID: " , boldoff, [DAM_ID], "\n"
PRINT FILE FileNam boldon, "LITTER_NO: " , boldoff, [LITTER_NO], "\n"
PRINT FILE FileNam boldon, "BIRTH_DATE: " , boldoff, [BIRTH_DATE], "\n"
PRINT FILE FileNam boldon, "BURDEN_TYPE: " , boldoff, [BURDEN_TYPE], "\n"
PRINT FILE FileNam boldon, "QTY_INSULT_SI: " , boldoff, [QTY_INSULT_SI], "\n"
PRINT FILE FileNam boldon, "UNITS_SI: " , boldoff, [INSULT_UNITS_SI], "\n"
PRINT FILE FileNam boldon, "QTY_INSULT_TD: " , boldoff, [QTY_INSULT_TRADITIONAL], "\n"
PRINT FILE FileNam boldon, "UNITS_TRAD: " , boldoff, [INSULT_UNITS_TRADITIONAL], "\n"
PRINT FILE FileNam boldon, "FIRST_INSULT: " , boldoff, [FIRST_INSULT], "\n"
PRINT FILE FileNam boldon, "FINAL_INSULT: " , boldoff, [FINAL_INSULT], "\n"
PRINT FILE FileNam boldon, "FIRST_INS_WGT: " , boldoff, [WGT_FIRST_INSULT], " grams", "\n"
PRINT FILE FileNam boldon, "REMOVAL_DATE: " , boldoff, [REMOVAL_DATE], "\n"
PRINT FILE FileNam boldon, "DEATH_DATE: " , boldoff, [DEATH_DATE], "\n"
PRINT FILE FileNam boldon, "DISPOSITION: " , boldoff, [DISPOSITION], gap,
        [S\V\DISP -> DESCRIPTION], "\n"
PRINT FILE FileNam boldon, "TISS_ANALYSIS: " , boldoff, [TISS_QUALITY], "\n"

/* wordwrap the cause of death */
PRINT FILE FileNam boldon, "COD: " , boldoff
IF ( cod_lines <> 0 ) THEN
    PRINT FILE FileNam SPACES(11), cod_array[1], "\n"

    FOR i FROM 2 TO cod_lines
        PRINT FILE FileNam SPACES(16), cod_array[i], "\n"
    ENDFOR
    RELEASE VARS cod_array /* release the array */
ELSE
    PRINT FILE FileNam "\n"
ENDIF

/* wordwrap the findings */
PRINT FILE FileNam boldon, "FINDINGS: " , boldoff
IF ( findings_lines <> 0 ) THEN
    PRINT FILE FileNam SPACES(6), findings_array[1], "\n"

    FOR i FROM 2 TO findings_lines
        PRINT FILE FileNam SPACES(16), findings_array[i], "\n"
    ENDFOR
    RELEASE VARS findings_array /* release the array */
ELSE
    PRINT FILE FileNam "\n"
ENDIF

PRINT FILE FileNam boldon, "ACCESSION_DATE: " , boldoff, [ACCESSION_DATE], "\n"

ENDPROC
WRITELIB LIBNAME FPrintAni
RELEASE PROCS FPrintAni
;*****
/* END MODULE: x.x FPrintAni()
;*****

```



```

;*****
;* PROCEDURE: PrintTeffect()
;* MODULE:
;* PURPOSE: Prints the Teffect records which are currently linked to the
;*          ANI table.
;* CALLED BY: PrintSum()
;* CALLS:
;* INPUT ASSERTION:
;* OUTPUT ASSERTION:
;* GLOBAL VARIABLES:
;* AUTHOR: Sean K. Smith
;* DATE: 9/26/91
;*****/
PROC PrintTeffect()
PRIVATE numofdetail, /* count of detail records */
       table_title, /* table title */
       no_chars, /* number of chars to pad title */
       i /* counter of FOR loop */

/* prepare title line */
table_title = " Effects Summary for Animal Table " /* note spaces on either side */
no_chars = INT((80 - LEN(table_title))/2)
table_title = FILL(" ",no_chars) + table_title + FILL(" ",no_chars)

MOVETO "W\\TEFFECT1"
numofdetail = NIMAGERECORDS()

/* no detail notify */
IF numofdetail = 0 THEN
  AddLineNumber(8)
  PRINT "\n\n"
  PRINT boldon,table_title,boldoff,"\n\n"
  PRINT boldon,
    FORMAT("w10","TOPO_CODE"),gap,
    FORMAT("w15","TOPOGRAPHY"),gap,
    FORMAT("w10","MORPH_CODE"),gap,
    FORMAT("w16","MORPHOLOGY"),gap,
    FORMAT("w3","DX"),"\n"
  PRINT FILL("-",10),gap,
    FILL("-",15),gap,
    FILL("-",10),gap,
    FILL("-",16),gap,
    FILL("-",3),boldoff,"\n"
  PRINT "Nothing to report"
  RETURN
/* details records exist */
ELSE
  AddLineNumber(7)
  PRINT "\n\n"
  PRINT boldon,table_title,boldoff,"\n\n"
  /* print table header */
  PRINT boldon,
    FORMAT("w10","TOPO_CODE"),gap,
    FORMAT("w15","TOPOGRAPHY"),gap,
    FORMAT("w10","MORPH_CODE"),gap,
    FORMAT("w16","MORPHOLOGY"),gap,
    FORMAT("w3","DX"),"\n"
  PRINT FILL("-",10),gap,
    FILL("-",15),gap,
    FILL("-",10),gap,

```

```

        FILL("-",16),gap,
        FILL("-",3),boldoff,"\n"
; /* scan the multirecord form and test to see if new page started */
SCAN
    i = AddLineNumber(1) ; /* returns 1 if new page started */

; /* check to see if a new page has started */
; /* print table header if so */
IF (i = 1) THEN
    AddLineNumber(7)
    PRINT "\n\n"
    PRINT boldon,table_title,boldoff,"\n\n"

    PRINT boldon,
        FORMAT("w10","TOPO_CODE"),gap,
        FORMAT("w15","TOPOGRAPHY"),gap,
        FORMAT("w10","MORPH_CODE"),gap,
        FORMAT("w16","MORPHOLOGY"),gap,
        FORMAT("w3","DX"),"\n"
    PRINT FILL("-",10),gap,
        FILL("-",15),gap,
        FILL("-",10),gap,
        FILL("-",16),gap,
        FILL("-",3),boldoff,"\n"
ENDIF

; /* print out the fields */
PRINT FORMAT("w10",[TISSUE_CATEGORY]),gap,
        FORMAT("w15",[TISSUE_TRANS]),gap,
        FORMAT("w10",[MORP_CATEGORY]),gap,
        FORMAT("w16",[MORP_TRANS]),gap,
        FORMAT("w3",[DX]),"\n"

ENDSCAN

ENDIF

ENDPROC
WRITELIB LIBNAME PrintTeffect
RELEASE PROCS PrintTeffect
; /* *****
; * END MODULE: x.x PrintTeffect
; * ***** /

```

```

;*****
;* PROCEDURE: FPrintTeffect()
;* MODULE:
;* PURPOSE: Prints the TEeffects records to a file.
;* CALLED BY: FileSum()
;* CALLS:
;* INPUT ASSERTION:
;* OUTPUT ASSERTION:
;* GLOBAL VARIABLES:
;* AUTHOR: Sean K. Smith
;* DATE: 9/26/91
;*****/
PROC FPrintTeffect()

PRIVATE numofdetail, /* count of detail records */
       table_title, /* table title */
       no_chars, /* number of chars to pad title */
       i /* counter of FOR loop */

/* prepare title line */
table_title = " Effects Summary for Animal Table " /* note spaces on either side */
no_chars = INT((80 - LEN(table_title))/2)
table_title = FILL(" ",no_chars) + table_title + FILL(" ",no_chars)

MOVETO "W\\TEFFECT1"
numofdetail = NIMAGERECORDS()

/* no detail notify */
IF numofdetail = 0 THEN
  FAddLineNumber(8)
  PRINT FILE FileNam "\n\n"
  PRINT FILE FileNam boldon,table_title,boldoff,"\n\n"
  PRINT FILE FileNam boldon,
    FORMAT("w10","TOPO_CODE"),gap,
    FORMAT("w15","TOPOGRAPHY"),gap,
    FORMAT("w10","MORPH_CODE"),gap,
    FORMAT("w16","MORPHOLOGY"),gap,
    FORMAT("w3","DX"),"\n"
  PRINT FILE FileNam FILL("-",10),gap,
    FILL("-",15),gap,
    FILL("-",10),gap,
    FILL("-",16),gap,
    FILL("-",3),boldoff,"\n"
  PRINT FILE FileNam "Nothing to report"
  RETURN
/* details records exist */
ELSE
  FAddLineNumber(7)
  PRINT FILE FileNam "\n\n"
  PRINT FILE FileNam boldon,table_title,boldoff,"\n\n"
  /* print table header */
  PRINT FILE FileNam boldon,
    FORMAT("w10","TOPO_CODE"),gap,
    FORMAT("w15","TOPOGRAPHY"),gap,
    FORMAT("w10","MORPH_CODE"),gap,
    FORMAT("w16","MORPHOLOGY"),gap,
    FORMAT("w3","DX"),"\n"
  PRINT FILE FileNam FILL("-",10),gap,
    FILL("-",15),gap,
    FILL("-",10),gap,

```

```

        FILL("-",16),gap,
        FILL("-",3),boldoff,"\n"
; /* scan the multirecord form and test to see if new page started */
SCAN
    i = FAddLineNumber(1) ; /* returns 1 if new page started */

; /* check to see if a new page has started */
; /* print table header if so */
IF (i = 1) THEN
    FAddLineNumber(7)
    PRINT FILE FileNam "\n\n"
    PRINT FILE FileNam boldon,table_title,boldoff,"\n\n"

    PRINT FILE FileNam boldon,
        FORMAT("w10","TOPO_CODE"),gap,
        FORMAT("w15","TOPOGRAPHY"),gap,
        FORMAT("w10","MORPH_CODE"),gap,
        FORMAT("w16","MORPHOLOGY"),gap,
        FORMAT("w3","DX"),"\n"
    PRINT FILE FileNam FILL("-",10),gap,
        FILL("-",15),gap,
        FILL("-",10),gap,
        FILL("-",16),gap,
        FILL("-",3),boldoff,"\n"
ENDIF

; /* print out the fields */
PRINT FILE FileNam FORMAT("w10",[TISSUE_CATEGORY]),gap,
    FORMAT("w15",[TISSUE_TRANS]),gap,
    FORMAT("w10",[MORP_CATEGORY]),gap,
    FORMAT("w16",[MORP_TRANS]),gap,
    FORMAT("w3",[DX]),"\n"

ENDSCAN

ENDIF

ENDPROC
WRITELIB LIBNAME FPrintTeffect
RELEASE PROCS FPrintTeffect
; /* *****
; * END MODULE: x.x FPrintTeffect
; ***** */

```

```

;*****
;* PROCEDURE: PrintTDose()
;* MODULE:
;* PURPOSE: Prints the TDose records linked to the ANI table.
;* CALLED BY:
;* CALLS: PrintSum()
;* INPUT ASSERTION:
;* OUTPUT ASSERTION:
;* GLOBAL VARIABLES:
;* AUTHOR: Sean K. Smith
;* DATE: 9/26/91
;*****/
PROC PrintTDose()
    PRIVATE numofdetail, /* the number of detail records */
           table_title, /* the table title */
           no_chars, /* number of chars required to pad table */
           i /* counter of FOR loop */

    table_title = " Tissue Dose Summary for Animal Table " /* note spaces on either side */
    no_chars = INT((80 - LEN(table_title))/2)
    table_title = FILL(" ",no_chars) + table_title + FILL(" ",no_chars)

    MOVETO "W\\TDOSE1"
    numofdetail = NIMAGERECORDS()

    /* test for existence of detail records notify if empty */
    IF numofdetail = 0 THEN
        AddLineNumber(5) /* call to line count proc */
        PRINT "\n\n" /* space two lines */
        PRINT boldon,table_title,boldoff,"\n\n"
        PRINT "Nothing to report"
        RETURN
    /* detail records exists print fields */
    ELSE
        AddLineNumber(4)
        PRINT "\n\n"
        PRINT boldon,table_title,boldoff,"\n\n"

        /* scan the detail table printing each linked record */
        SCAN
            i = AddLineNumber(6) /* 6 lines required for each record */

            /* check to see if a new page started */
            IF (i = 1) THEN
                AddLineNumber(10) /* include table header and lines */
                PRINT "\n\n"
                PRINT boldon,table_title,boldoff,"\n\n"
            ENDIF

            /* print out the fields in a free form style */
            PRINT boldon, "TISSUE_CATEGORY: ", boldoff, [TISSUE_TRANS], "\n"
            PRINT boldon, "DOSE_UNITS: ", boldoff, [DOSE_UNITS], "\n"

            PRINT boldon, "LAST_NORMAL: ", boldoff,
                FORMAT("D1,W8",[LAST_NORMAL]), gap
            PRINT boldon, "TYPE_LAST_NORMAL: ", boldoff,
                FORMAT("W1",[TYPE_LAST_NORMAL]), gap
            PRINT boldon, "DOSE_LAST_NORMAL: ", boldoff,
                FORMAT("W5.2,AR",[DOSE_LAST_NORMAL]), gap
            PRINT boldon, "DOSE_ERR_LAST_NORMAL: ", boldoff,
                FORMAT("W5.2,AR",[DOSE_ERR_LAST_NORMAL]), "\n"

```

```

PRINT boldon, "DATE_CLIN_DX: ", boldoff,
  FORMAT("D1,W8",[DATE_CLIN_DX]), gap
PRINT boldon, "TYPE_CLIN_DX   : ", boldoff,
  FORMAT("W1",[TYPE_CLIN_DX]), gap
PRINT boldon, "DOSE_CLIN_DX   : ", boldoff,
  FORMAT("W5.2,AR",[DOSE_CLIN_DX]), gap
PRINT boldon, "DOSE_ERR_CLIN_DX : ", boldoff,
  FORMAT("W5.2,AR",[DOSE_ERR_CLIN_DX]), "\n"

PRINT boldon, "DATE_TISS_DX: ", boldoff,
  FORMAT("D1,W8",[DATE_TISS_DX]), gap
PRINT boldon, "TYPE_TISS_DX   : ", boldoff,
  FORMAT("W1",[TYPE_TISS_DX]), gap
PRINT boldon, "DOSE_TISS_DX   : ", boldoff,
  FORMAT("W5.2,AR",[DOSE_TISS_DX]), gap
PRINT boldon, "DOSE_ERR_TISS_DX : ", boldoff,
  FORMAT("W5.2,AR",[DOSE_ERR_TISS_DX]), "\n\n"

ENDSCAN
ENDIF
ENDPROC
WRITELIB LIBNAME PrintTDose
RELEASE PROCS PrintTDose
;*****
;* END MODULE: x.x PrintTDose
;*****

```

```

;*****
;* PROCEDURE: FPrintTDose()
;* MODULE:
;* PURPOSE: Print the TDose records linked to the ANI table.
;* CALLED BY: FileSum()
;* CALLS:
;* INPUT ASSERTION:
;* OUTPUT ASSERTION:
;* GLOBAL VARIABLES:
;* AUTHOR: Sean K. Smith
;* DATE: 9/26/91
;*****/
PROC FPrintTDose()
    PRIVATE numofdetail, /* the number of detail records */
           table_title, /* the table title */
           no_chars, /* number of chars required to pad table */
           i /* counter of FOR loop */

    table_title = " Tissue Dose Summary for Animal Table " /* note spaces on either side */
    no_chars = INT((80 - LEN(table_title))/2)
    table_title = FILL(" ",no_chars) + table_title + FILL(" ",no_chars)

    MOVETO "W\\TDOSE1"
    numofdetail = NIMAGERECORDS()

    /* test for existence of detail records notify if empty */
    IF numofdetail = 0 THEN
        FAddLineNumber(5) /* call to line count proc */
        PRINT FILE FileNam "\n\n" /* space two lines */
        PRINT FILE FileNam boldon,table_title,boldoff,"\n\n"
        PRINT FILE FileNam "Nothing to report"
        RETURN
    /* detail records exists print fields */
    ELSE
        FAddLineNumber(4)
        PRINT FILE FileNam "\n\n"
        PRINT FILE FileNam boldon,table_title,boldoff,"\n\n"

        /* scan the detail table printing each linked record */
        SCAN
            i = FAddLineNumber(6) /* 6 lines required for each record */

            /* check to see if a new page started */
            IF (i = 1) THEN
                FAddLineNumber(10) /* include table header and lines */
                PRINT FILE FileNam "\n\n"
                PRINT FILE FileNam boldon,table_title,boldoff,"\n\n"
            ENDIF

            /* print out the fields in a free form style */
            PRINT FILE FileNam boldon, "TISSUE_CATEGORY: ", boldoff, [TISSUE_TRANS], "\n"
            PRINT FILE FileNam boldon, "DOSE_UNITS: ", boldoff, [DOSE_UNITS], "\n"

            PRINT FILE FileNam boldon, "LAST_NORMAL: ", boldoff,
                FORMAT("D1,W8",[LAST_NORMAL]), gap
            PRINT FILE FileNam boldon, "TYPE_LAST_NORMAL: ", boldoff,
                FORMAT("W1",[TYPE_LAST_NORMAL]), gap
            PRINT FILE FileNam boldon, "DOSE_LAST_NORMAL: ", boldoff,
                FORMAT("W5.2,AR",[DOSE_LAST_NORMAL]), gap
            PRINT FILE FileNam boldon, "DOSE_ERR_LAST_NORMAL: ", boldoff,
                FORMAT("W5.2,AR",[DOSE_ERR_LAST_NORMAL]), "\n"

```

```

PRINT FILE FileNam boldon, "DATE_CLIN_DX: ", boldoff,
    FORMAT("D1,W8",[DATE_CLIN_DX]), gap
PRINT FILE FileNam boldon, "TYPE_CLIN_DX   : ", boldoff,
    FORMAT("W1",[TYPE_CLIN_DX]), gap
PRINT FILE FileNam boldon, "DOSE_CLIN_DX   : ", boldoff,
    FORMAT("W5.2,AR",[DOSE_CLIN_DX]), gap
PRINT FILE FileNam boldon, "DOSE_ERR_CLIN_DX : ", boldoff,
    FORMAT("W5.2,AR",[DOSE_ERR_CLIN_DX]), "\n"

PRINT FILE FileNam boldon, "DATE_TISS_DX: ", boldoff,
    FORMAT("D1,W8",[DATE_TISS_DX]), gap
PRINT FILE FileNam boldon, "TYPE_TISS_DX   : ", boldoff,
    FORMAT("W1",[TYPE_TISS_DX]), gap
PRINT FILE FileNam boldon, "DOSE_TISS_DX   : ", boldoff,
    FORMAT("W5.2,AR",[DOSE_TISS_DX]), gap
PRINT FILE FileNam boldon, "DOSE_ERR_TISS_DX : ", boldoff,
    FORMAT("W5.2,AR",[DOSE_ERR_TISS_DX]), "\n\n"

```

```

ENDSCAN
ENDIF

```

```

ENDPROC
WRITELIB LIBNAME FPrintTDose
RELEASE PROCS FPrintTDose
; /*****
; * END MODULE: x.x FPrintTDose
; *****/

```

```

;*****
;* PROCEDURE: PrintInv()
;* MODULE:
;* PURPOSE: Print the Inventory Form Detail record and linked detail table
;*          records as defined by the current record.
;* CALLED BY: PrintScreen()
;* CALLS:
;* INPUT ASSERTION:
;* OUTPUT ASSERTION:
;* GLOBAL VARIABLES:
;* AUTHOR: Sean K. Smith
;* DATE: 9/26/91
;*****/
PROC PrintInv()

    InitPrintVars()

    OPEN PRINTER

    StartOfPage()
    PrintDetail() /* print the Inventory Detail record which owns the form */
    PrintIAni()   /* print the animal record associated with this detail record */
    PrintMaster() /* print the master inventory record associated with this form */
    PrintLocation()/* print the location record associated with this form */
    EndOfPage()

    CLOSE PRINTER

    CLEAR
ENDPROC
WRITELIB LIBNAME PrintInv
RELEASE PROCS PrintInv
;*****
;* END MODULE: x.x PrintInv()
;*****/

```

```

; /*****
; * PROCEDURE: FileInv()
; * MODULE:
; * PURPOSE: Print the Inventory Detail record and the linked table records of
; *           the current form.
; * CALLED BY: PrintScreen()
; * CALLS:
; * INPUT ASSERTION:
; * OUTPUT ASSERTION:
; * GLOBAL VARIABLES:
; * AUTHOR: Sean K. Smith
; * DATE: 9/26/91
; *****/
PROC FileInv()

    InitPrintVars()

    FStartOfPage()
    FPrintDetail()
    FPrintIAI()
    FPrintMaster()
    FPrintLocation()
    FEndOfPage()

    CLEAR

ENDPROC
WRTLIB LIBNAME FileInv
RELEASE PROCS FileInv
; /*****
; * END MODULE: x.x FileInv()
; *****/

```

```

;*****
;* PROCEDURE: PrintDetail()
;* MODULE:
;* PURPOSE: Prints the Inventory detail record of this form which is the
;*           owner of the current form.
;* CALLED BY: PrintInv()
;* CALLS:
;* INPUT ASSERTION:
;* OUTPUT ASSERTION:
;* GLOBAL VARIABLES:
;* AUTHOR: Sean K. Smith
;* DATE: 9/26/91
;*****/
PROC PrintDetail()
    PRIVATE table_title, /* title of table */
           no_chars      /* number of chars required to pad the title */

    AddLineNumber(5) /* number of lines required to print table record */
    PRINT "\n\n"      /* print two blank lines */

    table_title = " Inventory Detail Table " /* note spaces on either side */
    no_chars = INT((80 - LEN(table_title))/2)
    table_title = FILL(" ",no_chars) + table_title + FILL(" ",no_chars)

    PRINT boldon,table_title,boldoff,"\n\n"

    /* print the fields of the table */
    PRINT boldon, "ANIMAL_ID: ", boldoff,[ANIMAL_ID], gap
    PRINT boldon, "QUALIFIER: ", boldoff,[QUALIFIER], gap
    PRINT boldon, "QUANTITY: ", boldoff,[QUANTITY], gap
    PRINT boldon, "BAR_CODE: ", boldoff, [BAR_CODE], "\n"

ENDPROC
WRITELIB LIBNAME PrintDetail
RELEASE PROCS PrintDetail
;*****
;* END MODULE: x.x PrintDetail
;*****/

```

```

;*****
;* PROCEDURE: FPrintDetail()
;* MODULE:
;* PURPOSE: Print the Inventory Detail record to a file.
;* CALLED BY: FileInv()
;* CALLS:
;* INPUT ASSERTION:
;* OUTPUT ASSERTION:
;* GLOBAL VARIABLES:
;* AUTHOR: Sean K. Smith
;* DATE: 9/26/91
;*****/
PROC FPrintDetail()
    PRIVATE table_title, /* title of table */
           no_chars      /* number of chars required to pad the title */

    FAddLineNumber(5) /* number of lines required to print table record */
    PRINT FILE FileNam "\n\n" /* print two blank lines */

    table_title = " Inventory Detail Table " /* note spaces on either side */
    no_chars = INT((80 - LEN(table_title))/2)
    table_title = FILL(" ",no_chars) + table_title + FILL(" ",no_chars)

    PRINT FILE FileNam boldon,table_title,boldoff,"\n\n"

    /* print the fields of the table */
    PRINT FILE FileNam boldon, "ANIMAL_ID: ", boldoff,[ANIMAL_ID], gap
    PRINT FILE FileNam boldon, "QUALIFIER: ", boldoff,[QUALIFIER], gap
    PRINT FILE FileNam boldon, "QUANTITY: ", boldoff,[QUANTITY], gap
    PRINT FILE FileNam boldon, "BAR_CODE: ", boldoff, [BAR_CODE], "\n"

ENDPROC
WRITELIB LIBNAME FPrintDetail
RELEASE PROCS FPrintDetail
;*****
;* END MODULE: x.x FPrintDetail
;*****/

```

```

;*****
;* PROCEDURE: PrintANI()
;* MODULE:
;* PURPOSE: Prints the ANI record linked to the Inventory record
;* CALLED BY: PrintInv()
;* CALLS:
;* INPUT ASSERTION:
;* OUTPUT ASSERTION:
;* GLOBAL VARIABLES:
;* AUTHOR: Sean K. Smith
;* DATE: 9/26/91
;*****/
PROC PrintANI()
    PRIVATE x,          /* number of total lines required on the page */
            i,          /* counter of FOR loop */
            table_title, /* holds the table title */
            no_chars,    /* temp to hold number of pad chars in title */
            cod_lines,   /* number of lines COD wrapped onto */
            cod_ln_add,  /* number of additional required on page */
            finding_lines, /* number of lines FINDINGS wrapped onto */
            find_ln_add  /* number of additional required on page */

    MOVETO "S\D\ANI"

    /* word wrap the COD field and return number of lines */
    cod_lines = WordWrap([CAUSE_OF_DEATH],"cod_array")

    /* calculate the number of additional lines required on the page */
    IF (cod_lines > 1) THEN
        cod_ln_add = cod_lines - 1
    ELSE
        cod_ln_add = 0
    ENDIF

    /* word wrap the FINDINGS field and return the number of lines */
    findings_lines = WordWrap([FINDINGS], "findings_array")

    /* calculate the number of additional lines required on the page */
    IF (findings_lines > 1) THEN
        find_ln_add = findings_lines - 1
    ELSE
        find_ln_add = 0
    ENDIF

    x = cod_ln_add + find_ln_add + 30 /* lines required to print record */

    AddLineNumber(x + 2) /* call to line counter proc */

    PRINT "\n\n"

    /* create and print the title */
    table_title = " Animal Summary " /* note spaces on either side */
    no_chars = INT((80 - LEN(table_title))/2)
    table_title = FILL(" ",no_chars) + table_title + FILL(" ",no_chars)
    PRINT boldon,table_title,boldoff,"\n\n"

    /* print the fields */
    PRINT boldon, "ANIMAL_ID:      ",boldoff,[ANIMAL_ID]," \n"
    PRINT boldon, "LAB_ID:          ",boldoff,[LAB_ID]," \n"

```

```

PRINT boldon, "STUDY_ID:      ", boldoff, [STUDY_ID], "\n"
PRINT boldon, "GROUP_ID:     ", boldoff, [GROUP_ID], "\n"
PRINT boldon, "SPECIES:       ", boldoff, [SPECIES], "\n"
PRINT boldon, "PRIMARY:        ", boldoff, [PRIMARY], "\n"
PRINT boldon, "CONTROL:         ", boldoff, [CONTROL], "\n"
PRINT boldon, "RESULTS_FINAL:   ", boldoff, [RESULTS_FINAL], "\n"
PRINT boldon, "ASSIGNED_ID:     ", boldoff, [ASSIGNED_ID], "\n"
PRINT boldon, "SEX:            ", boldoff, [SEX], "\n"
PRINT boldon, "SIRE_ID:         ", boldoff, [SIRE_ID], "\n"
PRINT boldon, "DAM_ID:          ", boldoff, [DAM_ID], "\n"
PRINT boldon, "LITTER_NO:       ", boldoff, [LITTER_NO], "\n"
PRINT boldon, "BIRTH_DATE:      ", boldoff, [BIRTH_DATE], "\n"
PRINT boldon, "BURDEN_TYPE:      ", boldoff, [BURDEN_TYPE], "\n"
PRINT boldon, "QTY_INSULT_SI:    ", boldoff, [QTY_INSULT_SI], "\n"
PRINT boldon, "UNITS_SI:         ", boldoff, [INSULT_UNITS_SI], "\n"
PRINT boldon, "QTY_INSULT_TD:    ", boldoff, [QTY_INSULT_TRADITIONAL], "\n"
PRINT boldon, "UNITS_TRAD:       ", boldoff, [INSULT_UNITS_TRADITIONAL], "\n"
PRINT boldon, "FIRST_INSULT:     ", boldoff, [FIRST_INSULT], "\n"
PRINT boldon, "FINAL_INSULT:     ", boldoff, [FINAL_INSULT], "\n"
PRINT boldon, "FIRST_INS_WGT:    ", boldoff, [WGT_FIRST_INSULT], "grams", "\n"
PRINT boldon, "REMOVAL_DATE:     ", boldoff, [REMOVAL_DATE], "\n"
PRINT boldon, "DEATH_DATE:       ", boldoff, [DEATH_DATE], "\n"
PRINT boldon, "DISPOSITION:      ", boldoff, [DISPOSITION], "\n"
PRINT boldon, "TISS_ANALYSIS:    ", boldoff, [TISS_QUALITY], "\n"

/* wordwrap the cause of death */
PRINT boldon, "COD: ", boldoff
IF ( cod_lines <> 0 ) THEN
  PRINT SPACES(11), cod_array[1], "\n"

  FOR i FROM 2 TO cod_lines
    PRINT SPACES(16), cod_array[i], "\n"
  ENDFOR
  RELEASE VARS cod_array
ELSE
  PRINT "\n"
ENDIF

/* wordwrap the findings */
PRINT boldon, "FINDINGS: ", boldoff
IF ( findings_lines <> 0 ) THEN
  PRINT SPACES(6), findings_array[1], "\n"

  FOR i FROM 2 TO findings_lines
    PRINT SPACES(16), findings_array[i], "\n"
  ENDFOR
  RELEASE VARS findings_array
ELSE
  PRINT "\n"
ENDIF

PRINT boldon, "ACCESSION_DATE: ", boldoff, [ACCESSION_DATE], "\n"

ENDPROC
WRITELIB LIBNAME PrintIani
RELEASE PROCS PrintIani
;*****
/* END MODULE: x.x PrintIani
;*****/

```

```

;*****
;* PROCEDURE: FPrintANI()
;* MODULE:
;* PURPOSE: Prints the ANI table records linked to the Inventory Detail
;*           through the ANIMAL_ID
;* CALLED BY: FileInv()
;* CALLS:
;* INPUT ASSERTION:
;* OUTPUT ASSERTION:
;* GLOBAL VARIABLES:
;* AUTHOR: Sean K. Smith
;* DATE: 9/26/91
;*****/
PROC FPrintANI()
    PRIVATE x,          /* number of total lines required on the page */
            i,          /* counter of FOR loop */
            table_title, /* holds the table title */
            no_chars,    /* temp to hold number of pad chars in title */
            cod_lines,   /* number of lines COD wrapped onto */
            cod_ln_add,  /* number of additional required on page */
            findings_lines, /* number of lines FINDINGS wrapped onto */
            find_ln_add  /* number of additional required on page */

    MOVETO "S\D\ANI"

    /* word wrap the COD field and return number of lines */
    cod_lines = WordWrap([CAUSE_OF_DEATH],"cod_array")

    /* calculate the number of additional lines required on the page */
    IF (cod_lines > 1) THEN
        cod_ln_add = cod_lines - 1
    ELSE
        cod_ln_add = 0
    ENDIF

    /* word wrap the FINDINGS field and return the number of lines */
    findings_lines = WordWrap([FINDINGS], "findings_array")

    /* calculate the number of additional lines required on the page */
    IF (findings_lines > 1) THEN
        find_ln_add = findings_lines - 1
    ELSE
        find_ln_add = 0
    ENDIF

    x = cod_ln_add + find_ln_add + 30 /* lines required to print record */

    FAddLineNumber(x + 2) /* call to line counter proc */

    PRINT FILE FileNam "\n\n"

    /* create and print the title */
    table_title = " Animal Summary " /* note spaces on either side */
    no_chars = INT((80 - LEN(table_title))/2)
    table_title = FILL(" ",no_chars) + table_title + FILL(" ",no_chars)
    PRINT FILE FileNam boldon,table_title,boldoff,"\n\n"

    /* print the fields */
    PRINT FILE FileNam boldon, "ANIMAL_ID:      ",boldoff,[ANIMAL_ID]," \n"

```

```

PRINT FILE FileNam boldon, "LAB_ID:           ", boldoff, [LAB_ID], "\n"
PRINT FILE FileNam boldon, "STUDY_ID:          ", boldoff, [STUDY_ID], "\n"
PRINT FILE FileNam boldon, "GROUP_ID:           ", boldoff, [GROUP_ID], "\n"
PRINT FILE FileNam boldon, "SPECIES:            ", boldoff, [SPECIES], "\n"
PRINT FILE FileNam boldon, "PRIMARY:             ", boldoff, [PRIMARY], "\n"
PRINT FILE FileNam boldon, "CONTROL:            ", boldoff, [CONTROL], "\n"
PRINT FILE FileNam boldon, "RESULTS_FINAL:      ", boldoff, [RESULTS_FINAL], "\n"
PRINT FILE FileNam boldon, "ASSIGNED_ID:        ", boldoff, [ASSIGNED_ID], "\n"
PRINT FILE FileNam boldon, "SEX:                 ", boldoff, [SEX], "\n"
PRINT FILE FileNam boldon, "SIRE_ID:             ", boldoff, [SIRE_ID], "\n"
PRINT FILE FileNam boldon, "DAM_ID:              ", boldoff, [DAM_ID], "\n"
PRINT FILE FileNam boldon, "LITTER_NO:          ", boldoff, [LITTER_NO], "\n"
PRINT FILE FileNam boldon, "BIRTH_DATE:         ", boldoff, [BIRTH_DATE], "\n"
PRINT FILE FileNam boldon, "BURDEN_TYPE:        ", boldoff, [BURDEN_TYPE], "\n"
PRINT FILE FileNam boldon, "QTY_INSULT_SI:      ", boldoff, [QTY_INSULT_SI], "\n"
PRINT FILE FileNam boldon, "UNITS_SI:           ", boldoff, [INSULT_UNITS_SI], "\n"
PRINT FILE FileNam boldon, "QTY_INSULT_TD:      ", boldoff, [QTY_INSULT_TRADITIONAL], "\n"
PRINT FILE FileNam boldon, "UNITS_TRAD:         ", boldoff, [INSULT_UNITS_TRADITIONAL], "\n"
PRINT FILE FileNam boldon, "FIRST_INSULT:       ", boldoff, [FIRST_INSULT], "\n"
PRINT FILE FileNam boldon, "FINAL_INSULT:       ", boldoff, [FINAL_INSULT], "\n"
PRINT FILE FileNam boldon, "FIRST_INS_WGT:      ", boldoff, [WGT_FIRST_INSULT], "grams", "\n"
PRINT FILE FileNam boldon, "REMOVAL_DATE:       ", boldoff, [REMOVAL_DATE], "\n"
PRINT FILE FileNam boldon, "DEATH_DATE:         ", boldoff, [DEATH_DATE], "\n"
PRINT FILE FileNam boldon, "DISPOSITION:        ", boldoff, [DISPOSITION], "\n"
PRINT FILE FileNam boldon, "TISS_ANALYSIS:      ", boldoff, [TISS_QUALITY], "\n"

/* wordwrap the cause of death */
PRINT FILE FileNam boldon, "COD: ", boldoff
IF ( cod_lines <> 0 ) THEN
  PRINT FILE FileNam SPACES(11), cod_array[1], "\n"

  FOR i FROM 2 TO cod_lines
    PRINT FILE FileNam SPACES(16), cod_array[i], "\n"
  ENDFOR
  RELEASE VARS cod_array
ELSE
  PRINT FILE FileNam "\n"
ENDIF

/* wordwrap the findings */
PRINT FILE FileNam boldon, "FINDINGS: ", boldoff
IF ( findings_lines <> 0 ) THEN
  PRINT FILE FileNam SPACES(6), findings_array[1], "\n"

  FOR i FROM 2 TO findings_lines
    PRINT FILE FileNam SPACES(16), findings_array[i], "\n"
  ENDFOR
  RELEASE VARS findings_array
ELSE
  PRINT FILE FileNam "\n"
ENDIF

PRINT FILE FileNam boldon, "ACCESSION_DATE: ", boldoff, [ACCESSION_DATE], "\n"

ENDPROC
WRITELIB LIBNAME FPrintIani
RELEASE PROCS FPrintIani
;*****
/* END MODULE: x.x FPrintIani
;*****

```

```

;*****
;* PROCEDURE: PrintMaster()
;* MODULE:
;* PURPOSE: Prints the Inventory Master record linked to the Inventory Detail.
;* CALLED BY:
;* CALLS:
;* INPUT ASSERTION:
;* OUTPUT ASSERTION:
;* GLOBAL VARIABLES:
;* AUTHOR: Sean K. Smith
;* DATE: 9/26/91
;*****/
PROC PrintMaster()
  PRIVATE table_title, /* title of table */
         no_chars /* number of chars required to pad the title */

  MOVETO "I\\D\\MASTER"

  AddLineNumber(11) /* 11 lines required */
  PRINT "\n\n"

  /* calculate and print the title */
  table_title = " Inventory Master Table " /* note spaces on either side */
  no_chars = INT((80 - LEN(table_title))/2)
  table_title = FILL(" ",no_chars) + table_title + FILL(" ",no_chars)
  PRINT boldon,table_title,boldoff,"\n\n"

  /* print the fields */
  PRINT boldon, "ITEM: ", boldoff,[ITEM_CODE], gap, gap , gap, gap, gap
  PRINT boldon, "ACCESSION_DATE: ", boldoff,[ACCESSION_DATE], "\n"
  PRINT boldon, "FORE_CODE: ", boldoff,[FORE_CODE], "\n"
  PRINT boldon, "SNOOOG: ", boldoff,[SNOOOG_CODE], "\n"
  PRINT boldon, "CONTAINER: ", boldoff, [CONTAINER_CODE], "\n"
  PRINT boldon, "ANI_QTY: ", boldoff, [ANI_QTY], "\n"
  PRINT boldon, "ITEM_QTY: ", boldoff, [ITH_QTY], "\n"
  PRINT boldon, "COMMENTS: ", boldoff, [COMMENTS], "\n"

ENDPROC
WRITELIB LIBNAME PrintMaster
RELEASE PROCS PrintMaster
;*****
;* END MODULE: x.x PrintMaster
;*****/

```

```

;*****
;* PROCEDURE: FPrintMaster()
;* MODULE:
;* PURPOSE: Print the Inventory Master table linked to the Inventory Detail.
;* CALLED BY: FileInv()
;* CALLS:
;* INPUT ASSERTION:
;* OUTPUT ASSERTION:
;* GLOBAL VARIABLES:
;* AUTHOR: Sean K. Smith
;* DATE: 9/26/91
;*****/
PROC FPrintMaster()
  PRIVATE table_title, /* title of table */
         no_chars /* number of chars required to pad the title */

  MOVETO "I\\D\\MASTER"

  FAddLineNumber(11) /* 11 lines required */
  PRINT FILE FileNam "\n\n"

  /* calculate and print the title */
  table_title = " Inventory Master Table " /* note spaces on either side */
  no_chars = INT((80 - LEN(table_title))/2)
  table_title = FILL(" ",no_chars) + table_title + FILL(" ",no_chars)
  PRINT FILE FileNam boldon,table_title,boldoff,"\n\n"

  /* print the fields */
  PRINT FILE FileNam boldon, "ITEM:      ", boldoff,[ITEM_CODE], gap, gap , gap, gap, gap
  PRINT FILE FileNam boldon, "ACCESSION_DATE: ", boldoff,[ACCESSION_DATE], "\n"
  PRINT FILE FileNam boldon, "FORE_CODE: ", boldoff,[FORE_CODE], "\n"
  PRINT FILE FileNam boldon, "SNODOG:      ", boldoff,[SNODOG_CODE], "\n"
  PRINT FILE FileNam boldon, "CONTAINER: ", boldoff, [CONTAINER_CODE], "\n"
  PRINT FILE FileNam boldon, "ANI_QTY:     ", boldoff, [ANI_QTY], "\n"
  PRINT FILE FileNam boldon, "ITEM_QTY:    ", boldoff, [ITM_QTY], "\n"
  PRINT FILE FileNam boldon, "COMMENTS:    ", boldoff, [COMMENTS], "\n"

ENDPROC
WRITELIB LIBNAME FPrintMaster
RELEASE PROCS FPrintMaster
;*****
;* END MODULE: x.x FPrintMaster
;*****/

```

```

;*****
; * PROCEDURE: PrintLocation()
; * MODULE:
; * PURPOSE: Prints the location record linked by the BAR_CODE field
; * CALLED BY: PrintInv() and PrintBib()
; * CALLS:
; * INPUT ASSERTION:
; * OUTPUT ASSERTION:
; * GLOBAL VARIABLES:
; * AUTHOR: Sean K. Smith
; * DATE: 9/26/91
;*****/
PROC PrintLocation()
    PRIVATE table_title, /* title of report section */
           no_chars      /* number of chars to pad title */

    MOVETO "I\\D\\LOCATION"

    AddLineNumber(12) /* number of lines required */
    PRINT "\n\n"

    /* calculate and print the title */
    table_title = " Inventory Location Table " /* note spaces on either side */
    no_chars = INT((80 - LEN(table_title))/2)
    table_title = FILL(" ",no_chars) + table_title + FILL(" ",no_chars)
    PRINT boldon,table_title,boldoff,"\n\n"

    /* print the fields */
    PRINT boldon, "LAB:      ", boldoff,[LAB], "\n"
    PRINT boldon, "DATE:      ", boldoff,[LOCATION_DATE], "\n"
    PRINT boldon, "STATUS:    ", boldoff,[STATUS], "\n"
    PRINT boldon, "BUILDING:  ", boldoff,[BUILDING], "\n"
    PRINT boldon, "ROOM:      ", boldoff,[ROOM], "\n"
    PRINT boldon, "CABINET:   ", boldoff,[CABINET], "\n"
    PRINT boldon, "SHELF:     ", boldoff,[SHELF], "\n"
    PRINT boldon, "POS:       ", boldoff,[POS], "\n"

ENDPROC
WRITELIB LIBNAME PrintLocation
RELEASE PROCS PrintLocation
;*****
; * END MODULE: x.x PrintLocation
;*****/

```

```

;*****
;* PROCEDURE: FPrintLocation()
;* MODULE:
;* PURPOSE: Prints the location record which is linked by the BAR_CODE field
;* CALLED BY: FileInv() and FileBib()
;* CALLS:
;* INPUT ASSERTION:
;* OUTPUT ASSERTION:
;* GLOBAL VARIABLES:
;* AUTHOR: Sean K. Smith
;* DATE: 9/26/91
;*****/
PROC FPrintLocation()
  PRIVATE table_title, /* title of report section */
         no_chars      /* number of chars to pad title */

  MOVETO "I\\D\\LOCATION"

  FAddLineNumber(12) /* number of lines required */
  PRINT FILE FileNam "\n\n"

  /* calculate and print the title */
  table_title = " Inventory Location Table " /* note spaces on either side */
  no_chars = INT((80 - LEN(table_title))/2)
  table_title = FILL(" ",no_chars) + table_title + FILL(" ",no_chars)
  PRINT FILE FileNam boldon,table_title,boldoff,"\n\n"

  /* print the fields */
  PRINT FILE FileNam boldon, "LAB:      ", boldoff,[LAB], "\n"
  PRINT FILE FileNam boldon, "DATE:   ", boldoff, [LOCATION_DATE],"\n"
  PRINT FILE FileNam boldon, "STATUS:  ", boldoff, [STATUS], "\n"
  PRINT FILE FileNam boldon, "BUILDING: ", boldoff,[BUILDING], "\n"
  PRINT FILE FileNam boldon, "ROOM:    ", boldoff,[ROOM], "\n"
  PRINT FILE FileNam boldon, "CABINET: ", boldoff, [CABINET], "\n"
  PRINT FILE FileNam boldon, "SHELF:   ", boldoff, [SHELF], "\n"
  PRINT FILE FileNam boldon, "POS:     ", boldoff, [POS], "\n"

ENDPROC
WRITELIB LIBNAME FPrintLocation
RELEASE PROCS FPrintLocation
;*****
;* END MODULE: x.x FPrintLocation
;*****/

```

```

; /*****
; * PROCEDURE: PrintBib()
; * MODULE:
; * PURPOSE: Driver proc to print out the current record of the BOOK table and
; *           the records of the tables linked to it through the current form.
; * CALLED BY: PrintScreen()
; * CALLS:
; * INPUT ASSERTION:
; * OUTPUT ASSERTION:
; * GLOBAL VARIABLES:
; * AUTHOR: Sean X. Smith
; * DATE: 9/26/91
; *****/
PROC PrintBib()

    InitPrintVars()

    OPEN PRINTER

    StartOfPage()
    PrintBook()
    PrintAuthor()
    PrintLocation()
    EndOfPage()

    CLOSE PRINTER

    CLEAR

ENDPROC
WRITELIB LIBNAME PrintBib
RELEASE PROCS PrintBib
; /*****
; * END MODULE: x.x PrintBib()
; *****/

```

```

;*****
;* PROCEDURE: FileBib()
;* MODULE:
;* PURPOSE: Print to a file the current record of the BOOK table and all
;*          the records in the table linked to it through the current.
;* CALLED BY:
;* CALLS:
;* INPUT ASSERTION:
;* OUTPUT ASSERTION:
;* GLOBAL VARIABLES:
;* AUTHOR: Sean K. Smith
;* DATE: 9/26/91
;*****/
PROC FileBib()

    InitPrintVars()

    FStartOfPage()
    FPrintBook()
    FPrintAuthor()
    FPrintLocation()
    FEndOfPage()

    CLEAR

ENDPROC
WRITELIB LIBNAME FileBib
RELEASE PROCS FileBib
;*****
;* END MODULE: x.x FileBib()
;*****/

```

```

;*****
;* PROCEDURE: PrintBook()
;* MODULE:
;* PURPOSE: Prints out the current record of the BOOK table.
;* CALLED BY: PrintBib()
;* CALLS:
;* INPUT ASSERTION:
;* OUTPUT ASSERTION:
;* GLOBAL VARIABLES:
;* AUTHOR: Sean K. Smith
;* DATE: 9/26/91
;*****
PROC PrintBook()
    PRIVATE x,                /* number of total lines required on the page */
            i,                /* counter of FOR loop */
            table_title,      /* holds the table title */
            no_chars,         /* temp to hold number of pad chars in title */
            title_lines,      /* title */
            title_ln_add,     /* number of additional required on page */
            subtitle_lines,   /* number of lines FINDINGS wrapped onto */
            sub_ln_add,       /* number of additional required on page */
            journal_lines,    /* number of JOURNAL LINES */
            journ_ln_add,     /* number of additional required on page */
            journal_sub_lines, /* number of JOURNAL_SUBTITLE lines */
            jsub_ln_add       /* number of additional require on page */

    /* word wrap the TITLE field and calculate the number of additional */
    title_lines = WordWrap([TITLE], "title_array")
    IF ( title_lines > 1) THEN
        title_ln_add = title_lines - 1
    ELSE
        title_ln_add = 0
    ENDIF

    /* word wrap the SUBTITLE field and calculate the number of additional */
    subtitle_lines = WordWrap([SUBTITLE], "subtitle_array")
    IF ( subtitle_lines > 1) THEN
        sub_ln_add = subtitle_lines - 1
    ELSE
        sub_ln_add = 0
    ENDIF

    /* word wrap the journal field and calculate the number of additional */
    journal_lines = WordWrap([JOURNAL], "journal_array")
    IF ( journal_lines > 1) THEN
        journ_ln_add = journal_lines - 1
    ELSE
        journ_ln_add = 0
    ENDIF

    /* word wrap the JOURNAL_SUBTITLE field and calculate # of additional */
    journal_sub_lines = WordWrap([JOURNAL_SUBTITLE], "journal_sub_array")
    IF ( journal_sub_lines > 1) THEN
        jsub_ln_add = journal_sub_lines - 1
    ELSE
        jsub_ln_add = 0
    ENDIF

    x = title_ln_add + sub_ln_add + journ_ln_add + jsub_ln_add + 11

```

```

AddLineNumber(x + 2) ;/* lines required to print record + two blank lines */
PRINT "\n\n"

;/* calculate the title and print it */
table_title = " Bibliography Book Table " ;/* note spaces on either side */
no_chars = INT((80 - LEN(table_title))/2)
table_title = FILL(" ",no_chars) + table_title + FILL(" ",no_chars)
PRINT boldon,table_title,boldoff,"\n\n"

;/* print the fields of the table */
PRINT boldon, "BAR_CODE:  ",[BAR_CODE],"\n"

;/* wordwrap the title */
PRINT boldon, "TITLE:    ",boldoff
IF ( title_lines <> 0 ) THEN
    PRINT SPACES(0), title_array[1],"\n"

    FOR i FROM 2 TO title_lines
        PRINT SPACES(12), title_array[i], "\n"
    ENDFOR
    RELEASE VARS title_array
ELSE
    PRINT "\n"
ENDIF

;/* wordwrap the subtitle */
PRINT boldon, "SUBTITLE:  ",boldoff
IF ( subtitle_lines <> 0 ) THEN
    PRINT SPACES(0), subtitle_array[1],"\n"

    FOR i FROM 2 TO subtitle_lines
        PRINT SPACES(12), subtitle_array[i], "\n"
    ENDFOR
    RELEASE VARS subtitle_array
ELSE
    PRINT "\n"
ENDIF

;/* wordwrap the journal */
PRINT boldon, "JOURNAL:    ",boldoff
IF ( journal_lines <> 0 ) THEN
    PRINT SPACES(0), journal_array[1],"\n"

    FOR i FROM 2 TO journal_lines
        PRINT SPACES(12), journal_array[i], "\n"
    ENDFOR
    RELEASE VARS journal_array
ELSE
    PRINT "\n"
ENDIF

;/* wordwrap the journal subtitle */
PRINT boldon, "J_SUBTITLE: ",boldoff
IF ( journal_sub_lines <> 0 ) THEN
    PRINT SPACES(0), journal_sub_array[1],"\n"

    FOR i FROM 2 TO journal_sub_lines
        PRINT SPACES(12), journal_sub_array[i], "\n"
    ENDFOR
    RELEASE VARS journal_sub_array
ELSE

```

```

        PRINT "\n"
    ENDIF

    PRINT boldon, "PUBLISHER: ", boldoff, [PUBLISHER], "\n"
    PRINT boldon, "LOCATION: ", boldoff, [LOCATION], "\n"
    PRINT boldon, "VOLUME: ", boldoff, [VOLUME], gap
    PRINT boldon, "YEAR: ", boldoff, [PUB_YR], gap
    PRINT boldon, "PAGES: ", boldoff, [PAGES], "\n"
    PRINT boldon, "REPORT_NO: ", boldoff, [REPORT_NO], "\n"
    PRINT boldon, "DOC_TYPE: ", boldoff, [DOC_TYPE], gap
    PRINT boldon, "DONOR_ID: ", boldoff, [DONOR_ID], gap
    PRINT boldon, "LAB_ID: ", boldoff, [LAB_ID], "\n"
    PRINT boldon, "ABSTRACT: ", boldoff, [ABSTRACT_FLAG], gap
    PRINT boldon, "KEYWORD: ", boldoff, [KEY_WORD_FLAG], gap
    PRINT boldon, "INVENTORY_FLAG: ", boldoff, [INVENTORY_FLAG], "\n"

ENDPROC
WRITELIB LIBNAME PrintBook
RELEASE PROCS PrintBook
; /*****
; * END MODULE: x.x PrintBook()
; *****/

```

```

;*****
;* PROCEDURE: FPrintBook()
;* MODULE:
;* PURPOSE: Prints the current BOOK table record to a file.
;* CALLED BY: FileBib()
;* CALLS:
;* INPUT ASSERTION:
;* OUTPUT ASSERTION:
;* GLOBAL VARIABLES:
;* AUTHOR: Sean K. Smith
;* DATE: 9/26/91
;*****/
PROC FPrintBook()
  PRIVATE x,                ;/* number of total lines required on the page */
         i,                ;/* counter of FOR loop */
         table_title,      ;/* holds the table title */
         no_chars,         ;/* temp to hold number of pad chars in title */
         title_lines,      ;/* title */
         title_ln_add,     ;/* number of additional required on page */
         subtitle_lines,   ;/* number of lines FINDINGS wrapped onto */
         sub_ln_add,       ;/* number of additional required on page */
         journal_lines,    ;/* number of JOURNAL LINES */
         journ_ln_add,     ;/* number of additional required on page */
         journal_sub_lines, ;/* number of JOURNAL_SUBTITLE lines */
         jsub_ln_add       ;/* number of additional require on page */

  ;/* word wrap the TITLE field and calculate the number of additional */
  title_lines = WordWrap([TITLE], "title_array")
  IF ( title_lines > 1) THEN
    title_ln_add = title_lines - 1
  ELSE
    title_ln_add = 0
  ENDIF

  ;/* word wrap the SUBTITLE field and calculate the number of additional */
  subtitle_lines = WordWrap([SUBTITLE], "subtitle_array")
  IF ( subtitle_lines > 1) THEN
    sub_ln_add = subtitle_lines - 1
  ELSE
    sub_ln_add = 0
  ENDIF

  ;/* word wrap the journal field and calculate the number of additional */
  journal_lines = WordWrap([JOURNAL], "journal_array")
  IF ( journal_lines > 1) THEN
    journ_ln_add = journal_lines - 1
  ELSE
    journ_ln_add = 0
  ENDIF

  ;/* word wrap the JOURNAL_SUBTITLE field and calculate # of additional */
  journal_sub_lines = WordWrap([JOURNAL_SUBTITLE], "journal_sub_array")
  IF ( journal_sub_lines > 1) THEN
    jsub_ln_add = journal_sub_lines - 1
  ELSE
    jsub_ln_add = 0
  ENDIF

  x = title_ln_add + sub_ln_add + journ_ln_add + jsub_ln_add + 11

```

```

FAddLineNumber(x + 2) ;/* lines required to print record + two blank lines */
PRINT FILE FileNam "\n\n"

;/* calculate the title and print it */
table_title = " Bibliography Book Table " ;/* note spaces on either side */
no_chars = INT((80 - LEN(table_title))/2)
table_title = FILL(" ",no_chars) + table_title + FILL(" ",no_chars)
PRINT FILE FileNam boldon,table_title,boldoff,"\n\n"

;/* print the fields of the table */
PRINT FILE FileNam boldon, "BAR_CODE:  ",[BAR_CODE],"\n"

;/* wordwrap the title */
PRINT FILE FileNam boldon, "TITLE:      ",boldoff
IF ( title_lines <> 0 ) THEN
    PRINT FILE FileNam SPACES(0), title_array[1],"\n"

    FOR i FROM 2 TO title_lines
        PRINT FILE FileNam SPACES(12), title_array[i], "\n"
    ENDFOR
    RELEASE VARS title_array
ELSE
    PRINT FILE FileNam "\n"
ENDIF

;/* wordwrap the subtitle */
PRINT FILE FileNam boldon, "SUBTITLE:  ",boldoff
IF ( subtitle_lines <> 0 ) THEN
    PRINT FILE FileNam SPACES(0), subtitle_array[1],"\n"

    FOR i FROM 2 TO subtitle_lines
        PRINT FILE FileNam SPACES(12), subtitle_array[i], "\n"
    ENDFOR
    RELEASE VARS subtitle_array
ELSE
    PRINT FILE FileNam "\n"
ENDIF

;/* wordwrap the journal */
PRINT FILE FileNam boldon, "JOURNAL:    ",boldoff
IF ( journal_lines <> 0 ) THEN
    PRINT FILE FileNam SPACES(0), journal_array[1],"\n"

    FOR i FROM 2 TO journal_lines
        PRINT FILE FileNam SPACES(12), journal_array[i], "\n"
    ENDFOR
    RELEASE VARS journal_array
ELSE
    PRINT FILE FileNam "\n"
ENDIF

;/* wordwrap the journal subtitle */
PRINT FILE FileNam boldon, "J_SUBTITLE: ",boldoff
IF ( journal_sub_lines <> 0 ) THEN
    PRINT FILE FileNam SPACES(0), journal_sub_array[1],"\n"

    FOR i FROM 2 TO journal_sub_lines
        PRINT FILE FileNam SPACES(12), journal_sub_array[i], "\n"
    ENDFOR
    RELEASE VARS journal_sub_array
ELSE

```

```

PRINT FILE FileNam "\n"
ENDIF

```

```

PRINT FILE FileNam boldon, "PUBLISHER: ", boldoff, [PUBLISHER], "\n"
PRINT FILE FileNam boldon, "LOCATION: ", boldoff, [LOCATION], "\n"
PRINT FILE FileNam boldon, "VOLUME: ", boldoff, [VOLUME], gap
PRINT FILE FileNam boldon, "YEAR: ", boldoff, [PUB_YR], gap
PRINT FILE FileNam boldon, "PAGES: ", boldoff, [PAGES], "\n"
PRINT FILE FileNam boldon, "REPORT_NO: ", boldoff, [REPORT_NO], "\n"
PRINT FILE FileNam boldon, "DOC_TYPE: ", boldoff, [DOC_TYPE], gap
PRINT FILE FileNam boldon, "DONOR_ID: ", boldoff, [DONOR_ID], gap
PRINT FILE FileNam boldon, "LAB_ID: ", boldoff, [LAB_ID], "\n"
PRINT FILE FileNam boldon, "ABSTRACT: ", boldoff, [ABSTRACT_FLAG], gap
PRINT FILE FileNam boldon, "KEYWORD: ", boldoff, [KEY_WORD_FLAG], gap
PRINT FILE FileNam boldon, "INVENTORY_FLAG: ", boldoff, [INVENTORY_FLAG], "\n"

```

```

ENDPROC
WRITELIB LIBNAME FPrintBook
RELEASE PROCS FPrintBook
; /*****
; * END MODULE: x.x FPrintBook()
; *****/

```

```

;*****
;* PROCEDURE: PrintAuthor()
;* MODULE:
;* PURPOSE: Prints the record of the AUTHOR table which are linked to the
;*           current BOOK record.
;* CALLED BY: PrintBib()
;* CALLS:
;* INPUT ASSERTION:
;* OUTPUT ASSERTION:
;* GLOBAL VARIABLES:
;* AUTHOR: Sean K. Smith
;* DATE: 9/26/91
;*****/
PROC PrintAuthor()
PRIVATE numofdetail, /* count of author records linked */
       table_title, /* title */
       no_chars, /* number of chars to pad title */
       i /* FOR loop counter */

/* prepare table title */
table_title = " Bibliography Author Table " /* note spaces on either side */
no_chars = INT((80 - LEN(table_title))/2)
table_title = FILL(" ",no_chars) + table_title + FILL(" ",no_chars)

MOVE TO "B\ \D\ \AUTHOR"
numofdetail = NIMAGERECORDS()

/* just print table headers if no images */
IF numofdetail = 0 THEN
  AddLineNumber(8)
  PRINT "\n\n"
  PRINT boldon,table_title,boldoff,"\n\n"
  PRINT boldon,
    FORMAT("w15","LAST_NAME"),gap,
    FORMAT("w15","FIRST_NAME"),gap,
    FORMAT("w15","MIDDLE"),gap,
    FORMAT("w8","STATUS"),gap,
    FORMAT("w6","RANK"),"\n"
  PRINT FILL("-",15),gap,
    FILL("-",15),gap,
    FILL("-",15),gap,
    FILL("-",8),gap,
    FILL("-",6),boldoff,"\n"
  PRINT "Nothing to report"
  RETURN
/* records linked print header and scan image */
ELSE
  AddLineNumber(7)
  PRINT "\n\n"
  PRINT boldon,table_title,boldoff,"\n\n"
  PRINT boldon,
    FORMAT("w15","LAST_NAME"),gap,
    FORMAT("w15","FIRST_NAME"),gap,
    FORMAT("w15","MIDDLE"),gap,
    FORMAT("w8","STATUS"),gap,
    FORMAT("w6","RANK"),"\n"
  PRINT FILL("-",15),gap,
    FILL("-",15),gap,
    FILL("-",15),gap,
    FILL("-",8),gap,
    FILL("-",6),boldoff,"\n"

```

```

/* scan and check to see if new page starts */
SCAN
  i = AddLineNumber(1) /* returns 1 if a new page is started */

  /* check to see if a new page has started */
  IF (i = 1) THEN
    AddLineNumber(7)
    PRINT "\n\n"
    PRINT boldon,table_title,boldoff,"\n\n"
    PRINT boldon,
      FORMAT("w15","LAST_NAME"),gap,
      FORMAT("w15","FIRST_NAME"),gap,
      FORMAT("w15","MIDDLE"),gap,
      FORMAT("w8","STATUS"),gap,
      FORMAT("w6","RANK"),"\n"
    PRINT FILL("-",15),gap,
      FILL("-",15),gap,
      FILL("-",15),gap,
      FILL("-",8),gap,
      FILL("-",6),boldoff,"\n"

    ENDIF

    /* print contents of the fields */
    PRINT FORMAT("w15",[LAST_NAME]),gap,
      FORMAT("w15",[FIRST_NAME]),gap,
      FORMAT("w15",[MIDDLE]),gap,
      FORMAT("w8",[STATUS]),gap,
      FORMAT("w3",[RANK]),"\n"

  ENDSCAN

ENDIF

ENDPROC
Writelib LIBNAME PrintAuthor
RELEASE PROCS PrintAuthor
/* *****
/* END MODULE: x.x PrintAuthor
/* *****

```

```

/******
; * PROCEDURE: FPrintAuthor()
; * MODULE:
; * PURPOSE: Prints to a file the author table records which are linked to the
; *           current book table record.
; * CALLED BY: FileBib()
; * CALLS:
; * INPUT ASSERTION:
; * OUTPUT ASSERTION:
; * GLOBAL VARIABLES:
; * AUTHOR: Sean K. Smith
; * DATE: 9/26/91
; *****/
PROC FPrintAuthor()
PRIVATE numofdetail, /* count of author records linked */
       table_title, /* title */
       no_chars, /* number of chars to pad title */
       i /* FOR loop counter */

/* prepare table title */
table_title = " Bibliography Author Table " /* note spaces on either side */
no_chars = INT((80 - LEN(table_title))/2)
table_title = FILL(" ",no_chars) + table_title + FILL(" ",no_chars)

MOVE TO "B\D\AUTHOR"
numofdetail = NIMAGERECORDS()

/* just print table headers if no images */
IF numofdetail = 0 THEN
  FAddLineNumber(8)
  PRINT FILE FileName "\n\n"
  PRINT FILE FileName boldon,table_title,boldoff,"\n\n"
  PRINT FILE FileName boldon,
    FORMAT("w15","LAST_NAME"),gap,
    FORMAT("w15","FIRST_NAME"),gap,
    FORMAT("w15","MIDDLE"),gap,
    FORMAT("w8","STATUS"),gap,
    FORMAT("w6","RANK"),"\n\n"
  PRINT FILE FileName FILL("-",15),gap,
    FILL("-",15),gap,
    FILL("-",15),gap,
    FILL("-",8),gap,
    FILL("-",6),boldoff,"\n"
  PRINT FILE FileName "Nothing to report"
  RETURN
/* records linked print header and scan image */
ELSE
  FAddLineNumber(7)
  PRINT FILE FileName "\n\n"
  PRINT FILE FileName boldon,table_title,boldoff,"\n\n"
  PRINT FILE FileName boldon,
    FORMAT("w15","LAST_NAME"),gap,
    FORMAT("w15","FIRST_NAME"),gap,
    FORMAT("w15","MIDDLE"),gap,
    FORMAT("w8","STATUS"),gap,
    FORMAT("w6","RANK"),"\n"
  PRINT FILE FileName FILL("-",15),gap,
    FILL("-",15),gap,
    FILL("-",15),gap,
    FILL("-",8),gap,
    FILL("-",6),boldoff,"\n"

```

```

; /* scan and check to see if new page starts */
SCAN
  i = FAddLineNumber(1) ; /* returns 1 if a new page is started */

; /* check to see if a new page has started */
IF (i = 1) THEN
  FAddLineNumber(7)
  PRINT FILE FileNam "\n\n"
  PRINT FILE FileNam boldon, table_title, boldoff, "\n\n"
  PRINT FILE FileNam boldon,
    FORMAT("w15", "LAST_NAME"), gap,
    FORMAT("w15", "FIRST_NAME"), gap,
    FORMAT("w15", "MIDDLE"), gap,
    FORMAT("w8", "STATUS"), gap,
    FORMAT("w6", "RANK"), "\n"
  PRINT FILE FileNam FILL("-", 15), gap,
    FILL("-", 15), gap,
    FILL("-", 15), gap,
    FILL("-", 8), gap,
    FILL("-", 6), boldoff, "\n"

  ENDIF

; /* print contents of the fields */
PRINT FILE FileNam FORMAT("w15", [LAST_NAME]), gap,
  FORMAT("w15", [FIRST_NAME]), gap,
  FORMAT("w15", [MIDDLE]), gap,
  FORMAT("w8", [STATUS]), gap,
  FORMAT("w3", [RANK]), "\n"

ENDSCAN

ENDIF

ENDPROC
WRITELIB LIBNAME FPrintAuthor
RELEASE PROCS FPrintAuthor
; /*****
; * END MODULE: x.x FPrintAuthor
; *****/

```

```

;*****
;* PROCEDURE: InitPrintVars()
;* MODULE:
;* PURPOSE: Initializes global variables used by printing routines
;*          PrintScreen is closed thus globals released.
;* COMMENTS: In the future may want to ask type of printer used and assign
;*          printer command strings to bold, underline etc. Also may
;*          want to ask which port to print to and set the port with
;*          the SETPRINTER command.
;* CALLED BY: called by the 6 print and file procs
;* CALLS:
;* INPUT ASSERTION:
;* OUTPUT ASSERTION:
;* GLOBAL VARIABLES:
;* AUTHOR: Sean K. Smith
;* DATE: 10/2/91
;*****/
PROC InitPrintVars()

    pagesize = 60
    boldon   = ""
    boldoff  = ""
    underlineon = ""
    underlineoff = ""
    linenum  = 1
    pagefooter = 3
    gap      = " "
    pgnum    = 1

ENDPROC
WRITELIB LIBNAME InitPrintVars
RELEASE PROCS InitPrintVars
;*****
;* END MODULE: x.x InitPrintVars()
;*****/

```

```

; /*****
; * PROCEDURE: StartOfPage()
; * MODULE:
; * PURPOSE: Print the header at the start of a new page
; * CALLED BY: AddLineNumber() and procs which drive reports.
; * CALLS:
; * INPUT ASSERTION:
; * OUTPUT ASSERTION:
; * GLOBAL VARIABLES:
; * AUTHOR: Adapted from PPPT by Salcedo & Rudy
; * DATE:
; *****/
PROC StartOfPage()

    PRINT "\n", boldon, FORMAT("w80,ac","NATIONAL RADIOBIOLOGY ARCHIVES") +
        boldoff, "\n\n"
    PRINT FORMAT("D1", TODAY()) + boldon +
        FORMAT("w64,ac", report_title) + boldoff +
        "Page: " + FORMAT("w4,al", pgnum) + "\n"
    linenum = linenum + 4

ENDPROC
WRITE LIB LIBNAME StartOfPage
RELEASE PROCS StartOfPage
; /*****
; * END MODULE: x.x StartOfPage()
; *****/

```

```

;*****
; * PROCEDURE: FStartOfPage()
; * MODULE:
; * PURPOSE: Prints header to file at the start of a new page.
; * CALLED BY: FAddLineNumber() and proc print file drivers
; * CALLS:
; * INPUT ASSERTION:
; * OUTPUT ASSERTION:
; * GLOBAL VARIABLES:
; * AUTHOR: Adapted from PPPT by Salcedo & Rudy
; * DATE:
;*****/
PROC FStartOfPage()
  PRINT FILE FileNam "\n",boldon,
    FORMAT("w80,ac","NATIONAL RADIOBIOLOGY ARCHIVES") +
    boldoff,"\n\n"
  PRINT FILE FileNam FORMAT("D1",TODAY()) + boldon +
    FORMAT("w64,ac",report_title) + boldoff +
    "Page: " + FORMAT("w4,al",pgnum) + "\n"
  linenumber = linenumber + 4
ENDPROC
WRITELIB LIBNAME FStartOfPage
RELEASE PROCS FStartOfPage
;*****
; * END MODULE: x.x FStartOfPage()
;*****/

```

```

;*****
;* PROCEDURE: EndOfPage()
;* MODULE:
;* PURPOSE: Prints footer at the bottom of a page.
;* COMMENTS: Note this resets the line number to 1
;* CALLED BY: AddLineNumber() procs which a print drivers
;* CALLS:
;* INPUT ASSERTION:
;* OUTPUT ASSERTION:
;* GLOBAL VARIABLES:
;* AUTHOR: Adapted from PPPT by Salcedo & Rudy
;* DATE:
;*****/
PROC EndOfPage()
    WHILE linenumber < pagesize - pagefooter
        PRINT "\n"
        linenumber = linenumber + 1
    ENDWHILE
    PRINT boldon,"\n",FORMAT("w80,ac","NATIONAL RADIOBIOLOGY ARCHIVES"),
        boldoff,"\f"
    pgnum = pgnum + 1

    linenumber = 1 ;/* reinitialize linenumber */

ENDPROC
WRITELIB LIBNAME EndOfPage
RELEASE PROCS EndOfPage
;*****
;* END MODULE: x.x EndOfPage()
;*****/

```

```

; /*****
; * PROCEDURE: FEndOfPage()
; * MODULE:
; * PURPOSE: Prints footer at the bottom of a page.
; * COMMENTS: Note this resets the line number to 1
; * CALLED BY: FAddLineNumber() & procs which are print drivers
; * CALLS:
; * INPUT ASSERTION:
; * OUTPUT ASSERTION:
; * GLOBAL VARIABLES:
; * AUTHOR: Adapted from PPPT by Salcedo & Rudy
; * DATE:
; *****/
PROC FEndOfPage()
    WHILE linenumber < pagesize - pagefooter
        PRINT FILE FileNam "\n"
        linenumber = linenumber + 1
    ENDWHILE

    PRINT FILE FileNam boldon,"\n",FORMAT("w80,ac","NATIONAL RADIOBIOLOGY ARCHIVES"),
        boldoff,"\f"
    pgnum = pgnum + 1
    linenumber = 1
ENDPROC
WRITELIB LIBNAME FEndOfPage
RELEASE PROCS FEndOfPage
; /*****
; * END MODULE: x.x FEndOfPage()
; *****/

```

```

;*****
;* PROCEDURE: AddLineNumber()
;* MODULE:
;* PURPOSE: Increments global LineNumber variable which tracks number of lines
;*          used or needed.
;* COMMENTS:
;* CALLED BY: All procs which print in this module.
;* CALLS:
;* INPUT ASSERTION:
;* OUTPUT ASSERTION: Returns a 1 if a Call to StartOfPage() occurs.
;*                  Returns a 0 if no call to StartOfPage().
;* GLOBAL VARIABLES:
;* AUTHOR: Adapted from PPPT by Salcedo & Rudy.
;* DATE:
;*****/
PROC AddLineNumber(incrmnt)
  IF linenumber + incrmnt > pagesize - pagefooter THEN
    EndOfPage()
    StartOfPage()
    linenumber = linenumber + incrmnt ;/* bug fix */
    RETURN 1
  ELSE
    linenumber = linenumber + incrmnt
    RETURN 0
  ENDIF

ENDPROC
WRITELIB LIBNAME AddLineNumber
RELEASE PROCS AddLineNumber
;*****
;* END MODULE: x.x AddLineNumber()
;*****/

```

```

;*****
;* PROCEDURE: FAddLineNumber()
;* MODULE:
;* PURPOSE: Increments global LineNumber variable which tracks number of lines
;*           used or needed.
;* COMMENTS:
;* CALLED BY: All procs which print in this module.
;* CALLS:
;* INPUT ASSERTION:
;* OUTPUT ASSERTION: Returns a 1 if a Call to StartOfPage() occurs.
;*                   Returns a 0 if no call to StartOfPage().
;* GLOBAL VARIABLES:
;* AUTHOR: Adapted from PPPT by Salcedo & Rudy.
;* DATE:
;*****/
PROC FAddLineNumber(incrmnt)
  IF linenumber + incrmnt > pagesize - pagefooter THEN
    FEndOfPage()
    FStartOfPage()
    linenumber = linenumber + incrmnt ;/* bug fix */
    RETURN 1
  ELSE
    linenumber = linenumber + incrmnt
    RETURN 0
  ENDIF

ENDPROC
WRITELIB LIBNAME FAddLineNumber
RELEASE PROCS FAddLineNumber
;*****
;* END MODULE: x.x FAddLineNumber()
;*****/

```

```

;*****
;* PROCEDURE: WordWrap(string, arrayname)
;* MODULE:
;* PURPOSE: This procedure will accept a string and word wrap it into 60 char
;*          strings. The 60 char strings are placed in a global array which
;*          can then be printed out.
;* CALLED BY: printing procs which need to word wrap long lines of text.
;* CALLS:
;* INPUT ASSERTION:
;* OUTPUT ASSERTION: Returns the number of lines string is wrapped onto
;* GLOBAL VARIABLES: str_array[]
;* AUTHOR: Sean K. Smith
;* DATE: 9/24/91
;*****/
PROC WordWrap(string, arrayname)
  PRIVATE char_curr,      /* the current character */
          line_cnt,      /* the current line being assigned a 60 char str */
          max_cnt,       /* the number of chars in the main string */
          abs_cnt,       /* the number of chars parsed so far */
          lst_spc,       /* the pos of the last space in the main string */
          line_str,      /* the pos of the starting char of the line */
          rel_cnt,       /* the pos in the current str the cur line */
          curr_str,      /* the current line a string of 60 or less chars */
          i,             /* temp counter */
          j              /* temp value */

  /* check to see if the string is null */
  IF (ISBLANK(string)) THEN
    RETURN 0
  ENDIF

  /* initialize variables */
  max_cnt = LEN(string)

  line_cnt = 0
  abs_cnt = 0

  /* loop all chars are read and wrapped onto 60 char lines */
  WHILE (TRUE)
    line_cnt = line_cnt + 1
    line_str = abs_cnt + 1
    rel_cnt = 0
    lst_spc = 0

    /* loop until 60 chars read or the end of the string */
    WHILE ((rel_cnt < 60) AND (abs_cnt < max_cnt))
      rel_cnt = rel_cnt + 1
      abs_cnt = abs_cnt + 1
      char_curr = SUBSTR(string,abs_cnt,1)

      /* check to see if curr is a space or max reached */
      IF ((char_curr = " ") OR (abs_cnt = max_cnt)) THEN
        lst_spc = abs_cnt
      ENDIF
    ENDWHILE

    /* assign new values */
    curr_line = SUBSTR(string,line_str,lst_spc - line_str + 1)
    abs_cnt = lst_spc
    rel_cnt = 0

```

```

        /* assign the curr_line to a variable */
        EXECUTE "str" + STRVAL(line_cnt) + " = " + "\"" + curr_line + "\""

        IF (abs_cnt = max_cnt) THEN
            QUITLOOP
        ENDIF
    ENDWHILE

    /* create an array place the strings in there and then release var */

    EXECUTE "ARRAY " + arrayname + "[" + STRVAL(line_cnt) + "]"

    FOR i FROM 1 to line_cnt
        EXECUTE arrayname + "[" + STRVAL(i) + "]" + " = " + "str" + STRVAL(i)
        EXECUTE "RELEASE VARS " + "str" + STRVAL(i)
    ENDFOR

    RETURN line_cnt

ENDPROC
WRITELIB LIBNAME WordWrap
RELEASE PROCS WordWrap
;*****
;*/
;* END MODULE: x.x WordWrap()
;*****/

```

```

;*****
; PROCEDURE: HelpKey()
; MODULE:
; PURPOSE:
; CALLED BY:
; CALLS:
; INPUT ASSERTION:
; OUTPUT ASSERTION:
; GLOBAL VARIABLES:
; AUTHOR: JC Prather
; DATE: August 1991
; REVISED 9/13/91 BY: Chuck Watson
;           Help files now located in \HELP rather than \TEXT
;*****/
Proc HelpKey()
  PRIVATE HELPFLAG

  HELPFLAG= "Y"

  IMAGEMEM=IMAGENO()
  RECORDMEM=RECNO()
  FIELDMEM=FIELD()
  FIELDNAME = SUBSTR(FIELD(),1,MIN(8,LEN(FIELD())))

  filename = "HELP\\"+FIELDNAME+".hlp"
  if isfile(filename)= false then
    filename = "HELP\nohelp.hlp"
  endif
  TextEditor(filename,"HELP for field: "+FIELD())
  FORMKEY
  MOVETO IMAGEMEM
  MOVETO RECORD RECORDMEM
  MOVETO FIELD FIELDMEM
  RELEASE VARS IMAGEMEM,RECORDMEM,FIELDMEM

  RETURN
ENDPROC
WRITELIB LIBNAME HelpKey
RELEASE PROCS HelpKey
;*****
; END MODULE: x.x HelpKey()
;*****/

```

```

;*****
;* PROCEDURE: TitleScreen()
;* MODULE:
;* PURPOSE:
;* CALLED BY:
;* CALLS:
;* INPUT ASSERTION:
;* OUTPUT ASSERTION:
;* GLOBAL VARIABLES:
;* AUTHOR: JC Prather
;* DATE: August 1991
;*****/
Proc CLOSED TitleScreen()

    CANVAS OFF
    CURSOR OFF
    CLEARALL
    CLEAR

    FOR i FROM 0 TO 12
        Row1 = 0 + i
        Col1 = 0 + i
        Row2 = 24 - i
        Col2 = 79 - i
        PAINTCANVAS BORDER FILL "█" ATTRIBUTE i Row1, Col1, Row2, Col2
    ENDFOR

    CANVAS ON

    FOR i FROM 0 TO 12
        Row1 = 12 - i
        Col1 = 12 - i
        Row2 = 12 + i
        Col2 = 67 + i
        PAINTCANVAS BORDER ATTRIBUTE 0 Row1, Col1, Row2, Col2
    ENDFOR

    CANVAS OFF
    if monitor() = "Color" then
        PAINTCANVAS ATTRIBUTE 17 0, 0, 24, 79
    else PAINTCANVAS ATTRIBUTE BKCOLOR 0,0,24,79
    endif

    STYLE ATTRIBUTE 23
    @ 0, 0 ?? "r"
    @ 0, 79 ?? "l"
    @ 24, 0 ?? "l"
    @ 24, 79 ?? "r"
    PAINTCANVAS BORDER FILL CHR(205) ATTRIBUTE 23 0, 1, 0, 78
    PAINTCANVAS BORDER FILL CHR(205) ATTRIBUTE 23 24, 1, 24, 78
    PAINTCANVAS BORDER FILL CHR(186) ATTRIBUTE 23 1, 0, 23, 0
    PAINTCANVAS BORDER FILL CHR(186) ATTRIBUTE 23 1, 79, 23, 79

    PAINTCANVAS ATTRIBUTE 0 4, 22, 13, 57
    STYLE ATTRIBUTE 7
    @4,22

```

```

?? " "
@13,22
?? " "
PAINTCANVAS BORDER FILL CHR(186) ATTRIBUTE 7 5, 22, 12, 22
PAINTCANVAS BORDER FILL CHR(186) ATTRIBUTE 7 5, 57, 12, 57

STYLE ATTRIBUTE 15
@6,25
?? "NATIONAL RADIOBIOLOGY ARCHIVES"

@8, 23
?? FORMAT("W33,AC","Distributed Access")
@9, 23
?? FORMAT("W33,AC","Information System")

@11, 30
?? " Version 1.0 1991"
PAINTCANVAS BORDER FILL CHR(177) ATTRIBUTE 16 14, 23, 14, 57
PAINTCANVAS BORDER FILL CHR(177) ATTRIBUTE 16 5, 58, 14, 59

PAINTCANVAS ATTRIBUTE 0 18, 8, 21, 23

STYLE ATTRIBUTE 7
@ 18, 8
?? " "
@ 21,8
?? " "
@ 19, 8 ?? " "
@ 20, 8 ?? " "
@ 19, 25 ?? " "
@ 20, 25 ?? " "

@ 19, 9
?? " Sean Smith "
@ 20, 9
?? " (509) 376 5684 "

PAINTCANVAS BORDER FILL CHR(177) ATTRIBUTE 16 22, 9, 22, 26
PAINTCANVAS BORDER FILL CHR(177) ATTRIBUTE 16 19, 26, 22, 27

PAINTCANVAS ATTRIBUTE 0 18, 55, 21, 72

STYLE ATTRIBUTE 7
@ 18, 55
?? " "
@ 21, 55
?? " "
@ 19, 55 ?? " "
@ 20, 55 ?? " "
@ 19, 72 ?? " "
@ 20, 72 ?? " "

@ 19, 56
?? " Chuck Watson "
@ 20, 56

```

```

?? " (509) 376 3483 "
PAINTCANVAS BORDER FILL CHR(177) ATTRIBUTE 16 22, 56, 22, 72
PAINTCANVAS BORDER FILL CHR(177) ATTRIBUTE 16 19, 73, 22, 74
CANVAS ON

WHILE CHARWAITING()
    x = GETCHAR()
ENDWHILE

STYLE ATTRIBUTE 79
@ 23, 28
?? "Press any key to continue..."

STYLE ATTRIBUTE 16
x = GETCHAR()
CLEAR

ENDPROC
WRITELIB LIBNAME TitleScreen
RELEASE PROCS TitleScreen
;*****
;* END MODULE: x.x TITLESCL()
;*****/

```

```

;*****
;* PROCEDURE: License1()
;* MODULE:
;* PURPOSE: This displays a screen containing a Battelle licensing agreement
;* CALLED BY: MAIN.SC
;* CALLS:
;* INPUT ASSERTION:
;* OUTPUT ASSERTION:
;* GLOBAL VARIABLES:
;* AUTHOR: Sean K. Smith
;* DATE: 9/26/91
;*****/
PROC CLOSED License1()
  USEVARS AutoLib

```

```

  CLEAR
  STYLE ATTRIBUTE 31
  CANVAS OFF
  PAINTCANVAS ATTRIBUTE 31 0,0,24,79

  @0,0
  TEXT

```

This computer software has been developed under sponsorship of the Department of Energy (DOE). Any further distribution by any holder of this software package, its accompanying documentation, or other data therein outside of DOE offices or other DOE contractors, unless otherwise specifically provided for, is prohibited without the approval of the National Energy Software Center (NESC). Requests from outside DOE and its contractors shall be submitted to:

Director
National Energy Software Center,
Argonne National Laboratory,
9700 South Cass Avenue,
Argonne, Illinois 60439.

This material was prepared as an account of work sponsored by an agency of the United States Government. Neither PNL nor the U.S. Government, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights.

ENDTEXT

```

CANVAS ON
WHILE (CHARWAITING())
  x = GETCHAR()
ENDWHILE

```

```

STYLE ATTRIBUTE 79
@ 23,28
?? "Press any key to continue..."
x = GETCHAR()

```

```
ENDPROC
WRITELIB LIBNAME License1
RELEASE PROCS License1
;*****
;* END MODULE: x.x License1()
;*****/
```

```

;*****
;* PROCEDURE: License2()
;* MODULE:
;* PURPOSE: Displays a screen containing a Borland licensing statement.
;* CALLED BY: Main.sc
;* CALLS:
;* INPUT ASSERTION:
;* OUTPUT ASSERTION:
;* GLOBAL VARIABLES:
;* AUTHOR: Sean K. Smith
;* DATE: 9/26/91
;*****/
PROC CLOSED License2()
  USEVARS AutoLib

  CLEAR
  STYLE ATTRIBUTE 31
  CANVAS OFF
  PAINTCANVAS ATTRIBUTE 31 0,0,24,79

  @0,0
  TEXT

  This software package is running under Paradox® Runtime (Copyright 1990,
  Borland International, all rights reserved). By pressing a key to continue,
  you agree to:

  ■ acknowledge that Paradox Runtime is owned by Borland International
    and may not be copied

  ■ look to PNL, the application developer, and not Borland
    International, for any support services required by this
    application

  ■ take no action against Borland International for any damages
    resulting from the use of this application, it being understood
    that Paradox Runtime is provided by Borland International "AS IS"
    and without warranties or liabilities for any damages.

  ENDTXT

  CANVAS ON
  WHILE (CHARWAITING())
    x = GETCHAR()
  ENDWHILE

  STYLE ATTRIBUTE 79
  @ 23,28
  ?? "Press any key to continue..."
  x = GETCHAR()
  CLEAR

; put up working message
  STYLE ATTRIBUTE 79
  @24,59 ?? " ONE MOMENT PLEASE"
  STYLE ATTRIBUTE 79+128
  @24,77 ?? "..."
  CANVAS ON
  CANVAS OFF

```

```
ENDPROC
WRITELIB LIBNAME License2
RELEASE PROCS License2
; /*****
;* END MODULE: x.x License2()
; *****/
```

```

;*****
;* PROCEDURE: ErrorHandler()
;* MODULE:
;* PURPOSE:
;* CALLED BY:
;* CALLS:
;* INPUT ASSERTION:
;* OUTPUT ASSERTION:
;* GLOBAL VARIABLES:
;* AUTHOR: JC Prather
;* DATE: August 1991
;*****/
PROC ErrorHandler()
    PRIVATE ERRORPROC
    CURSOR OFF
    CANVAS OFF
    CLEAR
    PAINTCANVAS FILL " " ATTRIBUTE HIGHLIGHT 0,0,24,79
    STYLE ATTRIBUTE HIGHLIGHT
    @ 0,0 ?? " " + fill("-",78) + " "
    PAINTCANVAS FILL ("|" + fill(" ",78) + "|") ATTRIBUTE HIGHLIGHT 1,0,23,79
    @ 24,0 ?? "L" + fill("-",78) + "J"
    STYLE ATTRIBUTE HIGHLIGHT
    @ 3,1 ?? FORMAT("W78,AC", " ")
    @ 4,1 ?? FORMAT("W78,AC", " ")
    @ 5,1 ?? FORMAT("W78,AC", " ")
    STYLE ATTRIBUTE HIGHLIGHT
    @ 4,26 ?? "ERROR CONDITION ENCOUNTERED"
    @ 8,1 ?? " ERROR MESSAGE:"
    STYLE ATTRIBUTE 31
    @ 9,5 ?? ERRORMESSAGE()
    STYLE ATTRIBUTE HIGHLIGHT
    @ 11,1 ?? " ERROR CODE:"
    STYLE ATTRIBUTE 31
    @ 12,5 ?? ERRORCODE()

    STYLE ATTRIBUTE 78
    @ 14,1 ?? FORMAT("W78,AC", "For assistance, contact: Jonathan C. Prather
    ")
    @ 15,1 ?? FORMAT("W78,AC", " (509) 376-5684
    ")
    @ 16,1 ?? FORMAT("W78,AC", " Battelle
    ")
    @ 17,1 ?? FORMAT("W78,AC", " Pacific Northwest Laboratories,
    Richland WA")

    CANVAS ON
    BEEP SLEEP 200 BEEP SLEEP 200 BEEP

; WRITE SYSEM STATUS INFORMATION TO ERROR LOG FILE
READLIB "UTIL\DEMOUTIL" ErrorPrint
ErrorPrint()

```

```

canvas off
STYLE ATTRIBUTE 112
@ 19,10 ?? " " + fill("-",60) + " "
@ 20,10 ?? " " + fill(" ",60) + " "
@ 21,10 ?? " " + fill(" ",60) + " "
@ 22,10 ?? " " + fill(" ",60) + " "

```

```

@ 23,10 ?? "L" + fill("-",60) + "J"

@ 20,11 ?? FORMAT("W59,AC"," Error file has been created: "+ FILENAM)
@ 21,11 ?? FORMAT("W59,AC","System will now be shut down.")
@ 22,11 ?? FORMAT("W59,AC"," Press any key to continue...")

        canvas on
GC = GETCHAR()

IF SYSMODE() <> "Main" THEN
    RESET
ENDIF
IF ISRUNTIME() = TRUE THEN
    QUIT
ENDIF

PAINTCANVAS FILL " " ATTRIBUTE BKCOLOR 0,0,24,79
RETURN

ENDPROC
WRITELIB LIBNAME ErrorHandler
RELEASE PROCS ErrorHandler
;*****
;* END MODULE: x.x ErrorHandler()
;*****

```

```

; /*****
; * PROCEDURE: ErrorPrint()
; * MODULE:
; * PURPOSE:
; * CALLED BY:
; * CALLS:
; * INPUT ASSERTION:
; * OUTPUT ASSERTION:
; * GLOBAL VARIABLES:
; * AUTHOR: JC Prather
; * DATE: August 1991
; *****/
PROC ErrorPrint()
PRIVATE errorproc, cntr, smode
;
; Private variables:
; errorproc - this is a private variable to prevent infinite loop if there
;             is an error in this procedure
; cntr       - counter variable to determine the file extension for the
;             error log file
; smode      - contains the result of SYSMODE so multiple calls to the
;             function do not have to be made
;
CANVAS ON
CANVAS OFF
IF DRIVESPACE(SUBSTR(DIRECTORY(),1,1)) < 3000 THEN
    CANVAS ON
    BEEP SLEEP 100 BEEP SLEEP 100 BEEP
    MESSAGE "Not enough disk space to write error log, process cancelled!"
    gc = getchar()
ENDIF
cntr = 1
filenam = "ERRLOG.001"           ; Find the name of last log file
WHILE ISFILE(filenam)           ; Find the name of last log file
    cntr = cntr + 1             ; printed
    filenam = "ERRLOG." +
        SUBSTR(FORMAT("w4,ez",cntr),2,3)
ENDWHILE

PRINT FILE filenam "Error log file: ",filenam,"\n\n"
PRINT FILE filenam "Current Date:    ",TODAY(),"\n"
PRINT FILE filenam "Current Time:    ",TIME(),"\n"
PRINT FILE filenam "Error Code:      ",ERRORCODE(),"\n"
PRINT FILE filenam "Error Message:   ",ERRORMESSAGE(),"\n"
PRINT FILE filenam "Error User:      ",ERRORUSER(),"\n"
PRINT FILE filenam "Central Memory:  ",MEMLEFT(),"\n"
PRINT FILE filenam "Code Pool Memory:",RMEMLEFT(),"\n"
PRINT FILE filenam "Menu Choice:    ",MENUCHOICE(),"\n"
IF MENUCHOICE() <> "Error" THEN      ; If cursor is at a menu need to
    MENU ESC                        ; cancel the menu selection
ENDIF
PRINT FILE filenam "Monitor Type:    ",MONITOR(),"\n"
PRINT FILE filenam "Network Type:    ",NETTYPE(),"\n"
PRINT FILE filenam "System Mode:     ",SYSMODE(),"\n"
smode = SYSMODE()
;
; If the process is in modes other than in a table on the workspace
; certain workspace informational functions either cause a script
; error or the return erroneous information. Therefore, the current
; mode needs to be Main, Edit, CoEdit, or DataEntry. If it is not

```

```

; one of these modes then the current mode is cancelled. Some modes
; require the (Cancel) (Yes) options. These are cancelled using
; CANCELEDIT. The other modes only need (Cancel), there is no Yes/No
; menu option.
;
; NOTE: If the current table is in form view, returning from Create,
; Graph, Password, Restructure, and Sort puts the table in
; Table View so the functions below testing for type of
; form will not be a true indicator of the workspace status.
;
IF smode = "Create" OR smode = "Restructure" OR smode = "Sort" THEN
    MENU (Cancel)
ENDIF
IF smode = "Form" OR smode = "Report" OR smode = "Password" OR
    smode = "Graph" THEN
    CANCELEDIT
ENDIF
PRINT FILE filenam "Directory:      ", DIRECTORY(), "\n"
PRINT FILE filenam "Disk space:      ",
    DRIVESPACE(SUBSTR(DIRECTORY(), 1, 1)), "\n"
PRINT FILE filenam "Is Runtime      ", FORMAT("LY", ISRUNTIME()), "\n"
PRINT FILE filenam "Is Blank Zero    ", FORMAT("LY", ISBLANKZERO()), "\n"
PRINT FILE filenam "Private Directory:", PRIVDIR(), "\n"
PRINT FILE filenam "User Name:      ", USERNAME(), "\n"
PRINT FILE filenam "Script Directory:", SDIR(), "\n"
PRINT FILE filenam "Query Order:     ", QUERYORDER(), "\n"
PRINT FILE filenam "Sort Order:      ", SORTORDER(), "\n"
PRINT FILE filenam "Version:        ", VERSION(), "\n"
IF ISASSIGNED(procname) THEN
    PRINT FILE filenam "Cur Proc:      ", procname, "\n"
ELSE
    PRINT FILE filenam "Cur Proc:      ** Not Set **\n"
ENDIF
;
; NOTE: The above is here to save the name of the current PROC being
; executed. This will only work if procname is set at the beginning
; of every procedure and it is a global variable
;
PRINT FILE filenam "# of images:      ", NIMAGES(), "\n"
PRINT FILE filenam "Image Type:      ", IMAGETYPE(), "\n"
IF IMAGETYPE() <> "None" THEN

    PRINT FILE filenam "Current Table:      ", TABLE(), "\n"
    PRINT FILE filenam "Table Shared:      ", FORMAT("LY", ISSHARED(TABLE())), "\n"
    IF IMAGETYPE() = "Display" THEN
        IF ISFORMVIEW() THEN
            PRINT FILE filenam "View Type:          Form\n"
            PRINT FILE filenam "Form #:              ", FORM(), "\n"
            PRINT FILE filenam "MultiTable Form:     ",
                FORMAT("LY", ISMULTIFORM(TABLE(), FORM())), "\n"
            FORMKEY
            ; Put into Table View
        ELSE
            PRINT FILE filenam "View Type:          Table\n"
        ENDIF
    ENDIF
    PRINT FILE filenam "Current Image #:     ", IMAGENO(), "\n"
    PRINT FILE filenam "Current Field:       ", FIELD(), "\n"
    PRINT FILE filenam "Cur View Rcd #:     ", RECNO(), "\n"
    PRINT FILE filenam "Cur Table Rcd #:    ", [#], "\n"
ENDIF
IF NIMAGES() > 0 THEN
    ; Print name and type of each
    ; image on the workspace
    PRINT FILE filenam "Tables placed on the workspace:\n"

```

```

        FOR i from 1 to NIMAGES()
            MOVETO i
            PRINT FILE filenam " Table # ",FORMAT("w3",i)," " ,TABLE(),
                " - ",IMAGETYPE(),"\n"
        ENDFOR
    ENDIF
ENDIF
SAVEVARS ALL                                ; This should be done only for
                                            ; versions 3.0 and above

;*** ADD SAVEVARS.SC TO ERRLOG FILE

PRINT FILE FILENAM "\n","\n","\n"
RESET
{Tools} {ExportImport} {Import} {Ascii} {Text}
TYPEIN PRIVDIR()+"SAVEVARS.SC"
ENTER
{VARTEMP}
VIEW "VARTEMP"
RIGHT
SCAN
TEXTVAR = []
PRINT FILE FILENAM TEXTVAR,"\n"
ENDSCAN
DELETE "VARTEMP"
{TOOLS} {DELETE} {SCRIPT}
TYPEIN PRIVDIR()+"SAVEVARS"
ENTER
{OK}
;***

ENDPROC
WRITELIB LIBNAME ErrorPrint
RELEASE PROCS ErrorPrint
; /*****
; * END MODULE: x.x ErrorPrint()
; *****/

```

```

;*****
;* PROCEDURE: SetPopUp()
;* MODULE:
;* PURPOSE:
;* CALLED BY:
;* CALLS:
;* INPUT ASSERTION:
;* OUTPUT ASSERTION:
;* GLOBAL VARIABLES:
;* AUTHOR: JC Prather
;* DATE: August 1991
;*****/
PROC SetPopUp(POPTBL,FLD)

```

```

; Copyright (c) 1988, 1989 Borland International. All Rights Reserved.
;
; General permission to re-distribute all or part of this script is granted,
; provided that this statement, including the above copyright notice, is not
; removed. You may add your own copyright notice to secure copyright
; protection for new matter that you add to this script, but Borland
; International will not support, nor assume any legal responsibility for,
; material added or changes made to this script.

```

```

; Private;PopTbl,      ;Source table for items of menu
;          ;Fld,        ;Source field for items of menu
; Global ;Item,        ;Array of items of menu
;          ;Width       ;Width of widest item
RELEASE VARS ITEM, WIDTH
IF ISEMPY(POPTBL) THEN BEEP
    WIDTH = 0
    ;DRAW BOX FOR MESSAGE
    STYLE ATTRIBUTE HIGHLIGHT
    PAINTCANVAS ATTRIBUTE BKCOLOR 24,0,24,79
    @ 6,5 ?? " "+FILL(" ",68)+" "
    @ 7,5 ?? " "+FILL(" ",68)+" "
    @ 8,5 ?? " "+FILL(" ",68)+" "
    @ 9,5 ?? " "+FILL(" ",68)+" "
    @ 10,5 ?? " "+FILL(" ",68)+" "
    @ 11,5 ?? " "+FILL(" ",68)+" "
    @ 8,6 ?? FORMAT("W68,AC","NO RECORDS FOUND")
    @ 9,6 ?? FORMAT("W68,AC"," Press any key to continue...")
    CANVAS ON
    CANVAS OFF
    GC=GETCHAR()
    RETURN
ENDIF

    ARRAY ITEM[NRECORDS(POPTBL)] ;Dimension Item array. One item per record
    VIEW POPTBL ; in PopTbl.
    MOVETO FIELD FLD
    WIDTH = 0
    IF SEARCH("A",FIELDTYPE()) = 0 ;If field is non-alphanumeric, convert it
    THEN SCAN ; to a string value before assigning it
        IF LEN(STRVAL([])) > 75 AND !ASSIGNED(TextEditorFLAG)=FALSE THEN
            ITEM[[]] = SUBSTR(STRVAL([]),1,74)
        ELSE ITEM[[]] = STRVAL([])
        ENDIF
        WIDTH = MAX(LEN(ITEM[[]]),WIDTH) ;Update max. width
    ENDSCAN

```

```

ELSE SCAN
  IF LEN( ) > 75 AND ISASSIGNED(TextEditorFLAG)=FALSE THEN
    ITEM[#] = SUBSTR( ,1,74)
    ELSE ITEM[#] = 
  ENDIF
  WIDTH = MAX(LEN( ),WIDTH)
ENDSCAN
ENDIF

IF WIDTH > 74 THEN WIDTH = 76
ENDIF

ENDPROC
WRITELIB LIBNAME SetPopUp
RELEASE PROCS SetPopUp
; /*****
; * END MODULE: x.x SetPopUp()
; *****/

```

```

; /*****
; * PROCEDURE: PopUp()
; * MODULE:
; * PURPOSE:
; * CALLED BY:
; * CALLS:
; * INPUT ASSERTION:
; * OUTPUT ASSERTION:
; * GLOBAL VARIABLES:
; * AUTHOR: JC Prather
; * DATE: August 1991
; * Revised October 29, 1991, SK Smith
; * added 1 space to WIDTH unless using texteditor
; * This is necessary to avoid a crash when user selects an item
; * from a list (F1 option). Selection causes an * to be appended to
; * the line. Some lines became too long to fit in the window.
; *
; *****/

; *****/

PROC PopUp(R,C,VNUM,DEFITEM,TITLE,PROMPT1,PROMPT2)
    PRIVATE R,          ;Row position of upper-left corner of menu box
           C,          ;Column position of upper-left corner menu box
           VNum,       ;Number of items to be displayed in one menu image
           DefItem,    ;Item (number) to show
           Title,      ;Title of PopUp box
           Prompt1,    ;First prompt line
           Prompt2,    ;Second prompt line
           NITEMS,     ;Number of items in menu list
           CHAR,       ;Keycode of last key pressed
           MENUPOS,    ;Current (row image) position within menu
           CINDEX,     ;Current choice index into Item
           X,          ;Counter variable
           PRMPTCOLR,  ;Color attribute for prompt
           BRDRCOLR,   ;Color attribute for box border
           LISTCOLR,   ;Color attribute for menu item list
           SLCTCOLR,   ;Color attribute for current menu selection
           COUNTSTAR

; Global ;Item,      ;Array of items of menu
           ;Width    ;Width of widest item

; INITIALIZE VARIABLES:
COUNTONLYONE = 0
COUNTSTAR = 0

IF ISASSIGNED(ITEM[1])=FALSE THEN
    ESCAPEFLAG = "Y"
    RETURN
ENDIF

; Echo Off          ;Freeze workspace image
CURSOR OFF          ;Hide blinking cursor
CANVAS OFF          ;Disable immediate printing to canvas

PRMPTCOLR =SysColor(2) ;Top two line prompt color
BRDRCOLR =SysColor(23) ;Border color
LISTCOLR =SysColor(2)  ;Menu list color
SLCTCOLR =HIGHLIGHT    ;Current menu selection color

IF ISASSIGNED(TextEditorFLAG) THEN
    LISTCOLR = SysColor(21)

```

```

ENDIF

IF ISASSIGNED(CLINICALFLAG) OR ISASSIGNED(HELPFLAG) THEN
    BRDRCOLR = SysColor(22)
    LISTCOLR = SysColor(21)
    PRMPTCOLR = SysColor(0)
ENDIF

IF MONITOR() <> "Color" THEN
    LISTCOLR = 2
    BRDRCOLR = 112
ENDIF

STYLE ATTRIBUTE PRMPTCOLR
IF ISASSIGNED(CLINICALFLAG)=FALSE
    AND ISASSIGNED(HELPFLAG)=FALSE AND ISASSIGNED(TextEditorFLAG) THEN
    @ 23,0
    ELSE @ 0,0 ;Display prompt information
ENDIF
?? SPACES(80)+PROMPT2+SPACES(80-LEN(PROMPT2))
IF ISASSIGNED(CLINICALFLAG)=FALSE
    AND ISASSIGNED(HELPFLAG)=FALSE AND ISASSIGNED(TextEditorFLAG) THEN
    @ 23,0
    ELSE @ 0,0
ENDIF
?? PROMPT1

WIDTH = MAX(LEN(TITLE),WIDTH) ;Expand box width if title is too wide
; ----- Revised October 29, 1991 -- SKS
IF ( (ISASSIGNED(TextEditorFLAG)) AND (TextEditorFLAG = "Y")) THEN
    /* width is fine */
ELSE
    WIDTH = WIDTH + 1
ENDIF
; ----- end of revision
NITEMS = ARRAYSIZE(ITEM) ;Set number of items in list
IF VNUM > NITEMS
    THEN VNUM = NITEMS
ENDIF

SWITCH
CASE DEFITEM < VNUM : ;Redraw top screen
    REDRAW = 0
    MENUPOS = DEFITEM
CASE DEFITEM > NITEMS-VNUM : ;Redraw last screen
    REDRAW = NITEMS-VNUM
    MENUPOS = DEFITEM-NITEMS+VNUM
OTHERWISE : ;Redraw intermediate screen
    REDRAW = DEFITEM-1 ;Place item at top of menu
    MENUPOS = 1
ENDSWITCH

;Set default menu settings:
CINDEX = DEFITEM ;First menu item
LASTPOS = MENUPOS ;Last image position is current pos.
LASTIDX = CINDEX ;Last menu item index

STYLE ATTRIBUTE BRDRCOLR
SETMARGIN C

@ R,C ;Draw menu skeleton and initial image
?? "┌",FILL("─",WIDTH+2),"┐"

```

```

@ R+1,C ?? " | ",FORMAT("AC,W"+STRVAL(WIDTH),TITLE)," | "
@ R+2,C ?? " | ",FILL("-",WIDTH+2)," | "
FOR X FROM 1 TO VNUM
@ R+2+X,C ?? " | ",SPACES(WIDTH+2)," | "
ENDFOR
@ R+2+VNUM+1,C ?? " | ",FILL("-",WIDTH+2)," | "

@ R+3,C+1
IF REDRAW = 0 ;Records above?
THEN ?? " "
ELSE ?? "^^" ; Yes- Show items are above
ENDIF
@ R+VNUM+2,C+1
IF REDRAW+VNUM = NITEMS ;Records below?
THEN ?? " "
ELSE ?? "^^" ; Yes- Show items are below
ENDIF

@ R+2,C+2
SETMARGIN C+2
STYLE ATTRIBUTE LISTCOLR
FOR X FROM 1 TO VNUM
? ITEM[X+REDRAW]+SPACES(WIDTH-LEN(ITEM[X+REDRAW]))
ENDFOR
SETMARGIN OFF

CANVAS ON ;Reenable immediate echoing to canvas

REDRAW = -1 ;Disable menu image redraw

WHILE TRUE

STYLE ATTRIBUTE SLCTCOLR
@ MENUPOS+R+2,C+2 ;Highlight current selection
?? ITEM[CINDEX]
STYLE ATTRIBUTE LISTCOLR

CHAR = GETCHAR()

SWITCH
CASE CHAR > 31: ;First character search?
IF ISASSIGNED(TextEditorFLAG) THEN RETURN
ENDIF

FOR X FROM 1 TO NITEMS ;Search (down) item array
IF UPPER(SUBSTR(ITEM[X],1,1)) = UPPER(CHR(CHAR))
THEN QUITLOOP ;Found a match
ENDIF
ENDFOR
IF X = NITEMS+1 ;Match not found
THEN BEEP
ELSE IF MENUPOS+X-CINDEX > VNUM OR MENUPOS+X-CINDEX < 1
;Is next item already visible?
THEN SWITCH ; No-
CASE X < VNUM : ;Redraw top screen
REDRAW = 0
MENUPOS = X
CASE X > NITEMS-VNUM : ;Redraw last screen
REDRAW = NITEMS-VNUM
MENUPOS = X-NITEMS+VNUM
OTHERWISE : ;Redraw intermediate screen
REDRAW = X-1 ;Place item at top of menu

```

```

        MENUPOS = 1
    ENDSWITCH
    ELSE MENUPOS = MENUPOS+X-CINDEX
    REDRAW = -1
ENDIF
CINDEX = X ;Update current item
ENDIF
CASE CHAR = -72 : ;Up
    IF CINDEX = 1 ;Already at first item?
        THEN BEEP
    ELSE IF MENUPOS > 1 ;Can move within menu image?
        THEN MENUPOS = MENUPOS-1 ; Yes- Move to previous item
        ELSE REDRAW = CINDEX-2 ; No- Redraw entire menu
    ENDIF
    CINDEX = CINDEX-1 ;Update current item
ENDIF
CASE CHAR = -80 : ;Down
    IF CINDEX = NITEMS ;On last item?
        THEN BEEP
        LOOP
    ELSE IF MENUPOS < VNUM ;Can move within menu image?
        THEN MENUPOS = MENUPOS+1 ; Yes- Move to next item
        ELSE REDRAW = CINDEX-VNUM+1 ; No- Redraw entire menu
    ENDIF
    CINDEX=CINDEX+1 ;Update current item
ENDIF
CASE CHAR = -71 : ;Home
    IF MENUPOS <> CINDEX ;Already viewing top of menu?
        THEN REDRAW = 0 ; No- Redraw top of menu
    ENDIF
    MENUPOS = 1 ;Position at first item
    CINDEX = 1 ;Select first item
CASE CHAR = -79 : ;End
    IF CINDEX+VNUM-MENUPOS <> NITEMS ;Already viewing end of menu?
        THEN REDRAW = NITEMS-VNUM ; No- Redraw end of menu
    ENDIF
    MENUPOS = VNUM ;Position at bottom of menu
    CINDEX = NITEMS ;Select last item
CASE CHAR = -73 : ;PgUp
    IF MENUPOS = CINDEX ;Are we within first screen?
        THEN BEEP ; Yes- Disallow PgUp
    ELSE IF CINDEX-MENUPOS-VNUM > 0
        THEN CINDEX = CINDEX-MENUPOS-VNUM+1
        ELSE CINDEX = 1
    ENDIF
    REDRAW = CINDEX-1 ; No- Redraw previous page
    MENUPOS = 1 ; Position on that item
ENDIF
CASE CHAR = -81 : ;PgDn
    IF CINDEX+VNUM-MENUPOS = NITEMS ;Are we within last screen?
        THEN BEEP ; Yes- Disallow PgDn
    ELSE IF NITEMS-VNUM < CINDEX+VNUM-MENUPOS
        THEN CINDEX = NITEMS-VNUM+1
        ELSE CINDEX = CINDEX+VNUM-MENUPOS+1
    ENDIF
    REDRAW = CINDEX-1 ; No- Redraw next page
    MENUPOS = 1 ; Position on that item
ENDIF
CASE CHAR = 13 : ;Enter
    IF ISASSIGNED(TextEditorFLAG) THEN RETURN
ENDIF

```

```

IF ISASSIGNED(ONLYONEFLAG) THEN
  IF ONLYONEFLAG = "Y" THEN
    IF SUBSTR(ITEM[CINDEX],1,1) = "*" THEN
      ITEM[CINDEX] = SUBSTR(ITEM[CINDEX],2,LEN(ITEM[CINDEX])-1)
      @ MENUPOS+R+2,C+2 ?? ITEM[CINDEX]+" "
      COUNTONLYONE = 0
    ELSE IF COUNTONLYONE <> 1 THEN
      ITEM[CINDEX] = "*" + ITEM[CINDEX]
      COUNTONLYONE = 1
    ELSE BEEP
    ENDIF
  ENDIF
  LOOP
ENDIF
ENDIF

IF ISASSIGNED(TextEditorFLAG) THEN
  IF TextEditorFLAG = "Y" THEN
    LOOP
  ENDIF
ENDIF

IF SUBSTR(ITEM[CINDEX],1,1) = "*" THEN
  ITEM[CINDEX] = SUBSTR(ITEM[CINDEX],2,LEN(ITEM[CINDEX])-1)
  @ MENUPOS+R+2,C+2 ?? ITEM[CINDEX]+" "
  COUNTSTAR = COUNTSTAR - 1
ELSE
  IF COUNTSTAR < 21 THEN
    ITEM[CINDEX] = "*" + ITEM[CINDEX]
    COUNTSTAR = COUNTSTAR + 1
  ELSE BEEP
  ENDIF
ENDIF
CASE CHAR = -64: ;F6 PRINT
  IF ISASSIGNED(TextEditorFLAG) THEN
    PrintProc("W\TEMP","1")
    RETURN
  ELSE BEEP
  ENDIF

CASE CHAR = 27 : ;Esc
  STYLE
  ESCAPEFLAG = "Y"
  RETURN ;Return null selection
CASE CHAR = -60 : ;F2
  PAINTCANVAS ATTRIBUTE BKCOLOR 0,0,2,79
  RETURN
OTHERWISE: ;Illegal key
  BEEP
ENDSWITCH

IF REDRAW = -1 ;Need to redraw entire menu?
THEN @ LASTPOS+R+2,C+2
  ?? ITEM[LASTIDX]
ELSE CANVAS OFF ;Disable immediate canvas echo
  SETMARGIN C+2
  @ R+2,C+2
  FOR X FROM 1 TO VNUM ;Redraw entire menu box
    ? ITEM[REDRAW+X]+SPACES(WIDTH-LEN(ITEM[REDRAW+X]))
  ENDFOR
  SETMARGIN OFF
  @ R+3,C+1

```

```

        STYLE ATTRIBUTE BRDRCOLR
        IF REDRAW = 0
            THEN ?? " "
            ELSE ?? "^^"
        ENDIF
        @ R+VNUM+2,C+1
        IF REDRAW+VNUM = NITEMS
            THEN ?? " "
            ELSE ?? " ^ "
        ENDIF
        CANVAS ON
        REDRAW = -1
    ENDIF

    LASTPOS = MENUPOS
    LASTIDX = CINDEX

    ENDWHILE

ENDPROC
WRITELIB LIBNAME PopUp
RELEASE PROCS PopUp
;*****
;* END MODULE: x.x PopUp()
;*****/

```

```

;*****
;* PROCEDURE: PrintProc()
;* MODULE:
;* PURPOSE:
;* CALLED BY:
;* CALLS:
;* INPUT ASSERTION:
;* OUTPUT ASSERTION:
;* GLOBAL VARIABLES:
;* AUTHOR: JC Prather
;* DATE: August 1991
;*****/
PROC PrintProc(TBL,RPT)
    PRIVATE ANS,FILENAM

;init variables
ANS = ""
IF ISASSIGNED(TextEditorFLAG) THEN
    WHATITIS = "text"
ELSE WHATITIS = "report"
ENDIF

IF ISASSIGNED(HELPFLAG) = FALSE AND ISASSIGNED(CLINICALFLAG) = FALSE THEN
    .****remove menu instructions and maintitle
    PAINTCANVAS ATTRIBUTE BKCOLOR 0,0,3,79
ELSE PAINTCANVAS ATTRIBUTE BKCOLOR 0,0,1,79
ENDIF

SHOWMENU "FILE":"Send "+WHATITIS+" to file...",
        "PRINTER":"Send "+WHATITIS+" to printer..."
TO ANS

IF ANS <> "SCREEN" AND ANS <> "FILE" AND ANS <> "PRINTER" THEN RETURN
ENDIF

        CANVAS OFF
        STYLE ATTRIBUTE HIGHLIGHT

IF ANS <> "FILE" THEN
    STYLE ATTRIBUTE 78
    @ 0,0 ??FORMAT("W80,AC","Sending "+WHATITIS+" to "+ANS+"...")
ENDIF

        CANVAS ON

SWITCH
CASE ANS = "SCREEN":

    MENU {REPORT} {OUTPUT}
    TYPEIN TBL
    ENTER
    SELECT RPT

    {SCREEN}

```

```

CASE ANS = "FILE":

    ;init vars
    FILENAM = ""
    STYLE ATTRIBUTE HIGHLIGHT
    @ 0,0 ?? FORMAT("W80,AL","Enter file name: ")
    @ 1,0 ?? FORMAT("W80,AL","File extention will be .RPT")
    STYLE ATTRIBUTE 31
    CURSOR NORMAL

    @ 0,17
    ACCEPT "A30" TO FILENAM

; check for validity
WHILE (LEN(FILENAM) > 8 AND SEARCH("\\",FILENAM)=0) AND
      (SEARCH(".",FILENAM)<LEN(FILENAM)-3)
    @ 0,17
    ACCEPT "A30" TO FILENAM
ENDWHILE

CURSOR OFF

IF FILENAM = "" THEN RETURN
ENDIF

IF SEARCH(".",FILENAM) = 0 THEN
    FILENAM = FILENAM + ".RPT"
ENDIF
CURSOR OFF
TYPEIN FILENAM
ENTER

IF ISFILE(FILENAM) THEN
    STYLE ATTRIBUTE 78
    FILEANS = ""
    SHOWMENU "CANCEL":"Do not overwrite existing "+FILENAM+", cancel print...",
             "OVERWRITE":"Overwrite existing "+FILENAM+" with "+WHATITIS+"..."
    TO FILEANS
    IF FILEANS <> "OVERWRITE" THEN
        RETURN
    ENDIF
    PAINTCANVAS FILL " " ATTRIBUTE 79 0,0,1,79
    STYLE ATTRIBUTE 78
    @ 0,0?? FORMAT("W80,AC","Sending "+WHATITIS+" to file: "+FILENAM)
    MENU {REPORT} {OUTPUT}
    TYPEIN TBL
    ENTER
    SELECT RPT
    {FILE}
    TYPEIN FILENAM
    ENTER
    {REPLACE}
    SLEEP 3000
    RETURN
ENDIF

IF FILENAM = ".RPT" OR FILENAM = "ESC.." THEN
    RETURN
ENDIF
STYLE ATTRIBUTE 78

```

```
PAINTCANVAS FILL " " ATTRIBUTE HIGHLIGHT 0,0,1,79
@ 0,0 ?? FORMAT("W80,AC","Sending "+WHATITIS+" to file: "+FILENAM)
```

```
MENU {REPORT} {OUTPUT}
TYPEIN TBL
ENTER
SELECT RPT
{FILE}
TYPEIN FILENAM
ENTER
```

```
CASE ANS = "PRINTER":
```

```
MENU {REPORT} {OUTPUT}
TYPEIN TBL
ENTER
SELECT RPT
{PRINTER}
```

```
ENDSWITCH
```

```
SLEEP 1000
RETURN
```

```
ENDPROC
```

```
WRITELIB LIBNAME PrintProc
```

```
RELEASE PROCS PrintProc
```

```
;/*****
;* END MODULE: x.x PrintProc()
;*****/
```

```

; /*****
; * PROCEDURE: TextEditor()
; * MODULE:
; * PURPOSE:
; * CALLED BY:
; * CALLS:
; * INPUT ASSERTION:
; * OUTPUT ASSERTION:
; * GLOBAL VARIABLES:
; * AUTHOR: JC Prather
; * DATE: August 1991
; *****/
PROC TextEditor(DOFILE,TITLE)
    PRIVATE TextEditorFLAG,TEMPSTR,ADDFLAG,HOLDOVER,C,LASTREC
    TextEditorFLAG = "Y"

    style attribute highlight
    IF ISASSIGNED(CLINICALFLAG) OR ISASSIGNED(HELPFLAG) THEN @ 0,59
        ELSE @24,59
    ENDIF
    ?? " ONE MOMENT PLEASE"
    style attribute highlight+128
    IF ISASSIGNED(CLINICALFLAG) OR ISASSIGNED(HELPFLAG) THEN @ 0,77
        ELSE @24,77
    ENDIF
    ?? "... "

    canvas off

    if istable("W\\temp")
        then {Tools} {ExportImport} {Import} {Ascii} {Text} typein dofile enter
            {W\\temp} {replace}
        else {Tools} {ExportImport} {Import} {Ascii} {Text} typein dofile enter
            {W\\temp}
        endif

    copyreport "W\\text" "1" "W\\temp" "1"

    LASTREC = NRECORDS("W\\TEMP")

    EDIT "W\\TEMP" ; in PopTbl.
    MOVE TO FIELD "TEXT"
    HOLDOVER = ""
    WHILE TRUE ; to a string value before assigning it
        IF [] = "" THEN [] = " "
        ENDIF

        TEMPSTR = HOLDOVER + STRVAL([])

        IF LEN(TEMPSTR) > 76 THEN ;make sure not larger than box
            FOR X FROM 1 TO 76
                IF SUBSTR(TEMPSTR,76-X,1) = " " THEN
                    HOLDOVER = SUBSTR(TEMPSTR,76-X+1,LEN(TEMPSTR))
                    IF SUBSTR(HOLDOVER,LEN(HOLDOVER),1)<>" " THEN HOLDOVER=HOLDOVER+" "
                ENDIF
                TEMPSTR = SUBSTR(TEMPSTR,1,76-X)
            QUITLOOP
        ENDIF
    ENDFOR
    ELSE HOLDOVER = ""

```

```

ENDIF

IF ( [] = "" OR [] = " ") AND (TEMPSTR<>"" AND TEMPSTR<>" ") THEN
    ADDFLAG = "Y"
ELSE ADDFLAG = "N"
ENDIF
IF TEMPSTR = "" THEN TEMPSTR = " "
ENDIF

[] = TEMPSTR

IF ADDFLAG= "Y" THEN
    DOWN
    INS
    UP
ENDIF

DOWN

IF RECNO() > LASTREC AND TEMPSTR="" AND [] = "" AND HOLDOVER = "" THEN
    QUITLOOP
ENDIF

ENDWHILE

DO_IT!
;*****

SetPopUp("W\\TEMP","TEXT")

PROMPT2 = FORMAT("W80,AC","HOME,END,"+CHR(24)+",""+CHR(25)+",PGUP,PGDN-Scroll text F6-Print
text")
C = MAX(INT((80-WIDTH)/2)-2,0)

; CHUCK WANTS TO HIT ANY KEY TO GET OUT
PROMPT1 = PROMPT2
IF ISASSIGNED(HELPFLAG) THEN
    PROMPT2 = FORMAT("W80,AC","Press any other key to exit HELP")
ELSE IF ISASSIGNED(CLINICALFLAG) THEN
    PROMPT2 = FORMAT("W80,AC","Press any other key to return to Summary browse")
ELSE PROMPT2 = FORMAT("W80,AC","Press any other key to return to menu")
ENDIF
ENDIF

IF ISASSIGNED(CLINICALFLAG) THEN
    PopUp(7,C,14,1,TITLE,PROMPT1,PROMPT2)
ELSE PopUp(4,C,14,1,TITLE,PROMPT1,PROMPT2)
ENDIF

DELETE "W\\TEMP"

RELEASE VARS ITEM,WIDTH

ENDPROC
WRTLIB LIBNAME TextEditor
RELEASE PROCS TextEditor
;*****
;* END MODULE: x.x TextEditor()
;*****/

```

```

; /*****
; * PROCEDURE: SearchBo~()
; * MODULE:
; * PURPOSE:
; * CALLED BY:
; * CALLS:
; * INPUT ASSERTION:
; * OUTPUT ASSERTION:
; * GLOBAL VARIABLES:
; * AUTHOR: JC Prather
; * DATE: August 1991
; *****/
PROC SearchBox(SEARCHTITLE, SEARCHPROMPT, SEARCHWIDTH)
    PRIVATE SEARCHVAL, MESSAGE1, MESSAGE2, BOXWIDTH, STARTCOLUMN, THECOLUMN, TITLECOLUMN,
        PROMPTCOLUMN, MESSAGE1COLUMN, MESSAGE2COLUMN, THEROW, NEXTCHAR

    PAINTCANVAS ATTRIBUTE BKCOLOR 24,0,24,79

    SEARCHVAL = ""
    MESSAGE1="Type in search string (wildcard = ..)"
    MESSAGE2="[F1]-view list of choices      [F2],[ENTER]-execute search"
    BOXWIDTH = MAX(65,MAX((SEARCHWIDTH)+6,LEN(SEARCHTITLE)+2))
    STARTCOLUMN = INT((80-BOXWIDTH)/2)

    THECOLUMN = STARTCOLUMN+INT((BOXWIDTH-SEARCHWIDTH)/2)
    TITLECOLUMN = STARTCOLUMN+INT((BOXWIDTH-LEN(SEARCHTITLE))/2)
    PROMPTCOLUMN = STARTCOLUMN+INT((BOXWIDTH-LEN(SEARCHPROMPT))/2)
    MESSAGE1COLUMN = STARTCOLUMN+INT((BOXWIDTH-LEN(MESSAGE1))/2)
    MESSAGE2COLUMN = STARTCOLUMN+INT((BOXWIDTH-LEN(MESSAGE2))/2)
    THEROW = 8

    STYLE ATTRIBUTE HIGHLIGHT
    @ 6,STARTCOLUMN ?? " "+FILL("-",BOXWIDTH-2)+" "
    @ 7,STARTCOLUMN ?? " "+FILL(" ",BOXWIDTH-2)+" "
    @ 8,STARTCOLUMN ?? " "+FILL(" ",BOXWIDTH-2)+" "
    @ 9,STARTCOLUMN ?? " "+FILL(" ",BOXWIDTH-2)+" "
    @ 10,STARTCOLUMN ?? " "+FILL(" ",BOXWIDTH-2)+" "
    @ 11,STARTCOLUMN ?? " "+FILL("-",BOXWIDTH-2)+" "
    @ 6,TITLECOLUMN ?? SEARCHTITLE
    @ 7,PROMPTCOLUMN ?? SEARCHPROMPT
    @ 9,MESSAGE1COLUMN ?? MESSAGE1
    @ 10,MESSAGE2COLUMN ?? MESSAGE2
    STYLE ATTRIBUTE 31
    @ 8,THECOLUMN ?? FILL(" ",SEARCHWIDTH)

    WHILE TRUE
        @ THEROW,THECOLUMN ?? SEARCHVAL
        CURSOR NORMAL
        NEXTCHAR = GETCHAR()

        SWITCH
            CASE NEXTCHAR = 27 :      ;ESC
                RETURN ""

            CASE NEXTCHAR = -59 :      ;F1
                RETURN "F1"

            CASE NEXTCHAR = -60 OR NEXTCHAR = 13:      ;F2,ENTER

```

```

        RETURN SEARCHVAL

CASE 31 < NEXTCHAR AND NEXTCHAR < 127:
    IF LEN(SEARCHVAL) < SEARCHWIDTH THEN
        SEARCHVAL = SEARCHVAL + CHR(NEXTCHAR)
    ELSE BEEP
    ENDIF

CASE NEXTCHAR = 8 OR NEXTCHAR = -75:      ;BKSPACE OR LEFT
    SEARCHVAL = SUBSTR(SEARCHVAL,1,LEN(SEARCHVAL)-1)
    @ THEROW,THECOLUMN ?? SEARCHVAL+" "

ENDSWITCH
ENDWHILE

ENDPROC
WRITELIB LIBNAME SearchBox
RELEASE PROCS SearchBox
; /*****
;* END MODULE: x.x SearchBox()
; *****/

```

```

;*****
;* PROCEDURE: AuthorSearch()
;* MODULE:
;* PURPOSE:
;* CALLED BY:
;* CALLS:
;* INPUT ASSERTION:
;* OUTPUT ASSERTION:
;* GLOBAL VARIABLES:
;* AUTHOR: JC Prather
;* DATE: August 1991
;*****/
PROC AuthorSearch()
  PRIVATE SEARCHTITLE,SEARCHPROMPT,SEARCHWIDTH,SEARCHVAL,X,L,J,SHOWPRETTY1,
    SHOWPRETTY2,GC,C

  SEARCHTITLE = "BIBLIOGRAPHY SEARCH BY AUTHOR"
  SEARCHPROMPT = "AUTHOR'S LAST NAME:"
  SEARCHWIDTH = 18

  WHILE TRUE
    SEARCHVAL = SearchBox(SEARCHTITLE,SEARCHPROMPT,SEARCHWIDTH)

    CURSOR OFF
    SWITCH
      CASE SEARCHVAL = "" OR SEARCHVAL = " ": RETURN

      CASE SEARCHVAL = "F1": ;GET PopUp OF CHOICES
        ;***** put up working message
        STYLE ATTRIBUTE HIGHLIGHT
        @ 24,59 ?? " ONE MOMENT PLEASE"
        STYLE ATTRIBUTE HIGHLIGHT+128
        @ 24,77 ?? "... "
        CANVAS ON
        CANVAS OFF
        ;****remove menu instructions and maintitle
        PAINTCANVAS ATTRIBUTE BKCOLOR 0,0,3,79
        PAINTCANVAS ATTRIBUTE BKCOLOR 24,0,24,79
        CLEARALL
        ;initialize vars
        ESCAPEFLAG = "N"
        PRMPT1 = FORMAT("W80,AC","F2-Execute search for items selected
          Esc-Return to menu")
        PRMPT2 = FORMAT("W80,AC","ENTER-Select or un-select item          CURSOR
          KEYS,PGUP,PGDN,HOME,END-Scroll list")
        ;GET POSSIBLE CHOICES TO ANSWER FOR PopUp
        (ASK)
        TYPEIN "B\D\AUTHOR"
        ENTER
        MOVE TO FIELD "LAST_NAME"
        CHECK
        DO_IT!
        ;DO PopUp
        SetPopUp("ANSWER","LAST_NAME")
        PAINTCANVAS FILL " " ATTRIBUTE BKCOLOR 24,0,24,79
        C=MAX(INT((80-WIDTH)/2)-3,0)
        PopUp(4,C,14,1,"AUTHOR'S LAST NAME",PRMPT1,PRMPT2)
        IF ESCAPEFLAG = "Y" THEN
          RELEASE VARS ESCAPEFLAG,ITEM,WIDTH
          RETURN
        ELSE RELEASE VARS ESCAPEFLAG

```

```

ENDIF
;****DO SEARCH:
;***** put up working message
STYLE ATTRIBUTE HIGHLIGHT
@ 24,48 ?? " SEARCHING, ONE MOMENT PLEASE"
STYLE ATTRIBUTE HIGHLIGHT+128
@ 24,77 ?? "...
CLEARALL
CANVAS ON
CANVAS OFF
CURSOR OFF
;***
(ASK)
TYPEIN "B\\D\\AUTHOR"
ENTER
L=0
FOR X FROM 1 TO ARRAYSIZE(ITEM)
  IF SUBSTR(ITEM[X],1,1) = "" THEN
    MOVETO FIELD "BAR_CODE"
    EXAMPLE
    L = L + 1
    TYPEIN "A"+STRVAL(L)
    MOVETO FIELD "LAST_NAME"
    TYPEIN SUBSTR(ITEM[X],2,LEN(ITEM[X])-1)
    DOWN
  ENDIF
ENDFOR
UP
IF ISBLANK( ) THEN RETURN
ENDIF
(ASK)
TYPEIN "B\\D\\BOOK"
ENTER
FOR J FROM 1 TO L
  CHECK
  MOVETO FIELD "BAR_CODE"
  EXAMPLE
  TYPEIN "A"+STRVAL(J)
  MOVETO FIELD "#"
  DOWN
ENDFOR
DO_IT!
QUITLOOP

OTHERWISE: ;SEARCH ON SEARCHVAL
;***** put up working message
STYLE ATTRIBUTE HIGHLIGHT
@ 24,48 ?? " SEARCHING, ONE MOMENT PLEASE"
STYLE ATTRIBUTE HIGHLIGHT+128
@ 24,77 ?? "...
CLEARALL
CANVAS ON
CANVAS OFF
CURSOR OFF
(ASK)
;***
TYPEIN "B\\D\\AUTHOR"
ENTER
MOVETO FIELD "BAR_CODE"
EXAMPLE
TYPEIN "A"
MOVETO FIELD "LAST_NAME"

```

```

        TYPEIN SEARCHVAL
        (ASK)
        TYPEIN "B\\D\\BOOK"
        ENTER
        CHECK
        MOVETO FIELD "BAR_CODE"
        EXAMPLE
        TYPEIN "A"
        DO_IT!
        QUITLOOP

ENDSWITCH

ENDWHILE

;SHOW HOW MANY FOUND

STYLE ATTRIBUTE HIGHLIGHT
SHOWPRETTY1 = "MATCH"
SHOWPRETTY2 = "RECORD"

IF NRECORDS("ANSWER") > 1 THEN
    SHOWPRETTY1 = "MATCHES"
    SHOWPRETTY2 = "RECORDS"
ENDIF

;DRAW BOX FOR MESSAGE
PAINTCANVAS ATTRIBUTE BKCOLOR 24,0,24,79
@ 6,5 ?? " "+FILL("-",68)+" "
@ 7,5 ?? " "+FILL(" ",68)+" "
@ 8,5 ?? " "+FILL(" ",68)+" "
@ 9,5 ?? " "+FILL(" ",68)+" "
@ 10,5 ?? " "+FILL(" ",68)+" "
@ 11,5 ?? " "+FILL("-",68)+" "
CANVAS ON

IF IEMPTY("ANSWER") THEN
    @ 8,6 ?? FORMAT("W68,AC","NO MATCHES FOUND")
    @ 9,6 ?? FORMAT("W68,AC","  Press any key to continue...")
    GC = GETCHAR()
    RETURN
ELSE @ 8,6 ?? FORMAT("W68,AC",STRVAL(NRECORDS("ANSWER"))+" "+SHOWPRETTY1+" FOUND")
    @ 9,6 ?? FORMAT("W68,AC","Press any key to view "+SHOWPRETTY2)
    GC = GETCHAR()
ENDIF

;VIEW RECORDS FOUND
CLEARALL
RENAME "ANSWER" "TBOOK"
COPYFORM "B\\D\\BOOK" "2" "TBOOK" "2"
MasterView("TBOOK","2")
DELETE "TBOOK"
CLEARALL

ENDPROC
WRITELIB LIBNAME AuthorSearch

```

```
RELEASE PROCS AuthorSearch
; /*****
; * END MODULE: x.x AuthorSearch()
; *****/
```

```

; /*****
; * PROCEDURE: TitleSearch()
; * MODULE:
; * PURPOSE:
; * CALLED BY:
; * CALLS:
; * INPUT ASSERTION:
; * OUTPUT ASSERTION:
; * GLOBAL VARIABLES:
; * AUTHOR: JC Prather
; * DATE: August 1991
; *****/
PROC TitleSearch()
    PRIVATE SEARCHTITLE, SEARCHPROMPT, SEARCHWIDTH, SEARCHVAL, X, L, J, SHOWPRETTY1,
        SHOWPRETTY2, GC

    SEARCHTITLE = "BIBLIOGRAPHY SEARCH BY TITLE"
    SEARCHPROMPT = "TITLE:"
    SEARCHWIDTH = 65

    WHILE TRUE
        SEARCHVAL = SearchBox(SEARCHTITLE, SEARCHPROMPT, SEARCHWIDTH)

        CURSOR OFF
        SWITCH
            CASE SEARCHVAL = "" OR SEARCHVAL = " ": RETURN

            CASE SEARCHVAL = "F1": ;GET PopUp OF CHOICES
                ;***** put up working message
                STYLE ATTRIBUTE HIGHLIGHT
                @ 24,59 ?? " ONE MOMENT PLEASE"
                STYLE ATTRIBUTE HIGHLIGHT+128
                @ 24,77 ?? "... "
                CANVAS ON
                CANVAS OFF
                ;****remove menu instructions and maintitle
                PAINTCANVAS ATTRIBUTE BKCOLOR 0,0,3,79
                PAINTCANVAS ATTRIBUTE BKCOLOR 24,0,24,79
                CLEARALL
                ;initialize vars
                ESCAPEFLAG = "N"
                PRMPT1 = FORMAT("W80,AC","F2-Execute search for items selected
                    Esc-Return to menu")
                PRMPT2 = FORMAT("W80,AC","ENTER-Select or un-select item          CURSOR
KEYS,PGUP,PGDN,HOME,END-Scroll list")
                ;GET POSSIBLE CHOICES TO ANSWER FOR PopUp
                {ASK}
                TYPEIN "B\D\BOOK"
                ENTER
                MOVETO FIELD "TITLE"
                CHECK
                DO IT!
                ;DO PopUp
                SetPopUp("ANSWER","TITLE")
                PAINTCANVAS FILL " " ATTRIBUTE BKCOLOR 24,0,24,79
                C=MAX(INT((80-WIDTH)/2)-3,0)
                PopUp(4,C,13,1,"TITLE",PRMPT1,PRMPT2)
                IF ESCAPEFLAG = "Y" THEN
                    RELEASE VARS ESCAPEFLAG,ITEM,WIDTH
                    RETURN
                ELSE RELEASE VARS ESCAPEFLAG

```

```

ENDIF
;****DO SEARCH:
;***** put up working message
STYLE ATTRIBUTE HIGHLIGHT
@ 24,48 ?? " SEARCHING, ONE MOMENT PLEASE"
STYLE ATTRIBUTE HIGHLIGHT+128
@ 24,77 ?? "... "
CLEARALL
CANVAS ON
CANVAS OFF
CURSOR OFF
;***
(ASK)
TYPEIN "B\\D\\BOOK"
ENTER
L=0
FOR X FROM 1 TO ARRAYSIZE(ITEM)
  IF SUBSTR(ITEM[X],1,1) = "***" THEN
    MOVETO FIELD "BAR_CODE"
    LEFT
    CHECK
    MOVETO FIELD "TITLE"
    TYPEIN SUBSTR(ITEM[X],2,LEN(ITEM[X])-1)+"... "
    DOWN
  ENDIF
ENDFOR
UP
IF ISBLANK( ) THEN RETURN
ENDIF
DO_IT!
QUITLOOP

OTHERWISE: ;SEARCH ON SEARCHVAL
;***** put up working message
STYLE ATTRIBUTE HIGHLIGHT
@ 24,48 ?? " SEARCHING, ONE MOMENT PLEASE"
STYLE ATTRIBUTE HIGHLIGHT+128
@ 24,77 ?? "... "
CLEARALL
CANVAS ON
CANVAS OFF
CURSOR OFF
(ASK)
;***
TYPEIN "B\\D\\BOOK"
ENTER
CHECK
MOVETO FIELD "TITLE"
TYPEIN SEARCHVAL
DO_IT!
QUITLOOP

ENDSWITCH

ENDWHILE

;SHOW HOW MANY FOUND

```

```

STYLE ATTRIBUTE HIGHLIGHT
SHOWPRETTY1 = "MATCH"
SHOWPRETTY2 = "RECORD"

IF NRECORDS("ANSWER") > 1 THEN
    SHOWPRETTY1 = "MATCHES"
    SHOWPRETTY2 = "RECORDS"
ENDIF

;DRAW BOX FOR MESSAGE
PAINTCANVAS ATTRIBUTE BKCOLOR 24,0,24,79
@ 6,1 ?? " "+FILL(" ",76)+" "
@ 7,1 ?? " "+FILL(" ",76)+" "
@ 8,1 ?? " "+FILL(" ",76)+" "
@ 9,1 ?? " "+FILL(" ",76)+" "
@ 10,1 ?? " "+FILL(" ",76)+" "
@ 11,1 ?? " "+FILL(" ",76)+" "
CANVAS ON

IF ISEMPY("ANSWER") THEN
    @ 8,6 ?? FORMAT("W68,AC","NO MATCHES FOUND")
    @ 9,6 ?? FORMAT("W68,AC"," Press any key to continue...")
    GC = GETCHAR()
    RETURN
ELSE @ 8,6 ?? FORMAT("W68,AC",STRVAL(NRECORDS("ANSWER"))+" "+SHOWPRETTY1+" FOUND")
    @ 9,6 ?? FORMAT("W68,AC","Press any key to view "+SHOWPRETTY2)
    GC = GETCHAR()
ENDIF

;VIEW RECORDS FOUND
CLEARALL
RENAME "ANSWER" "TBOOK"
COPYFORM "B\D\BOOK" "2" "TBOOK" "2"
MasterView("TBOOK","2")
DELETE "TBOOK"
CLEARALL

ENDPROC
WRITELIB LIBNAME TitleSearch
RELEASE PROCS TitleSearch
;*****
;* END MODULE: x.x TitleSearch()
;*****

```

```

;*****
;* PROCEDURE: YearSearch()
;* MODULE:
;* PURPOSE:
;* CALLED BY:
;* CALLS:
;* INPUT ASSERTION:
;* OUTPUT ASSERTION:
;* GLOBAL VARIABLES:
;* AUTHOR: JC Prather
;* DATE: August 1991
;*****/
PROC YearSearch()
    PRIVATE SEARCHTITLE,SEARCHPROMPT,SEARCHWIDTH,SEARCHVAL,X,L,J,SHOWPRETTY1,
        SHOWPRETTY2,GC

    SEARCHTITLE = "BIBLIOGRAPHY SEARCH BY YEAR OF PUBLICATION"
    SEARCHPROMPT = "YEAR:"
    SEARCHWIDTH = 8

    WHILE TRUE
        SEARCHVAL = SearchBox(SEARCHTITLE,SEARCHPROMPT,SEARCHWIDTH)

        CURSOR OFF
        SWITCH
            CASE SEARCHVAL = "" OR SEARCHVAL = " ": RETURN

            CASE SEARCHVAL = "F1": ;GET PopUp OF CHOICES
                ;***** put up working message
                STYLE ATTRIBUTE HIGHLIGHT
                @ 24,59 ?? " ONE MOMENT PLEASE"
                STYLE ATTRIBUTE HIGHLIGHT+128
                @ 24,77 ?? "... "
                CANVAS ON
                CANVAS OFF
                ;****remove menu instructions and maintitle
                PAINTCANVAS ATTRIBUTE BKCOLOR 0,0,3,79
                PAINTCANVAS ATTRIBUTE BKCOLOR 24,0,24,79
                CLEARALL
                ;initialize vars
                ESCAPEFLAG = "N"
                PRMPT1 = FORMAT("W80,AC","F2-Execute search for items selected
                    Esc-Return to menu")
                PRMPT2 = FORMAT("W80,AC","ENTER-Select or un-select item          CURSOR
                    KEYS,PGUP,PGDN,HOME,END-Scroll list")
                ;GET POSSIBLE CHOICES TO ANSWER FOR PopUp
                {ASK}
                TYPEIN "B\D\BOOK"
                ENTER
                MOVETO FIELD "PUB_YR"
                CHECK
                DO_IT!
                ;DO PopUp
                SetPopUp("ANSWER","PUB_YR")
                PAINTCANVAS FILL " " ATTRIBUTE BKCOLOR 24,0,24,79
                C=MAX(INT((80-WIDTH)/2)-3,0)
                PopUp(4,C,13,1,"YEAR OF PUBLICATION",PRMPT1,PRMPT2)
                IF ESCAPEFLAG = "Y" THEN
                    RELEASE VARS ESCAPEFLAG,ITEM,WIDTH
                    RETURN
                ELSE RELEASE VARS ESCAPEFLAG

```

```

ENDIF
;****DO SEARCH:
;***** put up working message
STYLE ATTRIBUTE HIGHLIGHT
@ 24,48 ?? " SEARCHING, ONE MOMENT PLEASE"
STYLE ATTRIBUTE HIGHLIGHT+128
@ 24,77 ?? "...
CLEARALL
CANVAS ON
CANVAS OFF
CURSOR OFF
;***
{ASK}
TYPEIN "B\D\BOOK"
ENTER
L=0
FOR X FROM 1 TO ARRAYSIZE(ITEM)
    IF SUBSTR(ITEM[X],1,1) = "*" THEN
        MOVETO FIELD "BAR_CODE"
        LEFT
        CHECK
        MOVETO FIELD "PUB_YR"
        TYPEIN SUBSTR(ITEM[X],2,LEN(ITEM[X])-1)+".
        DOWN
    ENDIF
ENDFOR
UP
IF ISBLANK( ) THEN RETURN
ENDIF
DO_IT!
QUITLOOP

OTHERWISE: ;SEARCH ON SEARCHVAL
;***** put up working message
STYLE ATTRIBUTE HIGHLIGHT
@ 24,48 ?? " SEARCHING, ONE MOMENT PLEASE"
STYLE ATTRIBUTE HIGHLIGHT+128
@ 24,77 ?? "...
CLEARALL
CANVAS ON
CANVAS OFF
CURSOR OFF
{ASK}
;***
TYPEIN "B\D\BOOK"
ENTER
CHECK
MOVETO FIELD "PUB_YR"
TYPEIN SEARCHVAL
DO_IT!
QUITLOOP

ENDSWITCH

ENDWHILE

;SHOW HOW MANY FOUND

```

```

STYLE ATTRIBUTE HIGHLIGHT
SHOWPRETTY1 = "MATCH"
SHOWPRETTY2 = "RECORD"

IF NRECORDS("ANSWER") > 1 THEN
    SHOWPRETTY1 = "MATCHES"
    SHOWPRETTY2 = "RECORDS"
ENDIF

;DRAW BOX FOR MESSAGE
PAINTCANVAS ATTRIBUTE BKCOLOR 24,0,24,79
@ 6,1 ?? " " + FILL(" ",76) + " "
@ 7,1 ?? " " + FILL(" ",76) + " "
@ 8,1 ?? " " + FILL(" ",76) + " "
@ 9,1 ?? " " + FILL(" ",76) + " "
@ 10,1 ?? " " + FILL(" ",76) + " "
@ 11,1 ?? " " + FILL(" ",76) + " "
CANVAS ON

IF IEMPTY("ANSWER") THEN
    @ 8,6 ?? FORMAT("W68,AC","NO MATCHES FOUND")
    @ 9,6 ?? FORMAT("W68,AC","    Press any key to continue...")
    GC = GETCHAR()
    RETURN
ELSE @ 8,6 ?? FORMAT("W68,AC",STRVAL(NRECORDS("ANSWER"))+" "+SHOWPRETTY1+" FOUND")
    @ 9,6 ?? FORMAT("W68,AC","Press any key to view "+SHOWPRETTY2)
    GC = GETCHAR()
ENDIF

;VIEW RECORDS FOUND
CLEARALL
RENAME "ANSWER" "TBOOK"
COPYFORM "B\D\BOOK" "2" "TBOOK" "2"
MasterView("TBOOK","2")
DELETE "TBOOK"
CLEARALL

ENDPROC
WRITELIB LIBNAME YearSearch
RELEASE PROC YearSearch
;*****
;* END MODULE: x.x YearSearch()
;*****

```

```

; /*****
; * PROCEDURE: SumAnimalSearch()
; * MODULE:
; * PURPOSE:
; * CALLED BY:
; * CALLS:
; * INPUT ASSERTION:
; * OUTPUT ASSERTION:
; * GLOBAL VARIABLES:
; * AUTHOR: JC Prather
; * DATE: August 1991
; *****/
PROC SumAnimalSearch()
    PRIVATE SEARCHTITLE, SEARCHPROMPT, SEARCHWIDTH, SEARCHVAL, X, L, J, SHOWPRETTY1,
        SHOWPRETTY2, GC

    SEARCHTITLE = "SUMMARY SEARCH BY ANIMAL NUMBER"
    SEARCHPROMPT = "ANIMAL NUMBER:"
    SEARCHWIDTH = 12

    WHILE TRUE
        SEARCHVAL = SearchBox(SEARCHTITLE, SEARCHPROMPT, SEARCHWIDTH)

        CURSOR OFF
        SWITCH
            CASE SEARCHVAL = "" OR SEARCHVAL = " ": RETURN

            CASE SEARCHVAL = "F1": ;GET PopUp OF CHOICES
                ;***** put up working message
                STYLE ATTRIBUTE HIGHLIGHT
                @ 24,59 ?? " ONE MOMENT PLEASE"
                STYLE ATTRIBUTE HIGHLIGHT+128
                @ 24,77 ?? "... "
                CANVAS ON
                CANVAS OFF
                ;****remove menu instructions and maintitle
                PAINTCANVAS ATTRIBUTE BKCOLOR 0,0,3,79
                PAINTCANVAS ATTRIBUTE BKCOLOR 24,0,24,79
                CLEARALL
                ;initialize vars
                ESCAPEFLAG = "N"
                PRMPT1 = FORMAT("W80,AC","F2-Execute search for items selected
                    Esc-Return to menu")
                PRMPT2 = FORMAT("W80,AC","ENTER-Select or un-select item          CURSOR
                    KEYS,PGUP,PGDN,HOME,END-Scroll list")
                ;GET POSSIBLE CHOICES TO ANSWER FOR PopUp
                (ASK)
                TYPEIN "S\D\ANI"
                ENTER
                MOVETO FIELD "ANIMAL_ID"
                CHECK
                DO IT!
                ;DO PopUp
                SetPopUp("ANSWER","ANIMAL_ID")
                PAINTCANVAS FILL " " ATTRIBUTE BKCOLOR 24,0,24,79
                C=MAX(INT((80-WIDTH)/2)-3,0)
                PopUp(4,C,13,1,"ANIMAL NUMBER",PRMPT1,PRMPT2)
                IF ESCAPEFLAG = "Y" THEN
                    RELEASE VARS ESCAPEFLAG, ITEM, WIDTH
                    RETURN
                ELSE RELEASE VARS ESCAPEFLAG

```

```

ENDIF
;****DO SEARCH:
;***** put up working message
STYLE ATTRIBUTE HIGHLIGHT
@ 24,48 ?? " SEARCHING, ONE MOMENT PLEASE"
STYLE ATTRIBUTE HIGHLIGHT+128
@ 24,77 ?? "... "
CLEARALL
CANVAS ON
CANVAS OFF
CURSOR OFF
;***
{ASK}
TYPEIN "S\\D\\ANI"
ENTER
L=0
FOR X FROM 1 TO ARRAYSIZE(ITEM)
    IF SUBSTR(ITEM[X],1,1) = "" THEN
        MOVETO FIELD "ANIMAL_ID"
        LEFT
        CHECK
        MOVETO FIELD "ANIMAL_ID"
        TYPEIN SUBSTR(ITEM[X],2,LEN(ITEM[X])-1)+"... "
        DOWN
    ENDIF
ENDFOR
UP
IF ISBLANK( ) THEN RETURN
ENDIF
DO_IT!
QUITLOOP

OTHERWISE: ;SEARCH ON SEARCHVAL
;***** put up working message
STYLE ATTRIBUTE HIGHLIGHT
@ 24,48 ?? " SEARCHING, ONE MOMENT PLEASE"
STYLE ATTRIBUTE HIGHLIGHT+128
@ 24,77 ?? "... "
CLEARALL
CANVAS ON
CANVAS OFF
CURSOR OFF
{ASK}
;***
TYPEIN "S\\D\\ANI"
ENTER
CHECK
MOVETO FIELD "ANIMAL_ID"
TYPEIN SEARCHVAL
DO_IT!
QUITLOOP

ENDSWITCH

ENDWHILE

;SHOW HOW MANY FOUND

```

```

STYLE ATTRIBUTE HIGHLIGHT
SHOWPRETTY1 = "MATCH"
SHOWPRETTY2 = "RECORD"

IF NRECORDS("ANSWER") > 1 THEN
    SHOWPRETTY1 = "MATCHES"
    SHOWPRETTY2 = "RECORDS"
ENDIF

;DRAW BOX FOR MESSAGE
PAINTCANVAS ATTRIBUTE BKCOLOR 24,0,24,79
@ 6,1 ?? "+FILL(" ",76)+" "
@ 7,1 ?? "+FILL(" ",76)+" "
@ 8,1 ?? "+FILL(" ",76)+" "
@ 9,1 ?? "+FILL(" ",76)+" "
@ 10,1 ?? "+FILL(" ",76)+" "
@ 11,1 ?? "+FILL(" ",76)+" "
CANVAS ON

IF IEMPTY("ANSWER") THEN
    @ 8,6 ?? FORMAT("W68,AC","NO MATCHES FOUND")
    @ 9,6 ?? FORMAT("W68,AC","    Press any key to continue...")
    GC = GETCHAR()
    RETURN
ELSE @ 8,6 ?? FORMAT("W68,AC",STRVAL(NRECORDS("ANSWER"))+" "+SHOWPRETTY1+" FOUND")
    @ 9,6 ?? FORMAT("W68,AC","Press any key to view "+SHOWPRETTY2)
    GC = GETCHAR()
ENDIF

;VIEW RECORDS FOUND
CLEARALL
RENAME "ANSWER" "TANI"
COPYFORM "S\D\ANI" "12" "TANI" "12"
MasterView("TANI","12")
DELETE "TANI"
CLEARALL

ENDPROC
WRITELIB LIBN\ME SumAnimalSearch
RELEASE PROCS SumAnimalSearch
;*****
;* END MODULE: x.x SumAnimalSearch()
;*****

```

```

; /*****
; * PROCEDURE: SumAssignedIdSearch()
; * MODULE:
; * PURPOSE:
; * CALLED BY:
; * CALLS:
; * INPUT ASSERTION:
; * OUTPUT ASSERTION:
; * GLOBAL VARIABLES:
; * AUTHOR: JC Prather
; * DATE: August 1991
; *****/
PROC SumAssignedIdSearch()
    PRIVATE SEARCHTITLE, SEARCHPROMPT, SEARCHWIDTH, SEARCHVAL, X, L, J, SHOWPRETTY1,
        SHOWPRETTY2, GC

    SEARCHTITLE = "SUMMARY SEARCH BY ASSIGNED ID"
    SEARCHPROMPT = "ASSIGNED ID:"
    SEARCHWIDTH = 20

    WHILE TRUE
        SEARCHVAL = SearchBox(SEARCHTITLE, SEARCHPROMPT, SEARCHWIDTH)

        CURSOR OFF
        SWITCH
            CASE SEARCHVAL = "" OR SEARCHVAL = " ": RETURN

            CASE SEARCHVAL = "F1": ;GET PopUp OF CHOICES
                ;***** put up working message
                STYLE ATTRIBUTE HIGHLIGHT
                @ 24,59 ?? " ONE MOMENT PLEASE"
                STYLE ATTRIBUTE HIGHLIGHT+128
                @ 24,77 ?? "..."
                CANVAS ON
                CANVAS OFF
                ;****remove menu instructions and maintitle
                PAINTCANVAS ATTRIBUTE BKCOLOR 0,0,3,79
                PAINTCANVAS ATTRIBUTE BKCOLOR 24,0,24,79
                CLEARALL
                ;initialize vars
                ESCAPEFLAG = "N"
                PRMPT1 = FORMAT("W80,AC","F2-Execute search for items selected
                    Esc-Return to menu")
                PRMPT2 = FORMAT("W80,AC","ENTER-Select or un-select item          CURSOR
                    KEYS,PGUP,PGDN,HOME,END-Scroll list")
                ;GET POSSIBLE CHOICES TO ANSWER FOR PopUp
                {ASK}
                TYPEIN "S\\D\\ANI"
                ENTER
                MOVETO FIELD "ASSIGNED_ID"
                CHECK
                DO IT!
                ;DO PopUp
                SetPopUp("ANSWER","ASSIGNED_ID")
                PAINTCANVAS FILL " " ATTRIBUTE BKCOLOR 24,0,24,79
                C=MAX(INT((80-WIDTH)/2)-3,0)
                PopUp(4,C,13,1,"ASSIGNED ID",PRMPT1,PRMPT2)
                IF ESCAPEFLAG = "Y" THEN
                    RELEASE VARS ESCAPEFLAG,ITEM,WIDTH
                RETURN
                ELSE RELEASE VARS ESCAPEFLAG

```

```

ENDIF
;****DO SEARCH:
;***** put up working message
STYLE ATTRIBUTE HIGHLIGHT
@ 24,48 ?? " SEARCHING, ONE MOMENT PLEASE"
STYLE ATTRIBUTE HIGHLIGHT+128
@ 24,77 ?? "... "
CLEARALL
CANVAS ON
CANVAS OFF
CURSOR OFF
;***
(ASK)
TYPEIN "S\D\ANI"
ENTER
L=0
FOR X FROM 1 TO ARRAYSIZE(ITEM)
    IF SUBSTR(ITEM[X],1,1) = "*" THEN
        MOVETO FIELD "ANIMAL_ID"
        LEFT
        CHECK
        MOVETO FIELD "ASSIGNED_ID"
        TYPEIN SUBSTR(ITEM[X],2,LEN(ITEM[X])-1)+". "
        DOWN
    ENDIF
ENDFOR
UP
IF ISBLANK( ) THEN RETURN
ENDIF
DO_IT!
QUITLOOP

OTHERWISE: ;SEARCH ON SEARCHVAL
;***** put up working message
STYLE ATTRIBUTE HIGHLIGHT
@ 24,48 ?? " SEARCHING, ONE MOMENT PLEASE"
STYLE ATTRIBUTE HIGHLIGHT+128
@ 24,77 ?? "... "
CLEARALL
CANVAS ON
CANVAS OFF
CURSOR OFF
(ASK)
;***
TYPEIN "S\D\ANI"
ENTER
CHECK
MOVETO FIELD "ASSIGNED_ID"
TYPEIN SEARCHVAL
DO_IT!
QUITLOOP

ENDSWITCH

ENDWHILE

;SHOW HOW MANY FOUND

```

```

STYLE ATTRIBUTE HIGHLIGHT
SHOWPRETTY1 = "MATCH"
SHOWPRETTY2 = "RECORD"

IF NRECORDS("ANSWER") > 1 THEN
    SHOWPRETTY1 = "MATCHES"
    SHOWPRETTY2 = "RECORDS"
ENDIF

;DRAW BOX FOR MESSAGE
PAINTCANVAS ATTRIBUTE BKCOLOR 24,0,24,79
@ 6,1 ?? " "+FILL("-",76)+" "
@ 7,1 ?? " "+FILL(" ",76)+" "
@ 8,1 ?? " "+FILL(" ",76)+" "
@ 9,1 ?? " "+FILL(" ",76)+" "
@ 10,1 ?? " "+FILL(" ",76)+" "
@ 11,1 ?? " "+FILL("-",76)+" "
CANVAS ON

IF IEMPTY("ANSWER") THEN
    @ 8,6 ?? FORMAT("W68,AC","NO MATCHES FOUND")
    @ 9,6 ?? FORMAT("W68,AC"," Press any key to continue...")
    GC = GETCHAR()
    RETURN
ELSE @ 8,6 ?? FORMAT("W68,AC",STRVAL(NRECORDS("ANSWER"))+" "+SHOWPRETTY1+" FOUND")
    @ 9,6 ?? FORMAT("W68,AC","Press any key to view "+SHOWPRETTY2)
    GC = GETCHAR()
ENDIF

;VIEW RECORDS FOUND
CLEARALL
RENAME "ANSWER" "TANI"
COPYFORM "S\D\\ANI" "12" "TANI" "12"
MasterView("TANI","12")
DELETE "TANI"
CLEARALL

ENDPROC
WRITELIB LIBNAME SumAssignedIdSearch
RELEASE PROCS SumAssignedIdSearch
;*****
;* END MODULE: x.x SumAssignedIdSearch()
;*****

```

```

; /*****
; * PROCEDURE: InvAnimalSearch()
; * MODULE:
; * PURPOSE:
; * CALLED BY:
; * CALLS:
; * INPUT ASSERTION:
; * OUTPUT ASSERTION:
; * GLOBAL VARIABLES:
; * AUTHOR: JC Prather
; * DATE: August 1991
; *****/
PROC InvAnimalSearch()
    PRIVATE SEARCHTITLE,SEARCHPROMPT,SEARCHWIDTH,SEARCHVAL,X,L,J,SHOWPRETTY1,
        SHOWPRETTY2,GC,C

    SEARCHTITLE = "INVENTORY SEARCH BY ANIMAL NUMBER"
    SEARCHPROMPT = "ANIMAL NUMBER:"
    SEARCHWIDTH = 12

    WHILE TRUE
        SEARCHVAL = SearchBox(SEARCHTITLE,SEARCHPROMPT,SEARCHWIDTH)

        CURSOR OFF
        SWITCH
            CASE SEARCHVAL = "" OR SEARCHVAL = " ": RETURN

            CASE SEARCHVAL = "F1": ;GET PopUp OF CHOICES
                ;***** put up working message
                STYLE ATTRIBUTE HIGHLIGHT
                @ 24,59 ?? " ONE MOMENT PLEASE"
                STYLE ATTRIBUTE HIGHLIGHT+128
                @ 24,77 ?? " .."
                CANVAS ON
                CANVAS OFF
                ;****remove menu instructions and maintitle
                PAINTCANVAS ATTRIBUTE BKCOLOR 0,0,3,79
                PAINTCANVAS ATTRIBUTE BKCOLOR 24,0,24,79
                CLEARALL
                ;initialize vars
                ESCAPEFLAG = "N"
                PRMPT1 = FORMAT("W80,AC","F2-Execute search for items selected
                    Esc-Return to menu")
                PRMPT2 = FORMAT("W80,AC","ENTER-Select or un-select item          CURSOR
                    KEYS,PGUP,PGDN,HOME,END-Scroll list")
                ;GET POSSIBLE CHOICES TO ANSWER FOR PopUp
                {ASK}
                TYPEIN "S\D\ANI"
                ENTER
                MOVETO FIELD "ANIMAL_ID"
                CHECK
                DO_IT!
                ;DO PopUp
                SetPopUp("ANSWER","ANIMAL_ID")
                PAINTCANVAS FILL " " ATTRIBUTE BKCOLOR 24,0,24,79
                C=MAX(INT((80-WIDTH)/2)-3,0)
                PopUp(4,C,14,1,"ANIMAL NUMBER",PRMPT1,PRMPT2)
                IF ESCAPEFLAG = "Y" THEN
                    RELEASE VARS ESCAPEFLAG,ITEM,WIDTH
                RETURN
                ELSE RELEASE VARS ESCAPEFLAG

```

```

ENDIF
;****DO SEARCH:
;***** put up working message
STYLE ATTRIBUTE HIGHLIGHT
@ 24,48 ?? " SEARCHING, ONE MOMENT PLEASE"
STYLE ATTRIBUTE HIGHLIGHT+128
@ 24,77 ?? "...
CLEARALL
CANVAS ON
CANVAS OFF
CURSOR OFF
;***
(ASK)
TYPEIN "S\D\ANI"
ENTER
L=0
FOR X FROM 1 TO ARRAYSIZE(ITEM)
  IF SUBSTR(ITEM[X],1,1) = "*" THEN
    MOVETO FIELD "ANIMAL_ID"
    EXAMPLE
    L = L + 1
    TYPEIN "A"+STRVAL(L)
    MOVETO FIELD "ANIMAL_ID"
    TYPEIN " "+SUBSTR(ITEM[X],2,LEN(ITEM[X])-1)
    DOWN
  ENDIF
ENDFOR
UP
IF ISBLANK( ) THEN RETURN
ENDIF
(ASK)
TYPEIN "I\D\DETAIL"
ENTER
FOR J FROM 1 TO L
  CHECK
  MOVETO FIELD "ANIMAL_ID"
  EXAMPLE
  TYPEIN "A"+STRVAL(J)
  MOVETO FIELD "#"
  DOWN
ENDFOR
DO_IT!
QUITLOOP

OTHERWISE: ;SEARCH ON SEARCHVAL
;***** put up working message
STYLE ATTRIBUTE HIGHLIGHT
@ 24,48 ?? " SEARCHING, ONE MOMENT PLEASE"
STYLE ATTRIBUTE HIGHLIGHT+128
@ 24,77 ?? "...
CLEARALL
CANVAS ON
CANVAS OFF
CURSOR OFF
(ASK)
;***
TYPEIN "S\D\ANI"
ENTER
MOVETO FIELD "ANIMAL_ID"
EXAMPLE
TYPEIN "A"
TYPEIN " "+SEARCHVAL

```

```

        (ASK)
        TYPEIN "I\\D\\DETAIL"
        ENTER
        CHECK
        MOVETO FIELD "ANIMAL_ID"
        EXAMPLE
        TYPEIN "A"
        DO_IT!
        QUITLOOP

ENDSWITCH

ENDWHILE

;SHOW HOW MANY FOUND

STYLE ATTRIBUTE HIGHLIGHT
SHOWPRETTY1 = "MATCH"
SHOWPRETTY2 = "RECORD"

IF NRECORDS("ANSWER") > 1 THEN
    SHOWPRETTY1 = "MATCHES"
    SHOWPRETTY2 = "RECORDS"
ENDIF

;DRAW BOX FOR MESSAGE
PAINTCANVAS ATTRIBUTE BKCOLOR 24,0,24,79
@ 6,5 ?? "+"FILL("-",68)+"|"
@ 7,5 ?? "+"FILL(" ",68)+"|"
@ 8,5 ?? "+"FILL(" ",68)+"|"
@ 9,5 ?? "+"FILL(" ",68)+"|"
@ 10,5 ?? "+"FILL(" ",68)+"|"
@ 11,5 ?? "+"FILL("-",68)+"|"
CANVAS ON

IF ISEMPY("ANSWER") THEN
    @ 8,6 ?? FORMAT("W68,AC","NO MATCHES FOUND")
    @ 9,6 ?? FORMAT("W68,AC"," Press any key to continue...")
    GC = GETCHAR()
    RETURN
ELSE @ 8,6 ?? FORMAT("W68,AC",STRVAL(NRECORDS("ANSWER"))+" "+SHOWPRETTY1+" FOUND")
    @ 9,6 ?? FORMAT("W68,AC","Press any key to view "+SHOWPRETTY2)
    GC = GETCHAR()
ENDIF

;VIEW RECORDS FOUND
CLEARALL
RENAME "ANSWER" "TDETAIL"
COPYFORM "I\\D\\DETAIL" "2" "TDETAIL" "2"
MasterView("TDETAIL","2")
DELETE "TDETAIL"
CLEARALL

ENDPROC
WRITELIB LIBNAME InvAnimalSearch
RELEASE PROCS InvAnimalSearch
;*****
;* END MODULE: x.x InvAnimalSearch()
;*****

```

```

; /*****
; * PROCEDURE: InvAssignedIdSearch()
; * MODULE:
; * PURPOSE:
; * CALLED BY:
; * CALLS:
; * INPUT ASSERTION:
; * OUTPUT ASSERTION:
; * GLOBAL VARIABLES:
; * AUTHOR: JC Prather
; * DATE: August 1991
; *****/
PROC InvAssignedIdSearch()
    PRIVATE SEARCHTITLE,SEARCHPROMPT,SEARCHWIDTH,SEARCHVAL,X,L,J,SHOWPRETTY1,
        SHOWPRETTY2,GC,C

    SEARCHTITLE = "INVENTORY SEARCH BY ASSIGNED ID"
    SEARCHPROMPT = "ASSIGNED ID:"
    SEARCHWIDTH = 20

    WHILE TRUE
        SEARCHVAL = SearchBox(SEARCHTITLE,SEARCHPROMPT,SEARCHWIDTH)

        CURSOR OFF
        SWITCH
            CASE SEARCHVAL = "" OR SEARCHVAL = " ": RETURN

            CASE SEARCHVAL = "F1": ;GET PopUp OF CHOICES
                ;***** put up working message
                STYLE ATTRIBUTE HIGHLIGHT
                @ 24,59 ?? " ONE MOMENT PLEASE"
                STYLE ATTRIBUTE HIGHLIGHT+128
                @ 24,77 ?? "..."
                CANVAS ON
                CANVAS OFF
                ;****remove menu instructions and maintitle
                PAINTCANVAS ATTRIBUTE BKCOLOR 0,0,3,79
                PAINTCANVAS ATTRIBUTE BKCOLOR 24,0,24,79
                CLEARALL
                ;initialize vars
                ESCAPEFLAG = "N"
                PRMPT1 = FORMAT("W80,AC","F~-Execute search for items selected
E~-Return to menu")
                PRMPT2 = FORMAT("W80,AC","ENTER-Select or un-select item          CURSOR
KEY~,PGUP,PGDN,HOME,END-Scroll list")
                ;GET POSSIBLE CHOICES TO ANSWER FOR PopUp
                (ASK)
                TYPEIN "S\\D\\ANI"
                ENTER
                MOVETO FIELD "ASSIGNED_ID"
                CHECK
                DO IT!
                ;DO PopUp
                SetPopUp("ANSWER","ASSIGNED_ID")
                PAINTCANVAS FILL " " ATTRIBUTE BKCOLOR 24,0,24,79
                C=MAX(INT((80-WIDTH)/2)-3,0)
                PopUp(4,C,14,1,"ASSIGNED ID",PRMPT1,PRMPT2)
                IF ESCAPEFLAG = "Y" THEN
                    RELEASE VARS ESCAPEFLAG,ITEM,WIDTH

```

```

        RETURN
    ELSE RELEASE VARS ESCAPEFLAG
ENDIF
;****DO SEARCH:
;***** put up working message
STYLE ATTRIBUTE HIGHLIGHT
@ 24,48 ?? " SEARCHING, ONE MOMENT PLEASE"
STYLE ATTRIBUTE HIGHLIGHT+128
@ 24,77 ?? "...
CLEARALL
CANVAS ON
CANVAS OFF
CURSOR OFF
;***
(ASK)
TYPEIN "S\D\ANI"
ENTER
L=0
FOR X FROM 1 TO ARRAYSIZE(ITEM)
    IF SUBSTR(ITEM[X],1,1) = "*" THEN
        MOVETO FIELD "ANIMAL_ID"
        EXAMPLE
        L = L + 1
        TYPEIN "A"+STRVAL(L)
        MOVETO FIELD "ASSIGNED_ID"
        TYPEIN SUBSTR(ITEM[X],2,LEN(ITEM[X])-1)
        DOWN
    ENDIF
ENDFOR
UP
IF ISBLANK({}) THEN RETURN
ENDIF
(ASK)
TYPEIN "I\D\DETAIL"
ENTER
FOR J FROM 1 TO L
    CHECK
    MOVETO FIELD "ANIMAL_ID"
    EXAMPLE
    TYPEIN "A"+STRVAL(J)
    MOVETO FIELD "#"
    DOWN
ENDFOR
DO_IT!
QUITLOOP

OTHERWISE: ;SEARCH ON SEARCHVAL
;***** put up working message
STYLE ATTRIBUTE HIGHLIGHT
@ 24,48 ?? " SEARCHING, ONE MOMENT PLEASE"
STYLE ATTRIBUTE HIGHLIGHT+128
@ 24,77 ?? "...
CLEARALL
CANVAS ON
CANVAS OFF
CURSOR OFF
(ASK)
;***
TYPEIN "S\D\ANI"
ENTER
MOVETO FIELD "ANIMAL_ID"
EXAMPLE

```

```

        TYPEIN "A"
        MOVETO FIELD "ASSIGNED_ID"
        TYPEIN SEARCHVAL
        {ASK}
        TYPEIN "I\\D\\DETAIL"
        ENTER
        CHECK
        MOVETO FIELD "ANIMAL_ID"
        EXAMPLE
        TYPEIN "A"
        DO_IT!
        QUITLOOP

ENDSWITCH

ENDWHILE

;SHOW HOW MANY FOUND

STYLE ATTRIBUTE HIGHLIGHT
SHOWPRETTY1 = "MATCH"
SHOWPRETTY2 = "RECORD"

IF NRECORDS("ANSWER") > 1 THEN
    SHOWPRETTY1 = "MATCHES"
    SHOWPRETTY2 = "RECORDS"
ENDIF

;DRAW BOX FOR MESSAGE
PAINTCANVAS ATTRIBUTE BKCOLOR 24,0,24,79
@ 6,5 ?? " "+FILL(" ",68)+" "
@ 7,5 ?? " "+FILL(" ",68)+" "
@ 8,5 ?? " "+FILL(" ",68)+" "
@ 9,5 ?? " "+FILL(" ",68)+" "
@ 10,5 ?? " "+FILL(" ",68)+" "
@ 11,5 ?? " "+FILL(" ",68)+" "
CANVAS ON

IF IEMPTY("ANSWER") THEN
    @ 8,6 ?? FORMAT("W68,AC","NO MATCHES FOUND")
    @ 9,6 ?? FORMAT("W68,AC"," Press any key to continue...")
    GC = GETCHAR()
    RETURN
ELSE @ 8,6 ?? FORMAT("W68,AC",STRVAL(NRECORDS("ANSWER"))+" "+SHOWPRETTY1+" FOUND")
    @ 9,6 ?? FORMAT("W68,AC","Press any key to view "+SHOWPRETTY2)
    GC = GETCHAR()
ENDIF

;VIEW RECORDS FOUND

CLEARALL
CLEARALL
RENAME "ANSWER" "TDETAIL"
COPYFORM "I\\D\\DETAIL" "2" "TDETAIL" "2"
MasterView("TDETAIL","2")
DELETE "TDETAIL"

```

CLEARALL

ENDPROC

WRITELIB LIBNAME InvAssignedIdSearch

RELEASE PROCS InvAssignedIdSearch

;*****

;* END MODULE: x.x InvAssignedIdSearch()

;*****/

```

;*****
;* PROCEDURE: TEffectSearch()
;* MODULE:
;* PURPOSE:
;* CALLED BY:
;* CALLS:
;* INPUT ASSERTION:
;* OUTPUT ASSERTION:
;* GLOBAL VARIABLES:
;* AUTHOR: JC Prather
;* DATE: August 1991
;*****/
PROC TEffectSearch()
    PRIVATE SEARCHTITLE,SEARCHPROMPT,SEARCHWIDTH,SEARCHVAL,X,L,J,SHOWPRETTY1,
        SHOWPRETTY2,GC,C

CURSOR OFF

        ;***** put up working message
        STYLE ATTRIBUTE HIGHLIGHT
        @ 24,59 ?? " ONE MOMENT PLEASE"
        STYLE ATTRIBUTE HIGHLIGHT+128
        @ 24,77 ?? "... "
        CANVAS ON
        CANVAS OFF
        ;****remove menu instructions and maintitle
        PAINTCANVAS ATTRIBUTE BKCOLOR 0,0,3,79
        PAINTCANVAS ATTRIBUTE BKCOLOR 24,0,24,79
        CLEARALL
        ;initialize vars
        ESCAPEFLAG = "N"
        PRMPT1 = FORMAT("W80,AC","F2-Execute search for items selected
Esc-Return to menu")
        PRMPT2 = FORMAT("W80,AC","ENTER-Select or un-select item
PGDN,HOME,END-Scroll list")
        ;GET POSSIBLE CHOICES TO ANSWER FOR PopUp
        (ASK)
        TYPEIN "W\\TEFFECT1"
        ENTER
        MOVETO FIELD "TISSUE_TRANS"
        CHECK
        MOVETO FIELD "MORP_TRANS"
        CHECK
        DO IT!
        CLEARALL
        EMPTY "W\\TEXT"
        (ASK)
        TYPEIN "ANSWER"
        ENTER
        MOVETO FIELD "TISSUE_TRANS"
        EXAMPLE "A"
        MOVETO FIELD "MORP_TRANS"
        EXAMPLE "B"
        (ASK)
        TYPEIN "W\\TEXT"
        ENTER
        TYPEIN "INSERT"
        MOVETO FIELD "TEXT"
        EXAMPLE
        TYPEIN "A"+"+ \": \"+ "
        EXAMPLE

```

```

TYPEIN "B"
DO_IT!

;DO PopUp
SetPopUp("W\\TEXT","TEXT")
PAINTCANVAS FILL " " ATTRIBUTE BKCOLOR 24,0,24,79
C=MAX(INT((80-WIDTH)/2)-3,0)
PopUp(4,C,14,1,"T_EFFECT",PRMPT1,PRMPT2)
IF ESCAPEFLAG = "Y" THEN
    RELEASE VARS ESCAPEFLAG,ITEM,WIDTH
    RETURN
ELSE RELEASE VARS ESCAPEFLAG
ENDIF
;****DO SEARCH:
;***** put up working message
STYLE ATTRIBUTE HIGHLIGHT
@ 24,48 ?? " SEARCHING, ONE MOMENT PLEASE"
STYLE ATTRIBUTE HIGHLIGHT+128
@ 24,77 ?? "...
CLEARALL
CANVAS ON
CANVAS OFF
CURSOR OFF
;***
{ASK}
TYPEIN "W\\TEFFECT1"
ENTER
L=0
FOR X FROM 1 TO ARRAYSIZE(ITEM)
    IF SUBSTR(ITEM[X],1,1) = "*" THEN
        MOVETO FIELD "ANIMAL_ID"
        EXAMPLE
        L = L + 1
        TYPEIN "A"+STRVAL(L)
        MOVETO FIELD "TISSUE_TRANS"
        TYPEIN "\""+SUBSTR(ITEM[X],2,SEARCH(":",ITEM[X])-2)+"\"
        MOVETO FIELD "MORP_TRANS"
        TYPEIN "\""+SUBSTR(ITEM[X],SEARCH(":",ITEM[X])+4,LEN(ITEM[X]))+"\"
        DOWN
    ENDIF
ENDFOR
UP
IF ISBLANK([]) THEN RETURN
ENDIF

{ASK}
TYPEIN "S\\D\\ANI"
ENTER
FOR J FROM 1 TO L
    CHECK
    MOVETO FIELD "ANIMAL_ID"
    EXAMPLE
    TYPEIN "A"+STRVAL(J)
    MOVETO FIELD "#"
    DOWN
ENDFOR
DO_IT!

```

```

;SHOW HOW MANY FOUND

STYLE ATTRIBUTE HIGHLIGHT
SHOWPRETTY1 = "MATCH"
SHOWPRETTY2 = "RECORD"

IF NRECORDS("ANSWER") > 1 THEN
    SHOWPRETTY1 = "MATCHES"
    SHOWPRETTY2 = "RECORDS"
ENDIF

;DRAW BOX FOR MESSAGE
PAINTCANVAS ATTRIBUTE BKCOLOR 24,0,24,79
@ 6,5 ?? " "+FILL("-",68)+" "
@ 7,5 ?? " "+FILL(" ",68)+" "
@ 8,5 ?? " "+FILL(" ",68)+" "
@ 9,5 ?? " "+FILL(" ",68)+" "
@ 10,5 ?? " "+FILL(" ",68)+" "
@ 11,5 ?? " "+FILL("-",68)+" "
CANVAS ON

IF IEMPTY("ANSWER") THEN
    @ 8,6 ?? FORMAT("W68,AC","NO MATCHES FOUND")
    @ 9,6 ?? FORMAT("W68,AC","    Press any key to continue...")
    GC = GETCHAR()
    RETURN
ELSE @ 8,6 ?? FORMAT("W68,AC",STRVAL(NRECORDS("ANSWER"))+" "+SHOWPRETTY1+" FOUND")
    @ 9,6 ?? FORMAT("W68,AC","Press any key to view "+SHOWPRETTY2)
    GC = GETCHAR()
ENDIF

;VIEW RECORDS FOUND
CLEARALL
RENAME "ANSWER" "TANI"
COPYFORM "S\D\ANI" "12" "TANI" "12"
MasterView("TANI","12")
DELETE "TANI"
CLEARALL

ENDPROC
WRITELIB LIBNAME TEffectSearch
RELEASE PROCS TEffectSearch
;*****
;* END MODULE: x.x TEffectSearch()
;*****

```

```

;*****
;* PROCEDURE: TDoseSearch()
;* MODULE:
;* PURPOSE:
;* CALLED BY:
;* CALLS:
;* INPUT ASSERTION:
;* OUTPUT ASSERTION:
;* GLOBAL VARIABLES:
;* AUTHOR: JC Prather
;* DATE: August 1991
;*****/
PROC TDoseSearch()
  PRIVATE SEARCHTITLE,SEARCHPROMPT,SEARCHWIDTH,SEARCHVAL,X,L,J,SHOWPRETTY1,
    SHOWPRETTY2,GC,C

  CURSOR OFF

  ;***** put up working message
  STYLE ATTRIBUTE HIGHLIGHT
  @ 24,59 ?? " ONE MOMENT PLEASE"
  STYLE ATTRIBUTE HIGHLIGHT+128
  @ 24,77 ?? "... "
  CANVAS ON
  CANVAS OFF
  ;****remove menu instructions and maintitle
  PAINTCANVAS ATTRIBUTE BKCOLOR 0,0,3,79
  PAINTCANVAS ATTRIBUTE BKCOLOR 24,0,24,79
  CLEARALL
  ;initialize vars
  ESCAPEFLAG = "N"
  PRMPT1 = FORMAT("W80,AC","F2-Execute search for items selected
Esc-Return to menu")
  PRMPT2 = FORMAT("W80,AC","ENTER-Select or un-select item          CURSOR
KEYS,PGUP,PGDN,HOME,END-Scroll list")
  ;GET POSSIBLE CHOICES TO ANSWER FOR PopUp
  {ASK}
  TYPEIN "W\\TDOSE1"
  ENTER
  MOVETO FIELD "TISSUE_TRANS"
  CHECK
  MOVETO FIELD "DOSE_TISS_DX"
  CHECK
  MOVETO FIELD "DOSE_UNITS"
  CHECK
  DO_IT!

;MAKE POSSIBLE STRING MANIPULATION:
(Modify) (Restructure) (Answer) Right Right Down Backspace "A"
"30" Do_it!

  CLEARALL
  EMPTY "W\\TEXT"
  {ASK}
  TYPEIN "ANSWER"
  ENTER
  MOVETO FIELD "TISSUE_TRANS"
  EXAMPLE "A"
  MOVETO FIELD "DOSE_TISS_DX"
  EXAMPLE "B"
  MOVETO FIELD "DOSE_UNITS"

```

```

EXAMPLE "C"

{ASK}
TYPEIN "W\\TEXT"
ENTER
TYPEIN "INSERT"
MOVETO FIELD "TEXT"
EXAMPLE
TYPEIN "A"+ "+" \": \"+ "
EXAMPLE
TYPEIN "B"+ "+" \": \"+ "
EXAMPLE
TYPEIN "C"
DO_IT!

;DO PopUp
SetPopUp("W\\TEXT","TEXT")
PAINTCANVAS FILL " " ATTRIBUTE BKCOLOR 24,0,24,79
C=MAX(INT((80-WIDTH)/2)-3,0)
PopUp(4,C,14,1,"T_DOSE",PRMPT1,PRMPT2)
IF ESCAPEFLAG = "Y" THEN
    RELEASE VARS ESCAPEFLAG,ITEM,WIDTH
    RETURN
ELSE RELEASE VARS ESCAPEFLAG
ENDIF

;****DO SEARCH:
;***** put up working message
STYLE ATTRIBUTE HIGHLIGHT
@ 24,48 ?? " SEARCHING, ONE MOMENT PLEASE"
STYLE ATTRIBUTE HIGHLIGHT+128
@ 24,77 ?? "... "
CLEARALL
CANVAS ON
CANVAS OFF
CURSOR OFF
;***
{ASK}
TYPEIN "W\\TDOSE1"
ENTER
L=0
FOR X FROM 1 TO ARRAYSIZE(ITEM)
    IF SUBSTR(ITEM[X],1,1) = "*" THEN
        MOVETO FIELD "ANIMAL_ID"
        EXAMPLE
        L = L + 1
        TYPEIN "A"+STRVAL(L)
        MOVETO FIELD "TISSUE_TRANS"
        TYPEIN "\""+SUBSTR(ITEM[X],2,SEARCH(":",ITEM[X])-2)+"\"""
        MOVETO FIELD "DOSE_TISS_DX"
        TYPEIN
SUBSTR(ITEM[X],SEARCH(":",ITEM[X])+4,LEN(ITEM[X])-(SEARCH(":",ITEM[X])+4)-
(LEN(ITEM[X]) - SEARCH("!",ITEM[X])))
        MOVETO FIELD "DOSE_UNITS"
        TYPEIN "\""+SUBSTR(ITEM[X],SEARCH("!",ITEM[X])+2,LEN(ITEM[X]))+"\"""
        DOWN
    ENDIF
ENDFOR
UP
IF ISBLANK( ) THEN RETURN
ENDIF

```

```

        {ASK}
        TYPEIN "S\D\ANI"
        ENTER
        FOR J FROM 1 TO L
            CHECK
            MOVETO FIELD "ANIMAL_ID"
            EXAMPLE
            TYPEIN "A"+STRVAL(J)
            MOVETO FIELD "#"
            DOWN
        ENDFOR
        DO_IT!

;SHOW HOW MANY FOUND

STYLE ATTRIBUTE HIGHLIGHT
SHOWPRETTY1 = "MATCH"
SHOWPRETTY2 = "RECORD"

IF NRECORDS("ANSWER") > 1 THEN
    SHOWPRETTY1 = "MATCHES"
    SHOWPRETTY2 = "RECORDS"
ENDIF

;DRAW BOX FOR MESSAGE
PAINTCANVAS ATTRIBUTE BKCOLOR 24,0,24,79
@ 6,5 ?? " "+FILL("-",68)+" "
@ 7,5 ?? " "+FILL(" ",68)+" "
@ 8,5 ?? " "+FILL(" ",68)+" "
@ 9,5 ?? " "+FILL(" ",68)+" "
@ 10,5 ?? " "+FILL(" ",68)+" "
@ 11,5 ?? " "+FILL("-",68)+" "
CANVAS ON

IF ISEMPY("ANSWER") THEN
    @ 8,6 ?? FORMAT("W68,AC","NO MATCHES FOUND")
    @ 9,6 ?? FORMAT("W68,AC"," Press any key to continue...")
    GC = GETCHAR()
    RETURN
ELSE @ 8,6 ?? FORMAT("W68,AC",STRVAL(NRECORDS("ANSWER"))+" "+SHOWPRETTY1+" FOUND")
    @ 9,6 ?? FORMAT("W68,AC","Press any key to view "+SHOWPRETTY2)
    GC = GETCHAR()
ENDIF

;VIEW RECORDS FOUND
;VIEW RECORDS FOUND
CLEARALL
RENAME "ANSWER" "TANI"
COPYFORM "S\D\ANI" "12" "TANI" "12"
MasterView("TANI","12")
DELETE "TANI"
CLEARALL

ENDPROC
WRITELIB LIBNAME TDoseSearch
RELEASE PROCS TDoseSearch
;*****
;* END MODULE: x.x TDoseSearch()
;*****

```

```

; /*****
; * PROCEDURE: SumStudySearch()
; * MODULE:
; * PURPOSE:
; * CALLED BY:
; * CALLS:
; * INPUT ASSERTION:
; * OUTPUT ASSERTION:
; * GLOBAL VARIABLES:
; * AUTHOR: JC Prather
; * DATE: August 1991
; *****/
PROC SumStudySearch()
    PRIVATE SEARCHTITLE, SEARCHPROMPT, SEARCHWIDTH, SEARCHVAL, X, L, J, SHOWPRETTY1,
        SHOWPRETTY2, GC, C, TITLELENGTH

    SEARCHTITLE = "SUMMARY SEARCH BY STUDY NAME"
    SEARCHPROMPT = "STUDY NAME:"
    SEARCHWIDTH = 70

    WHILE TRUE
        SEARCHVAL = SearchBox(SEARCHTITLE, SEARCHPROMPT, SEARCHWIDTH)

        CURSOR OFF
        SWITCH
            CASE SEARCHVAL = "" OR SEARCHVAL = " ": RETURN

            CASE SEARCHVAL = "F1": ;GET PopUp OF CHOICES
                ;***** put up working message
                STYLE ATTRIBUTE HIGHLIGHT
                @ 24,59 ?? " ONE MOMENT PLEASE"
                STYLE ATTRIBUTE HIGHLIGHT+128
                @ 24,77 ?? "..."
                CANVAS ON
                CANVAS OFF
                ;****remove menu instructions and maintitle
                PAINTCANVAS ATTRIBUTE BKCOLOR 0,0,3,79
                PAINTCANVAS ATTRIBUTE BKCOLOR 24,0,24,79
                CLEARALL
                ;initialize vars
                ESCAPEFLAG = "N"
                PRMPT1 = FORMAT("W80,AC","F2-Execute search for items selected
                    Esc-Return to menu")
                PRMPT2 = FORMAT("W80,AC","ENTER-Select or un-select item
                    PGDN,HOME,END-Scroll list")
                ;GET POSSIBLE CHOICES TO ANSWER FOR PopUp

                (ASK)
                TYPEIN "S\\D\\STU"
                ENTER
                MOVETO FIELD "NRA_STUDY_NAME"
                CHECK
                DO_IT!
;COPY TO TEMP DB
                COPY "ANSWER" "W\\TEMP"

;ADD NO. OF OCCURANCES:
                EMPTY "W\\TEXT"
                CLEARALL
                (ASK)

```

```

TYPEIN "S\\D\\STU"
ENTER
MOVETO FIELD "LAB_ID"
EXAMPLE
TYPEIN "A"
MOVETO FIELD "STUDY_ID"
EXAMPLE
TYPEIN "B"
MOVETO FIELD "NRA_STUDY_NAME"
CHECK
(ASK)
TYPEIN "S\\D\\ANI"
ENTER
MOVETO FIELD "LAB_ID"
EXAMPLE
TYPEIN "A"
MOVETO FIELD "STUDY_ID"
EXAMPLE
TYPEIN "B"
MOVETO FIELD "ANIMAL_ID"
TYPEIN "CALC COUNT"
DO_IT!
(Modify) (Restructure) {W\\Temp} Down "Count of ANIMAL_ID" Right "A30"
Do_It!
(Modify) (Restructure) {answer} Down Right Right Backspace "A30" Do_It!
CLEARALL
(ASK)
TYPEIN "W\\TEMP"
ENTER
MOVETO FIELD "NRA_STUDY_NAME"
EXAMPLE
TYPEIN "A"
MOVETO FIELD "Count of ANIMAL_ID"
TYPEIN "CHANGETO "
EXAMPLE
TYPEIN "B"
(ASK)
TYPEIN "ANSWER"
ENTER
MOVETO FIELD "NRA_STUDY_NAME"
EXAMPLE
TYPEIN "A"
MOVETO FIELD "Count of ANIMAL_ID"
EXAMPLE
TYPEIN "B"
DO_IT!

CLEARALL
(ASK)
TYPEIN "W\\TEMP"
ENTER
MOVETO FIELD "Count of ANIMAL_ID"
TYPEIN "BLANK, CHANGETO 0"
DO_IT!

EDIT "W\\TEMP"
SCAN
    MOVETO FIELD "NRA_STUDY_NAME"
    TITLELENGTH = LEN( )
    IF TITLELENGTH > 65 THEN
        [ ] = SUBSTR( [ ],1,50)+
            "..."

```

```

        SUBSTR(,TITLELENGTH-10,TITLELENGTH)+
        " | "+"
        [Count of ANIMAL_ID]
ELSE
    [] = []+
    "..."
    SPACES(65 - TITLELENGTH) +
    " | "+"
    [Count of ANIMAL_ID]
ENDIF
ENDSCAN
DO_IT!

;DO PopUp
SetPopUp("W\\TEMP","NRA_STUDY_NAME")
DELETE "W\\TEMP"
PAINTCANVAS FILL " " ATTRIBUTE BKCOLOR 24,0,24,79
C=MAX(INT((80-WIDTH)/2)-3,0)
PopUp(4,C,14,1,"                NRA STUDY NAME                | # animals
available",PRMPT1,PRMPT2);
IF ESCAPEFLAG = "Y" THEN
    RELEASE VARS ESCAPEFLAG,ITEM,WIDTH
    RETURN
ELSE RELEASE VARS ESCAPEFLAG
ENDIF

;****DO SEARCH:
;***** put up working message
STYLE ATTRIBUTE HIGHLIGHT
@ 24,48 ?? " SEARCHING, ONE MOMENT PLEASE"
STYLE ATTRIBUTE HIGHLIGHT+128
@ 24,77 ?? "..."
CLEARALL
CANVAS ON
CANVAS OFF
CURSOR OFF
;***
(ASK)
TYPEIN "S\\D\\STU"
ENTER
L=0
FOR X FROM 1 TO ARRAYSIZE(ITEM)
    IF SUBSTR(ITEM[X],1,1) = "*" THEN
        MOVETO FIELD "LAB_ID"
        EXAMPLE
        L = L + 1
        TYPEIN "A"+STRVAL(L)
        MOVETO FIELD "STUDY_ID"
        EXAMPLE
        TYPEIN "B"+STRVAL(L)
        MOVETO FIELD "NRA_STUDY_NAME"
        TYPEIN "\""+SUBSTR(ITEM[X],2,SEARCH("...",ITEM[X])-2)+"\"..."
        DOWN
    ENDIF
ENDFOR
UP
IF ISBLANK([]) THEN RETURN
ENDIF
(ASK)
TYPEIN "S\\D\\ANI"
ENTER

```

```

FOR J FROM 1 TO L
  CHECK
  MOVETO FIELD "LAB_ID"
  EXAMPLE
  TYPEIN "A"+STRVAL(J)
  MOVETO FIELD "STUDY_ID"
  EXAMPLE
  TYPEIN "B"+STRVAL(J)
  MOVETO FIELD "#"
  DOWN
ENDFOR
DO_IT!
QUITLOOP

OTHERWISE: ;SEARCH ON SEARCHVAL
;***** put up working message
STYLE ATTRIBUTE HIGHLIGHT
@ 24,48 ?? " SEARCHING, ONE MOMENT PLEASE"
STYLE ATTRIBUTE HIGHLIGHT+128
@ 24,77 ?? "... "
CLEARALL
CANVAS ON
CANVAS OFF
CURSOR OFF
(ASK)
;***
TYPEIN "S\\D\\STU"
ENTER
MOVETO FIELD "LAB_ID"
EXAMPLE
TYPEIN "A"
MOVETO FIELD "STUDY_ID"
EXAMPLE
TYPEIN "B"
MOVETO FIELD "NRA_STUDY_NAME"
TYPEIN SEARCHVAL
(ASK)
TYPEIN "S\\D\\ANI"
ENTER
CHECK
MOVETO FIELD "LAB_ID"
EXAMPLE
TYPEIN "A"
MOVETO FIELD "STUDY_ID"
EXAMPLE
TYPEIN "B"
DO_IT!
QUITLOOP

ENDSWITCH

ENDWHILE

;SHOW HOW MANY FOUND

STYLE ATTRIBUTE HIGHLIGHT
SHOWPRETTY1 = "MATCH"
SHOWPRETTY2 = "RECORD"

IF NRECORDS("ANSWER") > 1 THEN
  SHOWPRETTY1 = "MATCHES"
  SHOWPRETTY2 = "RECORDS"

```

```

ENDIF

;DRAW BOX FOR MESSAGE
PAINTCANVAS ATTRIBUTE BKCOLOR 24,0,24,79
@ 6,5 ?? " "+FILL(" ",68)+" "
@ 7,5 ?? " "+FILL(" ",68)+" "
@ 8,5 ?? " "+FILL(" ",68)+" "
@ 9,5 ?? " "+FILL(" ",68)+" "
@ 10,5 ?? " "+FILL(" ",68)+" "
@ 11,5 ?? " "+FILL(" ",68)+" "
CANVAS ON

IF IEMPTY("ANSWER") THEN
    @ 8,6 ?? FORMAT("W68,AC","NO MATCHES FOUND")
    @ 9,6 ?? FORMAT("W68,AC","    Press any key to continue...")
    GC = GETCHAR()
    RETURN
ELSE @ 8,6 ?? FORMAT("W68,AC",STRVAL(NRECORDS("ANSWER"))+" "+SHOWPRETTY1+" FOUND")
    @ 9,6 ?? FORMAT("W68,AC","Press any key to view "+SHOWPRETTY2)
    GC = GETCHAR()
ENDIF

;VIEW RECORDS FOUND
;VIEW RECORDS FOUND
CLEARALL
RENAME "ANSWER" "TANI"
COPYFORM "S\D\ANI" "12" "TANI" "12"
MasterView("TANI","12")
DELETE "TANI"
CLEARALL

ENDPROC
WRITELIB LIBNAME SumStudySearch
RELEASE PROCS SumStudySearch
;*****
;* END MODULE: x.x SumStudySearch()
;*****/

```

```

; /*****
; * PROCEDURE: InvStudySearch()
; * MODULE:
; * PURPOSE:
; * CALLED BY:
; * CALLS:
; * INPUT ASSERTION:
; * OUTPUT ASSERTION:
; * GLOBAL VARIABLES:
; * AUTHOR: JC Prather
; * DATE: August 1991
; *****/
PROC InvStudySearch()
    PRIVATE SEARCHTITLE, SEARCHPROMPT, SEARCHWIDTH, SEARCHVAL, X, L, J, SHOWPRETTY1,
        SHOWPRETTY2, GC, C

    SEARCHTITLE = "INVENTORY SEARCH BY STUDY NAME"
    SEARCHPROMPT = "STUDY NAME:"
    SEARCHWIDTH = 70

    WHILE TRUE
        SEARCHVAL = SearchBox(SEARCHTITLE, SEARCHPROMPT, SEARCHWIDTH)

        CURSOR OFF
        SWITCH
            CASE SEARCHVAL = "" OR SEARCHVAL = " ": RETURN

            CASE SEARCHVAL = "F1": ;GET PopUp OF CHOICES
                ;***** put up working message
                STYLE ATTRIBUTE HIGHLIGHT
                @ 24,59 ?? " ONE MOMENT PLEASE"
                STYLE ATTRIBUTE HIGHLIGHT+128
                @ 24,77 ?? "... "
                CANVAS ON
                CANVAS OFF
                ;****remove menu instructions and maintitle
                PAINTCANVAS ATTRIBUTE BKCOLOR 0,0,3,79
                PAINTCANVAS ATTRIBUTE BKCOLOR 24,0,24,79
                CLEARALL
                ;initialize vars
                ESCAPEFLAG = "N"
                PRMPT1 = FORMAT("W80,AC","F2-Execute search for items selected
                    Esc-Return to menu")
                PRMPT2 = FORMAT("W80,AC","ENTER-Select or un-select item          CURSOR
                    KEYS,PGUP,PGDN,HOME,END-Scroll list")
                ;GET POSSIBLE CHOICES TO ANSWER FOR PopUp

                {ASK}
                TYPEIN "S\D\STU"
                ENTER
                MOVETO FIELD "NRA_STUDY_NAME"
                CHECK
                DO IT!
                ;COPY TO TEMP DB
                COPY "ANSWER" "W\TEMP"

                ;ADD NO. OF OCCURANCES:
                EMPTY "W\TEXT"
                CLEARALL
                {ASK}
                TYPEIN "S\D\STU"

```

```

ENTER
MOVETO FIELD "LAB_ID"
EXAMPLE
TYPEIN "A"
MOVETO FIELD "STUDY_ID"
EXAMPLE
TYPEIN "B"
MOVETO FIELD "NRA_STUDY_NAME"
CHECK
{ASK}
TYPEIN "S\\D\\ANI"
ENTER
MOVETO FIELD "LAB_ID"
EXAMPLE
TYPEIN "A"
MOVETO FIELD "STUDY_ID"
EXAMPLE
TYPEIN "B"
MOVETO FIELD "ANIMAL_ID"
TYPEIN "CALC COUNT"
DO_IT!
{Modify} {Restructure} {W\\Temp} Down "Count of ANIMAL_ID" Right "A30"
Do_It!
{Modify} {Restructure} {answer} Down Right Right Backspace "A30" Do_it!
CLEARALL
{ASK}
TYPEIN "W\\TEMP"
ENTER
MOVETO FIELD "NRA_STUDY_NAME"
EXAMPLE
TYPEIN "A"
MOVETO FIELD "Count of ANIMAL_ID"
TYPEIN "CHANGETO "
EXAMPLE
TYPEIN "B"
{ASK}
TYPEIN "ANSWER"
ENTER
MOVETO FIELD "NRA_STUDY_NAME"
EXAMPLE
TYPEIN "A"
MOVETO FIELD "Count of ANIMAL_ID"
EXAMPLE
TYPEIN "B"
DO_IT!

CLEARALL
{ASK}
TYPEIN "W\\TEMP"
ENTER
MOVETO FIELD "Count of ANIMAL_ID"
TYPEIN "BLANK, CHANGETO 0"
DO_IT!

EDIT "W\\TEMP"
SCAN
MOVETO FIELD "NRA_STUDY_NAME"
TITLELENGTH = LEN([])
IF TITLELENGTH > 65 THEN
  [] = SUBSTR([],1,50)+
    "... "+
    SUBSTR([],TITLELENGTH-10,TITLELENGTH)+

```

```

        " | "+
        [Count of ANIMAL_ID]
ELSE
    [] = []+
    ". "+
    SPACES(65 - TITLELENGTH) +
    " | "+
    [Count of ANIMAL_ID]
ENDIF
ENDSCAN
DO_IT!

;DO PopUp
SetPopUp("W\\TEMP","NRA_STUDY_NAME")
DELETE "W\\TEMP"
PAINTCANVAS FILL " " ATTRIBUTE BKCOLOR 24,0,24,79
C=MAX(INT((80-WIDTH)/2)-3,0)
PopUp(4,C,14,1,"          NRA STUDY NAME          | # animals
available",PRMPT1,PRMPT2)
IF ESCAPEFLAG = "Y" THEN
    RELEASE VARS ESCAPEFLAG,ITEM,WIDTH
    RETURN
ELSE RELEASE VARS ESCAPEFLAG
ENDIF

;****DO SEARCH:
;***** put up working message
STYLE ATTRIBUTE HIGHLIGHT
@ 24,48 ?? " SEARCHING, ONE MOMENT PLEASE"
STYLE ATTRIBUTE HIGHLIGHT+128
@ 24,77 ?? "... "
CLEARALL
CANVAS ON
CANVAS OFF
CURSOR OFF
;***
(ASK)
TYPEIN "S\\D\\STU"
ENTER
L=0
FOR X FROM 1 TO ARRAYSIZE(ITEM)
    IF SUBSTR(ITEM[X],1,1) = "*" THEN
        MOVETO FIELD "LAB_ID"
        EXAMPLE
        L = L + 1
        TYPEIN "A"+STRVAL(L)
        MOVETO FIELD "STUDY_ID"
        EXAMPLE
        TYPEIN "B"+STRVAL(L)
        MOVETO FIELD "NRA_STUDY_NAME"
        TYPEIN "\""+SUBSTR(ITEM[X],2,SEARCH(".",ITEM[X])-2)+"\"..."
        DOWN
    EN)IF
ENDFOR
UP
IF ISBLANK([]) THEN RETURN
ENDIF
(ASK)
TYPEIN "S\\D\\ANI"
ENTER
FOR J FROM 1 TO L

```

```

        MOVETO FIELD "LAB_ID"
        EXAMPLE
        TYPEIN "A"+STRVAL(J)
        MOVETO FIELD "STUDY_ID"
        EXAMPLE
        TYPEIN "B"+STRVAL(J)
        MOVETO FIELD "ANIMAL_ID"
        EXAMPLE
        TYPEIN "C"+STRVAL(J)
        MOVETO FIELD "#"
        DOWN
    ENDFOR
    (ASK)
    TYPEIN "I\\D\\DETAIL"
    ENTER
    FOR K FROM 1 TO L
        CHECK
        MOVETO FIELD "ANIMAL_ID"
        EXAMPLE
        TYPEIN "C"+STRVAL(K)
        MOVETO FIELD "#"
        DOWN
    ENDFOR
    DO_IT!
    QUITLOOP

OTHERWISE: ;SEARCH ON SEARCHVAL
;***** put up working message
STYLE ATTRIBUTE HIGHLIGHT
@ 24,48 ?? " SEARCHING, ONE MOMENT PLEASE"
STYLE ATTRIBUTE HIGHLIGHT+128
@ 24,77 ?? "... "
CLEARALL
CANVAS ON
CANVAS OFF
CURSOR OFF
(ASK)
;***
TYPEIN "S\\D\\STU"
ENTER
MOVETO FIELD "LAB_ID"
EXAMPLE
TYPEIN "A"
MOVETO FIELD "STUDY_ID"
EXAMPLE
TYPEIN "B"
MOVETO FIELD "NRA_STUDY_NAME"
TYPEIN SEARCHVAL
(ASK)
TYPEIN "S\\D\\ANI"
ENTER
MOVETO FIELD "LAB_ID"
EXAMPLE
TYPEIN "A"
MOVETO FIELD "STUDY_ID"
EXAMPLE
TYPEIN "B"
MOVETO FIELD "ANIMAL_ID"
EXAMPLE
TYPEIN "C"
(ASK)
TYPEIN "I\\D\\DETAIL"

```

```

ENTER
CHECK
MOVETO FIELD "ANIMAL_ID"
EXAMPLE
TYPEIN "C"
DO_IT!
QUITLOOP

ENDSWITCH

ENDWHILE

;SHOW HOW MANY FOUND

STYLE ATTRIBUTE HIGHLIGHT
SHOWPRETTY1 = "MATCH"
SHOWPRETTY2 = "RECORD"

IF NRECORDS("ANSWER") > 1 THEN
    SHOWPRETTY1 = "MATCHES"
    SHOWPRETTY2 = "RECORDS"
ENDIF

;DRAW BOX FOR MESSAGE
PAINTCANVAS ATTRIBUTE BKCOLOR 24,0,24,79
@ 6,5 ?? " "+FILL(" ",68)+" "
@ 7,5 ?? " "+FILL(" ",68)+" "
@ 8,5 ?? " "+FILL(" ",68)+" "
@ 9,5 ?? " "+FILL(" ",68)+" "
@ 10,5 ?? " "+FILL(" ",68)+" "
@ 11,5 ?? " "+FILL(" ",68)+" "
CANVAS ON

IF IEMPTY("ANSWER") THEN
    @ 8,6 ?? FORMAT("W68,AC","NO MATCHES FOUND")
    @ 9,6 ?? FORMAT("W68,AC"," Press any key to continue...")
    GC = GETCHAR()
    RETURN
ELSE @ 8,6 ?? FORMAT("W68,AC",STRVAL(NRECORDS("ANSWER"))+" "+SHOWPRETTY1+" FOUND")
    @ 9,6 ?? FORMAT("W68,AC","Press any key to view "+SHOWPRETTY2)
    GC = GETCHAR()
ENDIF

;VIEW RECORDS FOUND
;VIEW RECORDS FOUND
CLEARALL
RENAME "ANSWER" "TDETAIL"
COPYFORM "I\\D\\DETAIL" "2" "TDETAIL" "2"
MasterView("TDETAIL","2")
DELETE "TDETAIL"
CLEARALL

ENDPROC
WRITELIB LIBNAME InvStudySearch
RELEASE PROCS InvStudySearch
;*****
;* END MODULE: x.x InvStudySearch()
;*****/

```

UTIL\MENUMADE.SC

MENUMADE is the script which is produced by MAKEMENU when it processes the two tables MENUCHOI and MENULEVE. It is re-written each time the script MAKEMENU is played. This version was created on October 16, 1991.

```
LIBNAME = "UTIL\MENUMADE"
CREATELIB LIBNAME

PROC INIT()
  RESET
  AUTOLIB = "UTIL\MENUUTIL,UTIL\MENUMADE"
  ARRAY PREVX[20]
  ARRAY PREVY[20]
  ARRAY PREVNC[20]
  ARRAY PREVPW[20]
  ARRAY PREVCL1[20]
  ARRAY PREVCL2[20]
  ARRAY PREVN[20]
  ARRAY PREVML[20]
  LEVEL = 1
  BEGINX = 4
  BEGINY = -1
  POPWIDTH = 65
  BKCOLOR = 119
  HIGHLIGHT = 79
  IF MONITOR() <> "Color" THEN
    BKCOLOR = 0
    HIGHLIGHT = 112
  ENDIF
  COLOR1 = 10
  COLOR2 = 20
  MENUNAME = ""
  NUMBERCHOICES = 4
ENDPROC
WRITELIB LIBNAME INIT;
RELEASE PROCS INIT;

PROC BACKONE()
  LEVEL = LEVEL - 1
  BEGINX = PREVX[LEVEL]
  BEGINY = PREVY[LEVEL]
  NUMBERCHOICES = PREVNC[LEVEL]
  POPWIDTH = PREVPW[LEVEL]
  COLOR1 = PREVCL1[LEVEL]
  COLOR2 = PREVCL2[LEVEL]
  MENUNAME = PREVML[LEVEL]
  N = PREVN[LEVEL]
  RESET
ENDPROC;
WRITELIB LIBNAME BACKONE;
RELEASE PROCS BACKONE;

PROC FIRSTONE()
  SEANSBACKFLAG = LEVEL-3
  BACKONE()
```

```

ENDPROC
WRITELIB LIBNAME FIRSTONE
RELEASE PROCS FIRSTONE

```

```

PROC QUITPROC()
  PAINTCANVAS ATTRIBUTE BKCOLOR 0,0,3,79
  PAINTCANVAS ATTRIBUTE BKCOLOR 24,0,24,79
  IF ISRUNTIME() = TRUE THEN
    SHOWMENU "DOS": "EXIT TO DOS",
             "RETURN": "RETURN TO MENU"
    DEFAULT "RETURN"
    TO ANS
  ELSE
    SHOWMENU "PARADOX": "EXIT TO PARADOX",
             "DOS": "EXIT TO DOS",
             "RETURN": "RETURN TO MENU"
    DEFAULT "RETURN"
    TO ANS
  ENDIF
  SWITCH
  CASE ANS = "PARADOX" :
    CLEARALL
    RELEASE VARS ALL
    REFRESH
    QUIT
  CASE ANS = "DOS":
    EXIT
  CASE ANS = "RETURN":
    RETURN
  ENDSWITCH
ENDPROC
WRITELIB LIBNAME QUITPROC
RELEASE PROCS QUITPROC

```

```

PROC MENU0()
  MAINTITLE = "National Radiobiology Archives"
  MAINTITLECOLOR1 = 30
  MAINTITLECOLOR2 = 23
  IF ISASSIGNED(PREVN[2]) THEN N = PREVN[2]
  ELSE N = 1
  ENDIF
  INIT()
  PAINTCANVAS FILL " " ATTRIBUTE BKCOLOR 0,0,24,79
  PREVX[LEVEL] = BEGINX
  PREVY[LEVEL] = BEGINY
  PREVC[LEVEL] = NUMBERCHOICES
  PREVPW[LEVEL] = POPWIDTH
  PREVCL1[LEVEL] = COLOR1
  PREVCL2[LEVEL] = COLOR2
  PREVML[LEVEL] = MENUName
  PREVN[LEVEL] = N
  LEVEL = LEVEL + 1
  BEGINX = PREVX[LEVEL-1]+2 ;
  IF PREVY[LEVEL-1] > 8 THEN
    BEGINY = PREVY[LEVEL-1] ELSE
    BEGINY = PREVY[LEVEL-1]+2 ;
  ENDIF
  NUMBERCHOICES =7
  IF BEGINX + NUMBERCHOICES + 4 > 24 THEN
    BEGINX = 3
  ENDIF

```

```

ENDIF
    POPWIDTH = 69
    COLOR1 = 23
    COLOR2 = 30
    MENUName = "MAIN MENU"
    PREVX[LEVEL] = BEGINX
    PREVY[LEVEL] = BEGINY
    PREVNC[LEVEL] = NUMBERCHOICES
    PREVPW[LEVEL] = POPWIDTH
    PREVCL1[LEVEL] = COLOR1;
    PREVCL2[LEVEL] = COLOR2;
    PREVN[LEVEL] = N
    PREVML[LEVEL] = MENUName;
    WHILE (TRUE)
    ARRAY C[7]
    ARRAY D[7]
    ARRAY NICK[7]
    C[1] = "Information"
    D[1] = "Description of the National Radiobiology Archives."
    C[2] = "Summary"
    D[2] = "Access the animal related Dose-effects database."
    C[3] = "Inventory"
    D[3] = "Access Specimen and Documents Inventory database."
    C[4] = "Bibliography"
    D[4] = "Access the Bibliographic citation database."
    C[5] = "Detail"
    D[5] = "Description of the Detailed database tables."
    C[6] = "HELP"
    D[6] = "How to use this software."
    C[7] = "Quit"
    D[7] = "Exit, return to DOS."
    IF ISASSIGNED(N) = FALSE THEN
        N = 1
    ELSE
        N = N
    ENDIF
    LAST = ARRAYSIZE(C) ; LETS THE PROCEDURE KNOW THE 'BOTTOM' MENU CHOICE.
    IF ISASSIGNED(SEANSBACKFLAG) THEN
        IF SEANSBACKFLAG <> 0 THEN
            SEANSBACKFLAG = SEANSBACKFLAG-1
            BACKONE()
            QUITLOOP
        ENDIF
    ENDIF
    ENDIF
    FILLMENU()
    SELMENU()
    IF SEL = "F1MENUHELP" THEN
    SWITCH
    CASE N=1:
    HELPMESSAGE = "Information will pop up a window which describes the National Radiobiology Archives."
    CASE N=2:
    HELPMESSAGE = "Summary will pop up the Summary menu. The Summary database contains information about individual animals."
    CASE N=3:
    HELPMESSAGE = "Inventory will pop up the Inventory menu. The Inventory database describes the NRA collection of tissue specimens, slides, b"
    "locks, radiographs, clinical records, and bibliographic documents."
    CASE N=4:
    HELPMESSAGE = "Bibliography will pop up the Bibliography menu. The Bibliography database describes publications in the field of Radiobiolog"
    "y. Reprints of many of these are in the NRA Inventory."

```

```

CASE N=5:
HELPMESSAGE = "Detail will pop up the Detail menu. The NRA detail database is in the
format of the specified laboratory. It is not distrib"
uted in this demonstration."
CASE N=6:
HELPMESSAGE = "HELP will display a menu of choices of types of information which describe
this demonstration."
CASE N=7:
HELPMESSAGE = "Quit is used to exit the demonstration."
ENDSWITCH
IF HELPMESSAGE = "" THEN
HELPMESSAGE = "No help available for this menu choice."
ENDIF
STYLE ATTRIBUTE SYSCOLOR(22)
CANVAS OFF
@ 5,5 ?? " " + FILL(" ",68) + " "
@ 5,36 ?? " HELP "
@ 6,5 ?? " " + FILL(" ",68) + " "
@ 7,5 ?? " " + FILL(" ",68) + " "
@ 8,5 ?? " " + FILL(" ",68) + " "
@ 9,5 ?? " " + FILL(" ",68) + " "
@ 10,5 ?? " " + FILL(" ",68) + " "
@ 11,5 ?? " " + FILL(" ",68) + " "
STYLE ATTRIBUTE SYSCOLOR(24)
@ 6,6 ?? FORMAT("W68,AC","Menu Choice: "+c[n])
STYLE ATTRIBUTE SYSCOLOR(21)
SWITCH
CASE LEN(HELPMESSAGE)<69:
@ 8,6 ?? FORMAT("W68,AC",HELPMESSAGE)
OTHERWISE:
Q=0
ARRAY PRINTLINE[3]
WHILE LEN(HELPMESSAGE) > 67
FOR X FROM 1 TO 67
IF SUBSTR(HELPMESSAGE,67-X,1) = " " THEN
Q = Q + 1
PRINTLINE[Q] = SUBSTR(HELPMESSAGE,1,67-X)
HELPMESSAGE = SUBSTR(HELPMESSAGE,67-X+1,LEN(HELPMESSAGE))
QUITLOOP
ENDIF
ENDFOR
ENDWHILE
SWITCH
CASE Q = 3:
@ 7,7 ?? PRINTLINE[1]
@ 8,7 ?? PRINTLINE[2]
@ 9,7 ?? PRINTLINE[3]
@ 10,7 ?? HELPMESSAGE
CASE Q = 2:
@ 8,7 ?? PRINTLINE[1]
@ 9,7 ?? PRINTLINE[2]
@ 10,7 ?? HELPMESSAGE
CASE Q = 1:
@ 8,7 ?? PRINTLINE[1]
@ 9,7 ?? HELPMESSAGE
ENDSWITCH
RELEASE VARS Q,PRINTLINE,HELPMESSAGE
ENDSWITCH
STYLE ATTRIBUTE COLOR3
@ 24,0 ?? FORMAT("W80,AC","Press any key to exit help.")
CANVAS ON
GC = GETCHAR()

```

```

REDRAWPOPBOX(LEVEL)
ENDIF
IF SEL = "Esc" THEN
  QUITPROC()
ELSE
  SWITCH
    CASE SEL = "Esc" :      LEVEL = LEVEL - 1
                           BEGINX = PREVX[LEVEL]
                           BEGINY = PREVY[LEVEL]
                           NUMBERCHOICES = PREVNC[LEVEL]
                           POPWIDTH = PREVPW[LEVEL]
                           COLOR1 = PREVCL1[LEVEL]
                           COLOR2 = PREVCL2[LEVEL]
                           N = PREVN[LEVEL]
                           RESET
                           QUITLOOP
    CASE SEL = "Information" : AUTOLIB = "UTIL\\DEMOUTIL"
TEXTEDITOR("HELP\\NRA.TXT","NATIONAL RADIOBIOLOGY ARCHIVES")
      AUTOLIB = "UTIL\\MENUMADE,UTIL\\MENUUTIL"
      REDRAWPOPBOX(LEVEL)
    CASE SEL = "Summary" :
MENU2()
    CASE SEL = "Inventory" :
MENU3()
    CASE SEL = "Bibliography" :
MENU4()
    CASE SEL = "Detail" :
MENU5()
    CASE SEL = "HELP" :
MENU6()
    CASE SEL = "Quit" :
QUITPROC()
      REDRAWPOPBOX(LEVEL)
    ENDSWITCH
  ENDIF

  ENDWHILE
ENDPROC;
WRITELIB LIBNAME MENU0;
RELEASE PROCS MENU0;

PROC MENU2()
  PREVX[LEVEL] = BEGINX
  PREVY[LEVEL] = BEGINY
  PREVNC[LEVEL] = NUMBERCHOICES
  PREVPW[LEVEL] = POPWIDTH
  PREVCL1[LEVEL] = COLOR1
  PREVCL2[LEVEL] = COLOR2
  PREVML[LEVEL] = MENUNAME
  PREVN[LEVEL] = N
  N=1
  LEVEL = LEVEL + 1
  BEGINX = PREVX[LEVEL-1]+2 ;
IF PREVY[LEVEL-1] > 8 THEN
  BEGINY = PREVY[LEVEL-1] ELSE
  BEGINY = PREVY[LEVEL-1]+2 ;
ENDIF
NUMBERCHOICES =5
IF BEGINX + NUMBERCHOICES + 4 > 24 THEN
  BEGINX = 3
ENDIF

```

```

        POPWIDTH = 69
        COLOR1 =23
        COLOR2 =30
        MENUNAME = "SUMMARY DATABASE"
        PREVX[LEVEL] = BEGINX
        PREVY[LEVEL] = BEGINY
        PREVNC[LEVEL] = NUMBERCHOICES
        PREVPW[LEVEL] = POPWIDTH
        PREVCL1[LEVEL] = COLOR1;
        PREVCL2[LEVEL] = COLOR2;
        PREVN[LEVEL] = N
        PREVML[LEVEL] = MENUNAME;
        WHILE (TRUE)
        ARRAY C[5]
        ARRAY D[5]
        ARRAY NICK[5]
        C[1] = "Holdings"
        D[1] = "Read description of NRA Summary Database holdings."
        C[2] = "Browse"
        D[2] = "Browse Dose-effects Summary Database."
        C[3] = "Searches"
        D[3] = "Retrieve subset of Dose-effects Summary Database."
        C[4] = "Return"
        D[4] = "Return to previous menu."
        C[5] = "Quit"
        D[5] = "Exit, return to DOS."
        IF ISASSIGNED(N) = FALSE THEN
            N = 1
        ELSE
            N = N
        ENDIF
        LAST = ARRAYSIZE(C) ; LETS THE PROCEDURE KNOW THE 'BOTTOM' MENU CHOICE.
        IF ISASSIGNED(SEANSBACKFLAG) THEN
            IF SEANSBACKFLAG <> 0 THEN
                SEANSBACKFLAG = SEANSBACKFLAG-1
                BACKONE()
                QUITLOOP
            ENDIF
        ENDIF
        FILLMENU()
        SELMENU()
        IF SEL = "F1MENUHELP" THEN
        SWITCH
        CASE N=1:
            HELPMESSAGE = "Holdings will pop up a window which describes the scope of donations to the
            NRA Summary database."
        CASE N=2:
            HELPMESSAGE = "Browse will bring up a full screen form view of the NRA Summary database.
            The form is oriented to individual animals, with w"+
            "indows showing the ANIMAL table linked to scrolling windows of TISSUE EFFECTS and TISSUE
            DOSE."
        CASE N=3:
            HELPMESSAGE = "Searches will bring up a search menu for obtaining subsets of the NRA
            Summary database."
        CASE N=4:
            HELPMESSAGE = "Return will erase the current menu, returning the viewer to the next higher
            menu."
        CASE N=5:
            HELPMESSAGE = "Quit is used to exit the demonstration."
        ENDSWITCH
        IF HELPMESSAGE = "" THEN
            HELPMESSAGE = "No help available for this menu choice."

```

```

ENDIF
STYLE ATTRIBUTE SYSCOLOR(22)
CANVAS OFF
@ 5,5 ?? " " + FILL(" ",68) + " "
@ 5,36 ?? " HELP "
@ 6,5 ?? " " + FILL(" ",68) + " "
@ 7,5 ?? " " + FILL(" ",68) + " "
@ 8,5 ?? " " + FILL(" ",68) + " "
@ 9,5 ?? " " + FILL(" ",68) + " "
@ 10,5 ?? " " + FILL(" ",68) + " "
@ 11,5 ?? " " + FILL(" ",68) + " "
STYLE ATTRIBUTE SYSCOLOR(24)
@ 6,6 ?? FORMAT("W68,AC","Menu Choice: "+c[n])
STYLE ATTRIBUTE SYSCOLOR(21)
SWITCH
CASE LEN(HELPMESSAGE)<69:
  @ 8,6 ?? FORMAT("W68,AC",HELPMESSAGE)
OTHERWISE:
Q=0
ARRAY PRINTLINE[3]
WHILE LEN(HELPMESSAGE) > 67
  FOR X FROM 1 TO 67
    IF SUBSTR(HELPMESSAGE,67-X,1) = " " THEN
      Q = Q + 1
      PRINTLINE[Q] = SUBSTR(HELPMESSAGE,1,67-X)
      HELPMESSAGE = SUBSTR(HELPMESSAGE,67-X+1,LEN(HELPMESSAGE))
      QUITLOOP
    ENDIF
  ENDFOR
ENDWHILE
SWITCH
CASE Q = 3:
  @ 7,7 ?? PRINTLINE[1]
  @ 8,7 ?? PRINTLINE[2]
  @ 9,7 ?? PRINTLINE[3]
  @ 10,7 ?? HELPMESSAGE
CASE Q = 2:
  @ 8,7 ?? PRINTLINE[1]
  @ 9,7 ?? PRINTLINE[2]
  @ 10,7 ?? HELPMESSAGE
CASE Q = 1:
  @ 8,7 ?? PRINTLINE[1]
  @ 9,7 ?? HELPMESSAGE
ENDSWITCH
RELEASE VARS Q,PRINTLINE,HELPMESSAGE
ENDSWITCH
STYLE ATTRIBUTE COLOR3
@ 24,0 ?? FORMAT("W80,AC","Press any key to exit help.")
CANVAS ON
GC = GETCHAR()
REDRAWPOPBOX(LEVEL)
ENDIF
IF SEL = "Esc" THEN
  LEVEL = LEVEL - 1
  BEGINX = PREVX[LEVEL]
  BEGINY = PREVY[LEVEL]
  NUMBERCHOICES = PREVNC[LEVEL]
  POPWIDTH = PREVPW[LEVEL]
  COLOR1 = PREVCL1[LEVEL]
  COLOR2 = PREVCL2[LEVEL]
  MENUNAME = PREVML[LEVEL]
  N = PREVN[LEVEL]

```

```

        RESET
        QUITLOOP
    ELSE
    SWITCH
        CASE SEL = "Esc" :
            LEVEL = LEVEL - 1
            BEGINX = PREVX[LEVEL]
            BEGINY = PREVY[LEVEL]
            NUMBERCHOICES = PREVNC[LEVEL]
            POPWIDTH = PREVPW[LEVEL]
            COLOR1 = PREVCL1[LEVEL]
            COLOR2 = PREVCL2[LEVEL]
            N = PREVN[LEVEL]
            RESET
            QUITLOOP

        CASE SEL = "Holdings" : AUTOLIB = "UTIL\\DEMOUTIL"
        TEXTEDITOR("HELP\\SUMMARY.TXT","NRA SUMMARY DATABASE HOLDINGS")
            AUTOLIB = "UTIL\\MENUMADE,UTIL\\MENUUTIL"
            REDRAWPOPBOX(LEVEL)

        CASE SEL = "Browse" : AUTOLIB = "UTIL\\DEMOUTIL"
        MASTERVIEW("S\\D\\ANI","12")
            AUTOLIB = "UTIL\\MENUMADE,UTIL\\MENUUTIL"
            REDRAWPOPBOX(LEVEL)

        CASE SEL = "Searches" :
        MENU23()

        CASE SEL = "Return" :
        BACKONE()

        CASE SEL = "Quit" :
            QUITLOOP

    QUITPROC()
        REDRAWPOPBOX(LEVEL)
    ENDSWITCH
    ENDIF

    ENDWHILE
    ENDPROC;
    WRITELIB LIBNAME MENU2;
    RELEASE PROCS MENU2;

PROC MENU3()
    PREVX[LEVEL] = BEGINX
    PREVY[LEVEL] = BEGINY
    PREVNC[LEVEL] = NUMBERCHOICES
    PREVPW[LEVEL] = POPWIDTH
    PREVCL1[LEVEL] = COLOR1
    PREVCL2[LEVEL] = COLOR2
    PREVM[LEVEL] = MENUNAME
    PREVN[LEVEL] = N
    N=1
        LEVEL = LEVEL + 1
        BEGINX = PREVX[LEVEL-1]+2 ;
    IF PREVY[LEVEL-1] > 8 THEN
        BEGINY = PREVY[LEVEL-1] ELSE
        BEGINY = PREVY[LEVEL-1]+2 ;
    ENDIF
    NUMBERCHOICES =5
    IF BEGINX + NUMBERCHOICES + 4 > 24 THEN
        BEGINX = 3
    ENDIF
        POPWIDTH = 69
        COLOR1 =23
        COLOR2 =30

```

```

    MENUNAME = "INVENTORY DATABASE"
    PREVX[LEVEL] = BEGINX
    PREVY[LEVEL] = BEGINY
    PREVNC[LEVEL] = NUMBERCHOICES
    PREVPW[LEVEL] = POPWIDTH
    PREVCL1[LEVEL] = COLOR1;
    PREVCL2[LEVEL] = COLOR2;
    PREVN[LEVEL] = N
    PREVML[LEVEL] = MENUNAME;
    WHILE (TRUE)
ARRAY C(5)
ARRAY D(5)
ARRAY NICK(5)
    C[1] = "Holdings"
    D[1] = "Read description of NRA Inventory Database."
    C[2] = "Browse"
    D[2] = "Browse Specimen and Document Inventory Database."
    C[3] = "Searches"
    D[3] = "Retrieve subset of Inventory Database."
    C[4] = "Return"
    D[4] = "Return to previous menu."
    C[5] = "Quit"
    D[5] = "Exit, return to DOS."
    IF ISASSIGNED(N) = FALSE THEN
        N = 1
    ELSE
        N = N
    ENDIF
    LAST = ARRAYSIZE(C) ; LETS THE PROCEDURE KNOW THE 'BOTTOM' MENU CHOICE.
    IF ISASSIGNED(SEANSBACKFLAG) THEN
        IF SEANSBACKFLAG <> 0 THEN
            SEANSBACKFLAG = SEANSBACKFLAG-1
            BACKONE()
            QUITLOOP
        ENDIF
    ENDIF
    FILLMENU()
    SELMENU()
    IF SEL = "F1MENUHELP" THEN
        SWITCH
        CASE N=1:
            HELPMESSAGE = "Holdings will pop up a window showing information about the scope of the NRA
Inventory."
        CASE N=2:
            HELPMESSAGE = "Browse will bring up a full screen form showing a view of the records in the
Inventory database. The view is of a MASTER rec"+
"ord linked by BAR_CODE to a scrolling DETAIL window and a one line LOCATION window."
        CASE N=3:
            HELPMESSAGE = "Searches will pop up the Inventory search menu."
        CASE N=4:
            HELPMESSAGE = "Return will erase the current menu, returning the viewer to the next higher
menu."
        CASE N=5:
            HELPMESSAGE = "Quit is used to exit the demonstration."
        END SWITCH
        IF HELPMESSAGE = "" THEN
            HELPMESSAGE = "No help available for this menu choice."
        ENDIF
        STYLE ATTRIBUTE SYSCOLOR(22)
        CANVAS OFF
        @ 5,5 ?? "F"+FILL("-",68)+"J"
        @ 5,36 ?? "HELP "

```

```

@ 6,5 ?? " "+FILL(" ",68)+" "
@ 7,5 ?? " "+FILL(" ",68)+" "
@ 8,5 ?? " "+FILL(" ",68)+" "
@ 9,5 ?? " "+FILL(" ",68)+" "
@ 10,5 ?? " "+FILL(" ",68)+" "
@ 11,5 ?? " "+FILL(" ",68)+" "
STYLE ATTRIBUTE SYSCOLOR(24)
@ 6,6 ?? FORMAT("W68,AC","Menu Choice: "+c[n])
STYLE ATTRIBUTE SYSCOLOR(21)
SWITCH
CASE LEN(HELPMESSAGE)<69:
  @ 8,6 ?? FORMAT("W68,AC",HELPMESSAGE)
OTHERWISE:
Q=0
ARRAY PRINTLINE[3]
WHILE LEN(HELPMESSAGE) > 67
  FOR X FROM 1 TO 67
    IF SUBSTR(HELPMESSAGE,67-X,1) = " " THEN
      Q = Q + 1
      PRINTLINE[Q] = SUBSTR(HELPMESSAGE,1,67-X)
      HELPMESSAGE = SUBSTR(HELPMESSAGE,67-X+1,LEN(HELPMESSAGE))
      QUITLOOP
    ENDIF
  ENDFOR
ENDWHILE
SWITCH
CASE Q = 3:
  @ 7,7 ?? PRINTLINE[1]
  @ 8,7 ?? PRINTLINE[2]
  @ 9,7 ?? PRINTLINE[3]
  @ 10,7 ?? HELPMESSAGE
CASE Q = 2:
  @ 8,7 ?? PRINTLINE[1]
  @ 9,7 ?? PRINTLINE[2]
  @ 10,7 ?? HELPMESSAGE
CASE Q = 1:
  @ 8,7 ?? PRINTLINE[1]
  @ 9,7 ?? HELPMESSAGE
ENDSWITCH
RELEASE VARS Q,PRINTLINE,HELPMESSAGE
ENDSWITCH
STYLE ATTRIBUTE COLOR3
@ 24,0 ?? FORMAT("W80,AC","Press any key to exit help.")
CANVAS ON
GC = GETCHAR()
REDRAWPOPOBOX(LEVEL)
ENDIF
IF SEL = "Esc" THEN
  LEVEL = LEVEL - 1
  BEGINX = PREVX[LEVEL]
  BEGINY = PREVY[LEVEL]
  NUMBERCHOICES = PREVNC[LEVEL]
  POPWIDTH = PREVPW[LEVEL]
  COLOR1 = PREVCL1[LEVEL]
  COLOR2 = PREVCL2[LEVEL]
  MENUNAME = PREVML[LEVEL]
  N = PREVN[LEVEL]
  RESET
  QUITLOOP
ELSE
  SWITCH
    CASE SEL = "Esc" :      LEVEL = LEVEL - 1

```

```

        BEGINX = PREVX[LEVEL]
        BEGINY = PREVY[LEVEL]
        NUMBERCHOICES = PREVNC[LEVEL]
        POPWIDTH = PREVPW[LEVEL]
        - COLOR1 = PREVCL1[LEVEL]
        COLOR2 = PREVCL2[LEVEL]
        N = PREVN[LEVEL]
        RESET
        QUITLOOP

        CASE SEL = "Holdings" : AUTOLIB = "UTIL\DLMOUTIL"
TEXTEDITOR("HELP\INVENTOR.TXT","NRA INVENTORY DATABASE HOLDINGS")
        AUTOLIB = "UTIL\MENUMADE,UTIL\MENUUTIL"
        REDRAWPOPBOX(LEVEL)
        CASE SEL = "Browse" : AUTOLIB = "UTIL\DEMOUTIL"
MASTERVIEW("I\O\DETAIL", "2")
        AUTOLIB = "UTIL\MENUMADE,UTIL\MENUUTIL"
        REDRAWPOPBOX(LEVEL)
        CASE SEL = "Searches" :
MENU33()
        CASE SEL = "Return" :
BACKONE()
        CASE SEL = "Quit" :
QUITPROC()
        REDRAWPOPBOX(LEVEL)
        END SWITCH
        END IF

        END WHILE
        END PROC;
        WRITELIS LIBNAME MENU3;
        RELEASE PROCS MENU3;

PROC MENU4()
    PREVX[LEVEL] = BEGINX
    PREVY[LEVEL] = BEGINY
    PREVNC[LEVEL] = NUMBERCHOICES
    PREVPW[LEVEL] = POPWIDTH
    PREVCL1[LEVEL] = COLOR1
    PREVCL2[LEVEL] = COLOR2
    PREVML[LEVEL] = MENUName
    PREVN[LEVEL] = N
    N=1
        LEVEL = LEVEL + 1
        BEGINX = PREVX[LEVEL-1]+2 ;
    IF PREVY[LEVEL-1] > 8 THEN
        BEGINY = PREVY[LEVEL-1] ELSE
        BEGINY = PREVY[LEVEL-1]+2 ;
    END IF
    NUMBERCHOICES =5
    IF BEGINX + NUMBERCHOICES + 4 > 24 THEN
        BEGINX = 3
    END IF
        POPWIDTH = 69
        COLOR1 =23
        COLOR2 =30
        MENUName = "BIBLIOGRAPHY DATABASE"
        PREVX[LEVEL] = BEGINX
        PREVY[LEVEL] = BEGINY
        PREVNC[LEVEL] = NUMBERCHOICES
        PREVPW[LEVEL] = POPWIDTH

```

```

PREVCL1[LEVEL] = COLOR1;
PREVCL2[LEVEL] = COLOR2;
PREVN[LEVEL] = N
PREVML[LEVEL] = MENUNAME;
WHILE (TRUE)
ARRAY C[5]
ARRAY D[5]
ARRAY NICK[5]
C[1] = "Holdings"
D[1] = "Read description of NRA Bibliography Database."
C[2] = "Browse"
D[2] = "Browse Bibliographic citations."
C[3] = "Searches"
D[3] = "Retrieve subset of Bibliography database."
C[4] = "Return"
D[4] = "Return to previous menu."
C[5] = "Quit"
D[5] = "Exit, return to DOS."
IF ISASSIGNED(N) = FALSE THEN
    N = 1
ELSE
    N = N
ENDIF
LAST = ARRAYSIZE(C) ; LETS THE PROCEDURE KNOW THE 'BOTTOM' MENU CHOICE.
IF ISASSIGNED(SEANSBACKFLAG) THEN
    IF SEANSBACKFLAG <> 0 THEN
        SEANSBACKFLAG = SEANSBACKFLAG-1
        BACKONE()
        QUITLOOP
    ENDIF
ENDIF
FILLMENU()
SELMENU()
IF SEL = "F1MENUHELP" THEN
SWITCH
CASE N=1:
HELPMESSAGE = "Holdings will pop up a window showing information about the NRA
Bibliographic holdings."
CASE N=2:
HELPMESSAGE = "Browse will bring up a full screen form showing records in the Bibliography
database. The view shows each BOOK record linked"+
" by BAR_CODE to a scrolling window of AUTHORS and a one line window of LOCATION."
CASE N=3:
HELPMESSAGE = "Searches will pop up the Bibliography search menu which allows creation of
subsets of the Bibliography database."
CASE N=4:
HELPMESSAGE = "Return will erase the current menu, returning the viewer to the next higher
menu."
CASE N=5:
HELPMESSAGE = "Quit is used to exit the demonstration."
ENDSWITCH
IF HELPMESSAGE = "" THEN
    HELPMESSAGE = "No help available for this menu choice."
ENDIF
STYLE ATTRIBUTE SYSCOLOR(22)
CANVAS OFF
@ 5,5 ?? " "+FILL(" ",68)+"|"
@ 5,36 ?? " HELP "
@ 6,5 ?? " "+FILL(" ",68)+"|"
@ 7,5 ?? " "+FILL(" ",68)+"|"
@ 8,5 ?? " "+FILL(" ",68)+"|"
@ 9,5 ?? " "+FILL(" ",68)+"|"

```

```

@ 10,5 ?? "["+FILL(" ",68)+"]"
@ 11,5 ?? "L"+FILL("-",68)+"]"
STYLE ATTRIBUTE SYSCOLOR(24)
@ 6,6 ?? FORMAT("W68,AC","Menu Choice: "+c[n])
STYLE ATTRIBUTE SYSCOLOR(21)
SWITCH
CASE LEN(HELPMESSAGE)<69:
    @ 8,6 ?? FORMAT("W68,AC",HELPMESSAGE)
OTHERWISE:
Q=0
ARRAY PRINTLINE[3]
WHILE LEN(HELPMESSAGE) > 67
    FOR X FROM 1 TO 67
        IF SUBSTR(HELPMESSAGE,67-X,1) =" " THEN
            Q = Q + 1
            PRINTLINE[Q] = SUBSTR(HELPMESSAGE,1,67-X)
            HELPMESSAGE = SUBSTR(HELPMESSAGE,67-X+1,LEN(HELPMESSAGE))
            QUITLOOP
        ENDIF
    ENDFOR
ENDWHILE
SWITCH
CASE Q = 3:
    @ 7,7 ?? PRINTLINE[1]
    @ 8,7 ?? PRINTLINE[2]
    @ 9,7 ?? PRINTLINE[3]
    @ 10,7 ?? HELPMESSAGE
CASE Q = 2:
    @ 8,7 ?? PRINTLINE[1]
    @ 9,7 ?? PRINTLINE[2]
    @ 10,7 ?? HELPMESSAGE
CASE Q = 1:
    @ 8,7 ?? PRINTLINE[1]
    @ 9,7 ?? HELPMESSAGE
ENDSWITCH
RELEASE VARS Q,PRINTLINE,HELPMESSAGE
ENDSWITCH
STYLE ATTRIBUTE COLOR3
@ 24,0 ?? FORMAT("W80,AC","Press any key to exit help.")
CANVAS ON
GC = GETCHAR()
REDRAWPOPBOX(LEVEL)
ENDIF
IF SEL = "Esc" THEN
    LEVEL = LEVEL - 1
    BEGINX = PREVX[LEVEL]
    BEGINY = PREVY[LEVEL]
    NUMBERCHOICES = PREVNC[LEVEL]
    POPWIDTH = PREVPW[LEVEL]
    COLOR1 = PREVCL1[LEVEL]
    COLOR2 = PREVCL2[LEVEL]
    MENUNAME = PREVM[LEVEL]
    N = PREVN[LEVEL]

    RESET
    QUITLOOP
ELSE
SWITCH
CASE SEL = "Esc" :
    LEVEL = LEVEL - 1
    BEGINX = PREVX[LEVEL]
    BEGINY = PREVY[LEVEL]
    NUMBERCHOICES = PREVNC[LEVEL]
    POPWIDTH = PREVPW[LEVEL]

```

```

                                COLOR1 = PREVCL1[LEVEL]
                                COLOR2 = PREVCL2[LEVEL]
                                N = PREVN[LEVEL]
                                RESET
                                QUITLOOP
                                CASE SEL = "Holdings" : AUTOLIB = "UTIL\\DEMOUTIL"
TEXTEDITOR("HELP\\BIBLIOGR.TXT","NRA BIBLIOGRAPHY DATABASE HOLDINGS")
                                AUTOLIB = "UTIL\\MENU\\MADE,UTIL\\MENU\\UTIL"
                                REDRAWPOPBOX(LEVEL)
                                CASE SEL = "Browse" : AUTOLIB = "UTIL\\DEMOUTIL"
MASTERVIEW("B\\D\\BOOK","2")
                                AUTOLIB = "UTIL\\MENU\\MADE,UTIL\\MENU\\UTIL"
                                REDRAWPOPBOX(LEVEL)
                                CASE SEL = "Searches" :
MENU43()
                                CASE SEL = "Return" :
BACKONE()
                                QUITLOOP
                                CASE SEL = "Quit" :
QUITPROC()
                                REDRAWPOPBOX(LEVEL)
                                ENDSWITCH
                                ENDIF

                                ENDWHILE
ENDPROC;
WRITELIB LIBNAME MENU4;
RELEASE PROCS MENU4;

PROC MENU5()
    PREVX[LEVEL] = BEGINX
    PREVY[LEVEL] = BEGINY
    PREVNC[LEVEL] = NUMBERCHOICES
    PREVPW[LEVEL] = POPWIDTH
    PREVCL1[LEVEL] = COLOR1
    PREVCL2[LEVEL] = COLOR2
    PREVML[LEVEL] = MENU4NAME
    PREVN[LEVEL] = N
    N=1
        LEVEL = LEVEL + 1
        BEGINX = PREVX[LEVEL-1]+2 ;
    IF PREVY[LEVEL-1] > 8 THEN
        BEGINY = PREVY[LEVEL-1] ELSE
        BEGINY = PREVY[LEVEL-1]+2 ;
    ENDIF
    NUMBERCHOICES =3
    IF BEGINX + NUMBERCHOICES + 4 > 24 THEN
        BEGINX = 3
    ENDIF
        POPWIDTH = 69
        COLOR1 =23
        COLOR2 =30
        MENU4NAME = "DETAILED DATABASE TABLES"
        PREVX[LEVEL] = BEGINX
        PREVY[LEVEL] = BEGINY
        PREVNC[LEVEL] = NUMBERCHOICES
        PREVPW[LEVEL] = POPWIDTH
        PREVCL1[LEVEL] = COLOR1;
        PREVCL2[LEVEL] = COLOR2;
        PREVN[LEVEL] = N
        PREVML[LEVEL] = MENU4NAME;

```

```

    WHILE (TRUE)
    ARRAY C[3]
    ARRAY D[3]
    ARRAY NICK[3]
    C[1] = "Holdings"
    D[1] = "Read description of NRA Detailed Database Tables."
    C[2] = "Return"
    D[2] = "Return to previous menu."
    C[3] = "Quit"
    D[3] = "Exit, return to DOS."
    IF ISASSIGNED(N) = FALSE THEN
        N = 1
    ELSE
        N = N
    ENDIF
    LAST = ARRAYSIZE(C) ; LETS THE PROCEDURE KNOW THE 'BOTTOM' MENU CHOICE.
    IF ISASSIGNED(SEANSBACKFLAG) THEN
        IF SEANSBACKFLAG <> 0 THEN
            SEANSBACKFLAG = SEANSBACKFLAG-1
            BACKONE()
            QUITLOOP
        ENDIF
    ENDIF
    FILLMENU()
    SELMENU()
    IF SEL = "F1MENUHELP" THEN
        SWITCH
        CASE N=1:
            HELPMESSAGE = "Holdings will pop up a window showing information about the NRA Detail
            database."
        CASE N=2:
            HELPMESSAGE = "Return will erase the current menu, returning the viewer to the next higher
            menu."
        CASE N=3:
            HELPMESSAGE = "Quit is used to exit the demonstration"
        ENDSWITCH
        IF HELPMESSAGE = "" THEN
            HELPMESSAGE = "No help available for this menu choice."
        ENDIF
        STYLE ATTRIBUTE SYSCOLOR(22)
        CANVAS OFF
        @ 5,5 ?? "r"+FILL("-",68)+"r"
        @ 5,36 ?? " HELP "
        @ 6,5 ?? "r"+FILL(" ",68)+"r"
        @ 7,5 ?? "r"+FILL(" ",68)+"r"
        @ 8,5 ?? "r"+FILL(" ",68)+"r"
        @ 9,5 ?? "r"+FILL(" ",68)+"r"
        @ 10,5 ?? "r"+FILL(" ",68)+"r"
        @ 11,5 ?? "r"+FILL("-",68)+"r"
        STYLE ATTRIBUTE SYSCOLOR(24)
        @ 6,6 ?? FORMAT("W68,AC","Menu Choice: "+c[n])
        STYLE ATTRIBUTE SYSCOLOR(21)
        SWITCH
        CASE LEN(HELPMESSAGE)<69:
            @ 8,6 ?? FORMAT("W68,AC",HELPMESSAGE)
        OTHERWISE:
            Q=0
        ARRAY PRINTLINE[3]
        WHILE LEN(HELPMESSAGE) > 67
            FOR X FROM 1 TO 67
                IF SUBSTR(HELPMESSAGE,67-X,1) = " " THEN
                    Q = Q + 1

```

```

        PRINTLINE[Q] = SUBSTR(HELPMESSAGE,1,67-X)
        HELPMESSAGE = SUBSTR(HELPMESSAGE,67-X+1,LEN(HELPMESSAGE))
        QUITLOOP
    ENDIF
ENDFOR
ENDWHILE
SWITCH
CASE Q = 3:
    @ 7,7 ?? PRINTLINE[1]
    @ 8,7 ?? PRINTLINE[2]
    @ 9,7 ?? PRINTLINE[3]
    @ 10,7 ?? HELPMESSAGE
CASE Q = 2:
    @ 8,7 ?? PRINTLINE[1]
    @ 9,7 ?? PRINTLINE[2]
    @ 10,7 ?? HELPMESSAGE
CASE Q = 1:
    @ 8,7 ?? PRINTLINE[1]
    @ 9,7 ?? HELPMESSAGE
ENDSWITCH
RELEASE VARS Q,PRINTLINE,HELPMESSAGE
ENDSWITCH
STYLE ATTRIBUTE COLOR3
@ 24,0 ?? FORMAT("W80,AC","Press any key to exit help.")
CANVAS ON
GC = GETCHAR()
REDRAWPOPBOX(LEVEL)
ENDIF
IF SEL = "Esc" THEN
    LEVEL = LEVEL - 1
    BEGINX = PREVX[LEVEL]
    BEGINY = PREVY[LEVEL]
    NUMBERCHOICES = PREVNC[LEVEL]
    POPWIDTH = PREVPW[LEVEL]
    COLOR1 = PREVCL1[LEVEL]
    COLOR2 = PREVCL2[LEVEL]
    MENUUNAME = PREVM[LEVEL]
    N = PREVN[LEVEL]
    RESET
    QUITLOOP
ELSE
    SWITCH
        CASE SEL = "Esc" :
            LEVEL = LEVEL - 1
            BEGINX = PREVX[LEVEL]
            BEGINY = PREVY[LEVEL]
            NUMBERCHOICES = PREVNC[LEVEL]
            POPWIDTH = PREVPW[LEVEL]
            COLOR1 = PREVCL1[LEVEL]
            COLOR2 = PREVCL2[LEVEL]
            N = PREVN[LEVEL]
            RESET
            QUITLOOP
        CASE SEL = "Holdings" : AUTOLIB = "UTIL\\DEMOUTIL"
        TEXTEDITOR("HELP\\DETAIL.TXT","NRA DETAIL DATABASE HOLDINGS")
        AUTOLIB = "UTIL\\MENUMADE,UTIL\\MENUUTIL"
        REDRAWPOPBOX(LEVEL)
        CASE SEL = "Return" :
    BACKONE()
    QUITLOOP
        CASE SEL = "Quit" :
    QUITPROC()
        REDRAWPOPBOX(LEVEL)

```

```

ENDSWITCH
ENDIF

ENDWHILE
ENDPROC;
WRITELIB LIBNAME MENU5;
RELEASE PROCS MENU5;

PROC MENU6()
    PREVX[LEVEL] = BEGINX
    PREVY[LEVEL] = BEGINY
    PREVNC[LEVEL] = NUMBERCHOICES
    PREVPW[LEVEL] = POPWIDTH
    PREVCL1[LEVEL] = COLOR1
    PREVCL2[LEVEL] = COLOR2
    PREVML[LEVEL] = MENUNAME
    PREVN[LEVEL] = N
    N=1
        LEVEL = LEVEL + 1
        BEGINX = PREVX[LEVEL-1]+2 ;
    IF PREVY[LEVEL-1] > 8 THEN
        BEGINY = PREVY[LEVEL-1] ELSE
        BEGINY = PREVY[LEVEL-1]+2 ;
    ENDIF
    NUMBERCHOICES =5
    IF BEGINX + NUMBERCHOICES + 4 > 24 THEN
        BEGINX = 3
    ENDIF
        POPWIDTH = 69
        COLOR1 =23
        COLOR2 =30
        MENUNAME = "HELP WITH DATABASE ACCESS SOFTWARE"
        PREVX[LEVEL] = BEGINX
        PREVY[LEVEL] = BEGINY
        PREVNC[LEVEL] = NUMBERCHOICES
        PREVPW[LEVEL] = POPWIDTH
        PREVCL1[LEVEL] = COLOR1;
        PREVCL2[LEVEL] = COLOR2;
        PREVN[LEVEL] = N
        PREVML[LEVEL] = MENUNAME;
        WHILE (TRUE)
        ARRAY C[5]
        ARRAY D[5]
        ARRAY NICK[5]
        C[1] = "Purpose"
        D[1] = "Information about scope and purpose."
        C[2] = "Cursor"
        D[2] = "Information about cursor movement."
        C[3] = "Function Key"
        D[3] = "Information about function key usage."
        C[4] = "Return"
        D[4] = "Return to previous menu."
        C[5] = "Quit"
        D[5] = "Exit, return to DOS."
        IF ISASSIGNED(N) = FALSE THEN
            N = 1
        ELSE
            N = N
        ENDIF
        LAST = ARRAYSIZE(C) ; LETS THE PROCEDURE KNOW THE 'BOTTOM' MENU CHOICE.
        IF ISASSIGNED(SEANSBACKFLAG) THEN

```

```

IF SEANSBACKFLAG <> 0 THEN
    SEANSBACKFLAG = SEANSBACKFLAG-1
    BACKONE()
    QUITLOOP
ENDIF
ENDIF
FILLMENU()
SELMENU()
IF SEL = "F1MENUHELP" THEN
    SWITCH
    CASE N=1:
        HELPMESSAGE = "Purpose will pop up a window describing the scope and purpose of this
        demonstration."
    CASE N=2:
        HELPMESSAGE = "Cursor will pop up a window describing the control of cursor movement in
        this demonstration."
    CASE N=3:
        HELPMESSAGE = "Function Key will pop up a window describing the use of function keys in
        this demonstration."
    CASE N=4:
        HELPMESSAGE = "Return will erase the current menu, returning the viewer to the next higher
        menu."
    CASE N=5:
        HELPMESSAGE = "Quit is used to exit the demonstration."
    ENDSWITCH
    IF HELPMESSAGE = "" THEN
        HELPMESSAGE = "No help available for this menu choice."
    ENDIF
    STYLE ATTRIBUTE SYSCOLOR(22)
    CANVAS OFF
    @ 5,5 ?? " " + FILL(" ",68) + " "
    @ 5,36 ?? " HELP "
    @ 6,5 ?? " " + FILL(" ",68) + " "
    @ 7,5 ?? " " + FILL(" ",68) + " "
    @ 8,5 ?? " " + FILL(" ",68) + " "
    @ 9,5 ?? " " + FILL(" ",68) + " "
    @ 10,5 ?? " " + FILL(" ",68) + " "
    @ 11,5 ?? " " + FILL(" ",68) + " "
    STYLE ATTRIBUTE SYSCOLOR(24)
    @ 6,6 ?? FORMAT("W68,AC","Menu Choice: "+c[n])
    STYLE ATTRIBUTE SYSCOLOR(21)
    SWITCH
    CASE LEN(HELPMESSAGE)<69:
        @ 8,6 ?? FORMAT("W68,AC",HELPMESSAGE)
    OTHERWISE:
        Q=0
    ARRAY PRINTLINE[3]
    WHILE LEN(HELPMESSAGE) > 67
        FOR X FROM 1 TO 67
            IF SUBSTR(HELPMESSAGE,67-X,1) = " " THEN
                Q = Q + 1
                PRINTLINE[Q] = SUBSTR(HELPMESSAGE,1,67-X)
                HELPMESSAGE = SUBSTR(HELPMESSAGE,67-X+1,LEN(HELPMESSAGE))
                QUITLOOP
            ENDIF
        ENDFOR
    ENDWHILE
    SWITCH
    CASE Q = 3:
        @ 7,7 ?? PRINTLINE[1]
        @ 8,7 ?? PRINTLINE[2]
        @ 9,7 ?? PRINTLINE[3]

```

```

    @ 10,7 ?? HELPMESSAGE
CASE q = 2:
    @ 8,7 ?? PRINTLINE[1]
    @ 9,7 ?? PRINTLINE[2]
    @ 10,7 ?? HELPMESSAGE
CASE q = 1:
    @ 8,7 ?? PRINTLINE[1]
    @ 9,7 ?? HELPMESSAGE
ENDSWITCH
RELEASE VARS q,PRINTLINE,HELMMESSAGE
ENDSWITCH
STYLE ATTRIBUTE COLOR3
@ 24,0 ?? FORMAT("W80,AC","Press any key to exit help.")
CANVAS ON
GC = GETCHAR()
REDRAWPOPBOX(LEVEL)
ENDIF
    IF SEL = "Esc" THEN
        LEVEL = LEVEL - 1
        BEGINX = PREVX[LEVEL]
        BEGINY = PREVY[LEVEL]
        NUMBERCHOICES = PREVNC[LEVEL]
        POPWIDTH = PREVPW[LEVEL]
        COLOR1 = PREVCL1[LEVEL]
        COLOR2 = PREVCL2[LEVEL]
        MENUNAME = PREVM[LEVEL]
        N = PREVN[LEVEL]

        RESET
        QUITLOOP
    ELSE
        SWITCH
            CASE SEL = "Esc" :
                LEVEL = LEVEL - 1
                BEGINX = PREVX[LEVEL]
                BEGINY = PREVY[LEVEL]
                NUMBERCHOICES = PREVNC[LEVEL]
                POPWIDTH = PREVPW[LEVEL]
                COLOR1 = PREVCL1[LEVEL]
                COLOR2 = PREVCL2[LEVEL]
                N = PREVN[LEVEL]
                RESET
                QUITLOOP

            CASE SEL = "Purpose" : AUTOLIB = "UTIL\\DEMOUTIL"
TEXTEDITOR ("HELP\\PURPOSE.TXT","NRA DEMONSTRATION - PURPOSE")
                AUTOLIB = "UTIL\\MENUMADE,UTIL\\MENUUTIL"
                REDRAWPOPBOX(LEVEL)

            CASE SEL = "Cursor" : AUTOLIB = "UTIL\\DEMOUTIL"
TEXTEDITOR ("HELP\\CURSOR.TXT","NRA DEMONSTRATION - CURSOR CONTROL")
                AUTOLIB = "UTIL\\MENUMADE,UTIL\\MENUUTIL"
                REDRAWPOPBOX(LEVEL)

            CASE SEL = "Function Key" : AUTOLIB = "UTIL\\DEMOUTIL"
TEXTEDITOR ("HELP\\FUNCTION.TXT","NRA DEMONSTRATION - FUNCTION KEY DEFINITIONS")
                AUTOLIB = "UTIL\\MENUMADE,UTIL\\MENUUTIL"
                REDRAWPOPBOX(LEVEL)

            CASE SEL = "Return" :
BACKONE()

                QUITLOOP

            CASE SEL = "Quit" :
QUITPROC()

                REDRAWPOPBOX(LEVEL)
        ENDSWITCH
    ENDIF

```

```

ENDWHILE
ENDPROC;
WRITELIB LIBNAME MENU6;
RELEASE PROCS MENU6;

PROC MENU23()
    PREVX[LEVEL] = BEGINX
    PREVY[LEVEL] = BEGINY
    PREVNC[LEVEL] = NUMBERCHOICES
    PREVPW[LEVEL] = POPWIDTH
    PREVCL1[LEVEL] = COLOR1
    PREVCL2[LEVEL] = COLOR2
    PREVML[LEVEL] = MENUNAME
    PREVN[LEVEL] = N
    N=1
        LEVEL = LEVEL + 1
        BEGINX = PREVX[LEVEL-1]+2 ;
    IF PREVY[LEVEL-1] > 8 THEN
        BEGINY = PREVY[LEVEL-1] ELSE
        BEGINY = PREVY[LEVEL-1]+2 ;
    ENDIF
    NUMBERCHOICES =8
    IF BEGINX + NUMBERCHOICES + 4 > 24 THEN
        BEGINX = 3
    ENDIF
        POPWIDTH = 69
        COLOR1 =23
        COLOR2 =30
        MENUNAME = "SEARCH THE SUMMARY DATABASE"
        PREVX[LEVEL] = BEGINX
        PREVY[LEVEL] = BEGINY
        PREVNC[LEVEL] = NUMBERCHOICES
        PREVPW[LEVEL] = POPWIDTH
        PREVCL1[LEVEL] = COLOR1;
        PREVCL2[LEVEL] = COLOR2;
        PREVN[LEVEL] = N
        PREVML[LEVEL] = MENUNAME;
        WHILE (TRUE)
            ARRAY C[8]
            ARRAY D[8]
            ARRAY NICK[8]
            C[1] = "Animal"
            D[1] = "Retrieve Animals by animal number(s).\"
            C[2] = "Assigned ID\"
            D[2] = "Retrieve Animals by assigned ID(s).\"
            C[3] = "T-effect\"
            D[3] = "Retrieve Animals by tissue effect category(ies).\"
            C[4] = "T-dose\"
            D[4] = "Retrieve Animals by tissue dose(s).\"
            C[5] = "Study\"
            D[5] = "Retrieve Animals by study ID code(s).\"
            C[6] = "Return\"
            D[6] = "Return to previous menu.\"
            C[7] = "Main\"
            D[7] = "Return to Main Menu.\"
            C[8] = "Quit\"
            D[8] = "Exit, return to DOS.\"
            IF ISASSIGNED(N) = FALSE THEN
                N = 1
            ELSE
                N = N

```

```

ENDIF
LAST = ARRAYSIZE(C) ; LETS THE PROCEDURE KNOW THE 'BOTTOM' MENU CHOICE.
IF ISASSIGNED(SEANSBACKFLAG) THEN
  IF SEANSBACKFLAG <> 0 THEN
    SEANSBACKFLAG = SEANSBACKFLAG-1
    BACKONE()
    QUITLOOP
  ENDIF
ENDIF
FILLMENU()
SELMENU()
IF SEL = "F1MENUHELP" THEN
  SWITCH
  CASE N=1:
    HELPMESSAGE = "Animal will bring up a prompt window for entry of an ANIMAL_ID. Or, hit F1
for a list of possible animal numbers and select "+
"one or more using <RETURN>. After specification of animal(s), hit F2 to select the
subset."
  CASE N=2:
    HELPMESSAGE = "Assigned ID will bring up a prompt window for entry of an ASSIGNED_ID. Or,
hit F1 to list possible assigned IDs and choose o"+
"ne or more using <RETURN>. After specification of assigned ID(s), hit F2 to select the
subset."
  CASE N=3:
    HELPMESSAGE = "Teffect will display a list of possible tissue and morphology categories.
One or more may be selected. After specification"+
" of retrieval criteria, hit F2 to select the subset."
  CASE N=4:
    HELPMESSAGE = "Tdose will display a list of possible tissue categories and doses. One or
more may be selected. After specification of ret"+
"rieval criteria, hit F2 to select the subset."
  CASE N=5:
    HELPMESSAGE = "Study will bring up a prompt window for entry of a study title (use ...
after the first few characters). Or, hit F1 for a li"+
"st of possible titles. After specification of retrieval criteria, hit F2 to select the
subset."
  CASE N=6:
    HELPMESSAGE = "Return will erase the current menu, returning the viewer to the next higher
menu."
  CASE N=7:
    HELPMESSAGE = "Main will erase all submenus, returning to the main menu."
  CASE N=8:
    HELPMESSAGE = "Quit is used to exit the demonstration."
  ENDSWITCH
  IF HELPMESSAGE = "" THEN
    HELPMESSAGE = "No help available for this menu choice."
  ENDIF
  STYLE ATTRIBUTE SYSCOLOR(22)
  CANVAS OFF
  @ 5,5 ?? " "+FILL("-",68)+"_"
  @ 5,36 ?? " HELP "
  @ 6,5 ?? " "+FILL(" ",68)+"|"
  @ 7,5 ?? " "+FILL(" ",68)+" "
  @ 8,5 ?? " "+FILL(" ",68)+" "
  @ 9,5 ?? " "+FILL(" ",68)+"|"
  @ 10,5 ?? " "+FILL(" ",68)+"|"
  @ 11,5 ?? " "+FILL("-",68)+"_"
  STYLE ATTRIBUTE SYSCOLOR(24)
  @ 6,6 ?? FORMAT("W68,AC","Menu Choice: "+c[n])
  STYLE ATTRIBUTE SYSCOLOR(21)
  SWITCH
  CASE LEN(HELPMESSAGE)<69:

```

```

@ 8,6 ?? FORMAT("W68,AC",HELPMESSAGE)
OTHERWISE:
Q=0
ARRAY PRINTLINE[3]
WHILE LEN(HELPMESSAGE) > 67
  FOR X FROM 1 TO 67
    IF SUBSTR(HELPMESSAGE,67-X,1) = " " THEN
      Q = Q + 1
      PRINTLINE[Q] = SUBSTR(HELPMESSAGE,1,67-X)
      HELPMESSAGE = SUBSTR(HELPMESSAGE,67-X+1,LEN(HELPMESSAGE))
      QUITLOOP
    ENDIF
  ENDFOR
ENDWHILE
SWITCH
CASE Q = 3:
  @ 7,7 ?? PRINTLINE[1]
  @ 8,7 ?? PRINTLINE[2]
  @ 9,7 ?? PRINTLINE[3]
  @ 10,7 ?? HELPMESSAGE
CASE Q = 2:
  @ 8,7 ?? PRINTLINE[1]
  @ 9,7 ?? PRINTLINE[2]
  @ 10,7 ?? HELPMESSAGE
CASE Q = 1:
  @ 8,7 ?? PRINTLINE[1]
  @ 9,7 ?? HELPMESSAGE
ENDSWITCH
RELEASE VARS Q,PRINTLINE,HELPMESSAGE
ENDSWITCH
STYLE ATTRIBUTE COLOR3
@ 24,0 ?? FORMAT("W80,AC","Press any key to exit help.")
CANVAS ON
GC = GETCHAR()
REDRAWPOPBOX(LEVEL)
ENDIF
IF SEL = "Esc" THEN
  LEVEL = LEVEL - 1
  BEGINX = PREVX[LEVEL]
  BEGINY = PREVY[LEVEL]
  NUMBERCHOICES = PREVNC[LEVEL]
  POPWIDTH = PREVPW[LEVEL]
  COLOR1 = PREVCL1[LEVEL]
  COLOR2 = PREVCL2[LEVEL]
  MENUName = PREVM1[LEVEL]
  N = PREVN[LEVEL]
  RESET
  QUITLOOP
ELSE
  SWITCH
    CASE SEL = "Esc" :
      LEVEL = LEVEL - 1
      BEGINX = PREVX[LEVEL]
      BEGINY = PREVY[LEVEL]
      NUMBERCHOICES = PREVNC[LEVEL]
      POPWIDTH = PREVPW[LEVEL]
      COLOR1 = PREVCL1[LEVEL]
      COLOR2 = PREVCL2[LEVEL]
      N = PREVN[LEVEL]
      RESET
      QUITLOOP
    CASE SEL = "Animal" : AUTOLIB = "UTIL\\DEMOUTIL"
  SUMANIMALSEARCH()

```

```

        AUTOLIB ="UTIL\MENUMADE,UTIL\MENUUTIL"
        REDRAWPOPBOX(LEVEL)
        CASE SEL = "Assigned ID" : AUTOLIB = "UTIL\DEMOUTIL"
SUMASSIGNEDIDSEARCH()
        AUTOLIB ="UTIL\MENUMADE,UTIL\MENUUTIL"
        REDRAWPOPBOX(LEVEL)
        CASE SEL = "T-effect" : AUTOLIB = "UTIL\DEMOUTIL"
TEFFECTSEARCH()
        AUTOLIB ="UTIL\MENUMADE,UTIL\MENUUTIL"
        REDRAWPOPBOX(LEVEL)
        CASE SEL = "T-dose" : AUTOLIB = "UTIL\DEMOUTIL"
TDOSESEARCH()
        AUTOLIB ="UTIL\MENUMADE,UTIL\MENUUTIL"
        REDRAWPOPBOX(LEVEL)
        CASE SEL = "Study" : AUTOLIB = "UTIL\DEMOUTIL"
SUMSTUDYSEARCH()
        AUTOLIB ="UTIL\MENUMADE,UTIL\MENUUTIL"
        REDRAWPOPBOX(LEVEL)
        CASE SEL = "Return" :
BACKONE()
        CASE SEL = "Main" :           QUITLOOP
FIRSTONE()
        CASE SEL = "Quit" :          QUITLOOP
QUITPROC()
        REDRAWPOPBOX(LEVEL)
    ENDSWITCH
    ENDIF

    ENDWHILE
ENDPROC;
WRITELIB LIBNAME MENU23;
RELEASE PROCS MENU23;

PROC MENU33()
    PREVX[LEVEL] = BEGINX
    PREVY[LEVEL] = BEGINY
    PREVNC[LEVEL] = NUMBERCHOICES
    PREVPW[LEVEL] = POPWIDTH
    PREVCL1[LEVEL] = COLOR1
    PREVCL2[LEVEL] = COLOR2
    PREVML[LEVEL] = MENUNAME
    PREVN[LEVEL] = N
    N=1
        LEVEL = LEVEL + 1
        BEGINX = PREVX[LEVEL-1]+2 ;
    IF PREVY[LEVEL-1] > 8 THEN
        BEGINY = PREVY[LEVEL-1] ELSE
        BEGINY = PREVY[LEVEL-1]+2 ;
    ENDIF
    NUMBERCHOICES =6
    IF BEGINX + NUMBERCHOICES + 4 > 24 THEN
        BEGINX = 3
    ENDIF
        POPWIDTH = 69
        COLOR1 =23
        COLOR2 =30
        MENUNAME = "SEARCH THE INVENTORY DATABASE"
        PREVX[LEVEL] = BEGINX
        PREVY[LEVEL] = BEGINY

```

```

PREVNC[LEVEL] = NUMBERCHOICES
PREVPW[LEVEL] = POPWIDTH
PREVCL1[LEVEL] = COLOR1;
PREVCL2[LEVEL] = COLOR2;
PREVN[LEVEL] = N
PREVM[LEVEL] = MENUNAME;
WHILE (TRUE)
ARRAY C[6]
ARRAY D[6]
ARRAY NICK[6]
C[1] = "Animal"
D[1] = "Retrieve Inventory records by animal number(s).\"
C[2] = "Assigned ID\"
D[2] = "Retrieve Inventory records by assigned ID(s).\"
C[3] = "Study\"
D[3] = "Retrieve Inventory records by study ID codes(s).\"
C[4] = "Return\"
D[4] = "Return to previous menu.\"
C[5] = "Main\"
D[5] = "Return to Main Menu.\"
C[6] = "Quit\"
D[6] = "Exit, return to DOS.\"
IF ISASSIGNED(N) = FALSE THEN
    N = 1
ELSE
    N = N
ENDIF
LAST = ARRAYSIZE(C) ; LETS THE PROCEDURE KNOW THE 'BOTTOM' MENU CHOICE.
IF ISASSIGNED(SEANSBACKFLAG) THEN
    IF SEANSBACKFLAG <> 0 THEN
        SEANSBACKFLAG = SEANSBACKFLAG-1
        BACKONE()
        QUITLOOP
    ENDIF
ENDIF
FILLMENU()
SELMENU()
IF SEL = "F1MENUHELP" THEN
SWITCH
CASE N=1:
HELPMESSAGE = "Animal will bring up a prompt window for entry of an animal number. Or, hit
F1 for a list of possible animals. One or more "+
"may be selected using <RETURN>. After specification of search criteria, hit F2 to select
subset.\"
CASE N=2:
HELPMESSAGE = "Assigned ID will bring up a prompt window for entry of an assigned ID. Or,
hit F1 for a list of possible assigned ID's. One"+
" or more may be selected using <RETURN>. After specification of search criteria, hit F2
to select subset.\"
CASE N=3:
HELPMESSAGE = "Study will bring up a prompt window for entry of study title (followed by
...). Or, hit F1 for a list and select one or more"+
" using <RETURN>. After specification of search criteria, hit F2 to select subset.\"
CASE N=4:
HELPMESSAGE = "Return will erase the current menu, returning the viewer to the next higher
menu.\"
CASE N=5:
HELPMESSAGE = "Main will erase all submenus, returning to the main menu.\"
CASE N=6:
HELPMESSAGE = "Quit is used to exit the demonstration.\"
ENDSWITCH
IF HELPMESSAGE = "" THEN

```

```

    HELPMESSAGE = "No help available for this menu choice."
ENDIF
STYLE ATTRIBUTE SYSCOLOR(22)
CANVAS OFF
@ 5,5 ?? " " + FILL("-",68) + " "
@ 5,36 ?? " HELP "
@ 6,5 ?? " " + FILL(" ",63) + " "
@ 7,5 ?? " " + FILL(" ",68) + " "
@ 8,5 ?? " " + FILL(" ",68) + " "
@ 9,5 ?? " " + FILL(" ",68) + " "
@ 10,5 ?? " " + FILL(" ",68) + " "
@ 11,5 ?? " " + FILL("-",68) + " "
STYLE ATTRIBUTE SYSCOLOR(24)
@ 6,6 ?? FORMAT("W68,AC","Menu Choice: "+c[n])
STYLE ATTRIBUTE SYSCOLOR(21)
SWITCH
CASE LEN(HELMMESSAGE)<69:
    @ 8,6 ?? FORMAT("W68,AC",HELMMESSAGE)
OTHERWISE:
Q=0
ARRAY PRINTLINE[3]
WHILE LEN(HELMMESSAGE) > 67
    FOR X FROM 1 TO 67
        IF SUBSTR(HELMMESSAGE,67-X,1) = " " THEN
            Q = Q + 1
            PRINTLINE[Q] = SUBSTR(HELMMESSAGE,1,67-X)
            HELMMESSAGE = SUBSTR(HELMMESSAGE,67-X+1,LEN(HELMMESSAGE))
            QUITLOOP
        ENDIF
    ENDFOR
ENDWHILE
SWITCH
CASE Q = 3:
    @ 7,7 ?? PRINTLINE[1]
    @ 8,7 ?? PRINTLINE[2]
    @ 9,7 ?? PRINTLINE[3]
    @ 10,7 ?? HELMMESSAGE
CASE Q = 2:
    @ 8,7 ?? PRINTLINE[1]
    @ 9,7 ?? PRINTLINE[2]
    @ 10,7 ?? HELMMESSAGE
CASE Q = 1:
    @ 8,7 ?? PRINTLINE[1]
    @ 9,7 ?? HELMMESSAGE
ENDSWITCH
RELEASE VARS Q,PRINTLINE,HELMMESSAGE
ENDSWITCH
STYLE ATTRIBUTE COLOR3
@ 24,0 ?? FORMAT("W80,AC","Press any key to exit help.")
CANVAS ON
GC = GETCHAR()
REDRAWPOPOBOX(LEVEL)
ENDIF
IF SEL = "Esc" THEN
    LEVEL = LEVEL - 1
    BEGINX = PREVX[LEVEL]
    BEGINY = PREVY[LEVEL]
    NUMBERCHOICES = PREVNC[LEVEL]
    POPWIDTH = PREVPW[LEVEL]
    COLOR1 = PREVCL1[LEVEL]
    COLOR2 = PREVCL2[LEVEL]
    MENUNAME = PREVML[LEVEL]

```

```

                                N = PREVN[LEVEL]
                                RESET
                                QUITLOOP
ELSE
SWITCH
    CASE SEL = "Esc" :          LEVEL = LEVEL - 1
                                BEGINX = PREVX[LEVEL]
                                BEGINY = PREVY[LEVEL]
                                NUMBERCHOICES = PREVNC[LEVEL]
                                POPWIDTH = PREVPW[LEVEL]
                                COLOR1 = PREVCL1[LEVEL]
                                COLOR2 = PREVCL2[LEVEL]
                                N = PREVN[LEVEL]
                                RESET
                                QUITLOOP
    CASE SEL = "Animal" : AUTOLIB = "UTIL\\DEMOUTIL"
INVANIMALSEARCH()
                                AUTOLIB ="UTIL\\MENUMADE,UTIL\\MENUUTIL"
                                REDRAWPOPBOX(LEVEL)
    CASE SEL = "Assigned ID" : AUTOLIB = "UTIL\\DEMOUTIL"
INVASSIGNEDIDSEARCH()
                                AUTOLIB ="UTIL\\MENUMADE,UTIL\\MENUUTIL"
                                REDRAWPOPBOX(LEVEL)
    CASE SEL = "Study" : AUTOLIB = "UTIL\\DEMOUTIL"
INVSTUDYSEARCH()
                                AUTOLIB ="UTIL\\MENUMADE,UTIL\\MENUUTIL"
                                REDRAWPOPBOX(LEVEL)
    CASE SEL = "Return" :
BACKONE()
                                QUITLOOP
    CASE SEL = "Main" :
FIRSTONE()
                                QUITLOOP
    CASE SEL = "Quit" :
QUITPROC()
                                REDRAWPOPBOX(LEVEL)
ENDSWITCH
ENDIF

ENDWHILE
ENDPROC;
WRITELIB LIBNAME MENU33;
RELEASE PROCS MENU33;

PROC MENU43()
    PREVX[LEVEL] = BEGINX
    PREVY[LEVEL] = BEGINY
    PREVNC[LEVEL] = NUMBERCHOICES
    PREVPW[LEVEL] = POPWIDTH
    PREVCL1[LEVEL] = COLOR1
    PREVCL2[LEVEL] = COLOR2
    PREVML[LEVEL] = MENUNAME
    PREVN[LEVEL] = N
    N=1
    LEVEL = LEVEL + 1
    BEGINX = PREVX[LEVEL-1]+2 ;
    IF PREVY[LEVEL-1] > 8 THEN
        BEGINY = PREVY[LEVEL-1] ELSE
        BEGINY = PREVY[LEVEL-1]+2 ;
    ENDIF
    NUMBERCHOICES =6

```

```

IF BEGINX + NUMBERCHOICES + 4 > 24 THEN
  BEGINX = 3
ENDIF
  POPWIDTH = 69
  COLOR1 = 23
  COLOR2 = 30
  MENUNAME = "SEARCH THE BIBLIOGRAPHY DATABASE"
  PREVX[LEVEL] = BEGINX
  PREVY[LEVEL] = BEGINX
  PREVNC[LEVEL] = NUMBERCHOICES
  PREVPW[LEVEL] = POPWIDTH
  PREVCL1[LEVEL] = COLOR1;
  PREVCL2[LEVEL] = COLOR2;
  PREVN[LEVEL] = N
  PREVML[LEVEL] = MENUNAME;
  WHILE (TRUE)
    ARRAY C[6]
    ARRAY D[6]
    ARRAY NICK[6]
    C[1] = "Author"
    D[1] = "Retrieve subset by author(s)."
    C[2] = "Title"
    D[2] = "Retrieve subset by title(s)."
    C[3] = "Year"
    D[3] = "Retrieve subset by publication year(s)."
    C[4] = "Return"
    D[4] = "Return to previous menu."
    C[5] = "Main"
    D[5] = "Return to Main Menu."
    C[6] = "Quit"
    D[6] = "Exit, return to DOS."
    IF ISASSIGNED(N) = FALSE THEN
      N = 1
    ELSE
      N = N
    ENDIF
    LAST = ARRAYSIZE(C) ; LETS THE PROCEDURE KNOW THE 'BOTTOM' MENU CHOICE.
    IF ISASSIGNED(SEANSBACKFLAG) THEN
      IF SEANSBACKFLAG <> 0 THEN
        SEANSBACKFLAG = SEANSBACKFLAG-1
        BACKONE()
        QUITLOOP
      ENDIF
    ENDIF
    FILLMENU()
    SELMENU()
    IF SEL = "F1MENUHELP" THEN
      SWITCH
      CASE N=1:
        HELPMESSAGE = "Author will bring up a prompt window for entry of an author's last name.
        Or, hit F1 for a list choose one or more using <RET">
        "URN>. Hit F2 to select the subset."
      CASE N=2:
        HELPMESSAGE = "Title will bring up a prompt window for entry of the first portion of a
        title (followed by ..). Or, hit F1 for a list and se">
        "lect one or more using <RETURN>. After specification of search criteria, hit F2 to select
        the subset."
      CASE N=3:
        HELPMESSAGE = "Year will bring up a prompt window for entry of the 4 digit publication
        year. Or, hit F1 for a list and select one or more u">
        "sing <RETURN>. After specification of search criteria, hit F2 to select the subset."
      CASE N=4:

```

```

HELPMESSAGE = "Return will erase the current menu, returning the viewer to the next higher
menu."
CASE N=5:
HELPMESSAGE = "Main will erase all submenus, returning to the main menu."
CASE N=6:
HELPMESSAGE = "Quit is used to exit the demonstration."
ENDSWITCH
IF HELPMESSAGE = "" THEN
    HELPMESSAGE = "No help available for this menu choice."
ENDIF
STYLE ATTRIBUTE SYSCOLOR(22)
CANVAS OFF
@ 5,5 ?? " " + FILL(" ",68) + " "
@ 5,36 ?? " HELP "
@ 6,5 ?? " " + FILL(" ",68) + " "
@ 7,5 ?? " " + FILL(" ",68) + " "
@ 8,5 ?? " " + FILL(" ",68) + " "
@ 9,5 ?? " " + FILL(" ",68) + " "
@ 10,5 ?? " " + FILL(" ",68) + " "
@ 11,5 ?? " " + FILL(" ",68) + " "
STYLE ATTRIBUTE SYSCOLOR(24)
@ 6,6 ?? FORMAT("W68,AC","Menu Choice: "+c[n])
STYLE ATTRIBUTE SYSCOLOR(21)
SWITCH
CASE LEN(HELPMESSAGE) < 69:
    @ 8,6 ?? FORMAT("W68,AC",HELPMESSAGE)
OTHERWISE:
    Q=0
ARRAY PRINTLINE[3]
    WHILE LEN(HELPMESSAGE) > 67
        FOR X FROM 1 TO 67
            IF SUBSTR(HELPMESSAGE,67-X,1) = " " THEN
                Q = Q + 1
                PRINTLINE[Q] = SUBSTR(HELPMESSAGE,1,67-X)
                HELPMESSAGE = SUBSTR(HELPMESSAGE,67-X+1,LEN(HELPMESSAGE))
                QUITLOOP
            ENDIF
        ENDFOR
    ENDWHILE
SWITCH
CASE Q = 3:
    @ 7,7 ?? PRINTLINE[1]
    @ 8,7 ?? PRINTLINE[2]
    @ 9,7 ?? PRINTLINE[3]
    @ 10,7 ?? HELPMESSAGE
CASE Q = 2:
    @ 8,7 ?? PRINTLINE[1]
    @ 9,7 ?? PRINTLINE[2]
    @ 10,7 ?? HELPMESSAGE
CASE Q = 1:
    @ 8,7 ?? PRINTLINE[1]
    @ 9,7 ?? HELPMESSAGE
ENDSWITCH
RELEASE VARS Q,PRINTLINE,HELPMESSAGE
ENDSWITCH
STYLE ATTRIBUTE COLOR3
@ 24,0 ?? FORMAT("W80,AC","Press any key to exit help.")
CANVAS ON
GC = GETCHAR()
REDRAWPOPBOX(LEVEL)
ENDIF
IF SEL = "Esc" THEN

```

```

        LEVEL = LEVEL - 1
        BEGINX = PREVX[LEVEL]
        BEGINY = PREVY[LEVEL]
        NUMBERCHOICES = PREVNC[LEVEL]
        POPWIDTH = PREVPW[LEVEL]
        COLOR1 = PREVCL1[LEVEL]
        COLOR2 = PREVCL2[LEVEL]
        MENUNAME = PREVM[LEVEL]
        N = PREVN[LEVEL]

    RESET
    QUITLOOP
ELSE
SWITCH
    CASE SEL = "Esc" :          LEVEL = LEVEL - 1
                                BEGINX = PREVX[LEVEL]
                                BEGINY = PREVY[LEVEL]
                                NUMBERCHOICES = PREVNC[LEVEL]
                                POPWIDTH = PREVPW[LEVEL]
                                COLOR1 = PREVCL1[LEVEL]
                                COLOR2 = PREVCL2[LEVEL]
                                N = PREVN[LEVEL]
                                RESET
                                QUITLOOP

    CASE SEL = "Author" : AUTOLIB = "UTIL\DEMOUTIL"
AUTHORSEARCH()
                        AUTOLIB = "UTIL\MENUMADE,UTIL\MENUUTIL"
                        REDRAWPOPBOX(LEVEL)
    CASE SEL = "Title" : AUTOLIB = "UTIL\DEMOUTIL"
TITLESEARCH()
                        AUTOLIB = "UTIL\MENUMADE,UTIL\MENUUTIL"
                        REDRAWPOPBOX(LEVEL)
    CASE SEL = "Year" : AUTOLIB = "UTIL\DEMOUTIL"
YEARSEARCH()
                        AUTOLIB = "UTIL\MENUMADE,UTIL\MENUUTIL"
                        REDRAWPOPBOX(LEVEL)
    CASE SEL = "Return" :
BACKONE()
                                QUITLOOP

    CASE SEL = "Main" :
FIRSTONE()
                                QUITLOOP

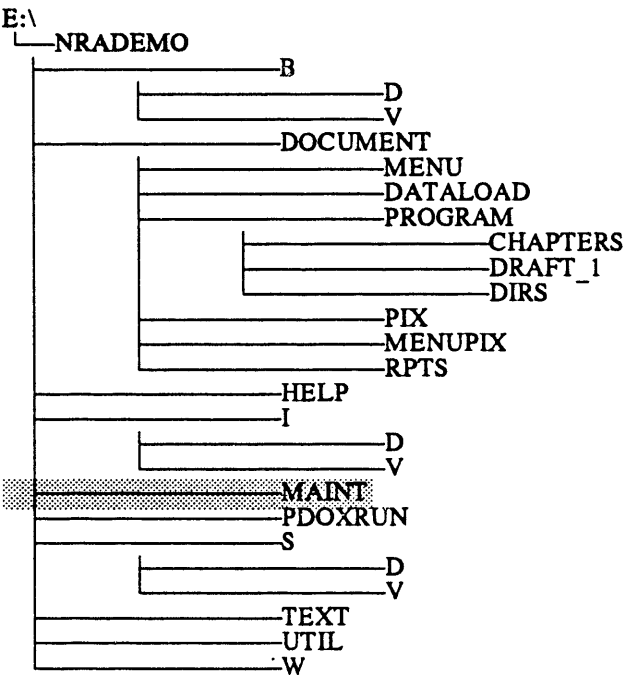
    CASE SEL = "Quit" :
QUITPROC()
                        REDRAWPOPBOX(LEVEL)
ENDSWITCH
ENDIF

ENDWHILE
ENDPROC;
WRITELIB LIBNAME MENU43;
RELEASE PROCS MENU43;

```

CHAPTER 4 - \MAINT

The Maintenance subdirectory contains scripts associated with populating the demonstration and preparing it on floppy diskettes for distribution. See the Dataload manual for usage instructions. Basically, you run the script MAINT\MAINTAIN and choose the option of interest. The other scripts are only used if the entire maintenance process is being revised.



Directory of E:\NRADEMO\MAINT

Volume in drive E is Jon
Directory of E:\NRADEMO\MAINT

```

.      <DIR>      8-25-91   3:49p
..     <DIR>      8-25-91   3:49p
COMPRESS BAT      889 10-21-91 10:53a
INSTALL BAT      2274 10-21-91 10:53a
MAINTAIN SC       1128 10-21-91 10:54a
MAINTUTL LIB     79991 10-18-91 12:17p
MAINTUTL SC      38458 10-18-91 12:17p
MAKEDISK BAT      626 10-21-91 10:57a
MAKEMENU SC      29310 8-24-91 12:02p
MENUCHOI DB      16384 8-25-91 1:29p
MENUCHOI F1       738 8-24-91 1:22p
MENUCHOI F2       946 8-24-91 1:22p
MENUCHOI F3      1488 8-24-91 1:22p
MENUCHOI PX      4096 8-25-91 1:29p
MENUCHOI R       4063 8-24-91 1:22p
MENUCHOI SET      200 8-24-91 1:22p
MENULEVE DB      4096 8-25-91 1:29p
MENULEVE F        968 8-22-91 8:17p
MENULEVE F1      1901 8-24-91 12:03p
MENULEVE F2      1253 8-24-91 12:04p
MENULEVE PX      4096 8-25-91 1:29p
MENULEVE R      3055 8-22-91 8:17p
MENULEVE SET      146 8-22-91 8:17p
MENULEVE VAL      115 8-22-91 8:17p
MENU MADE LIB    62777 8-25-91 1:29p
MENU MADE SC     20412 8-25-91 1:28p
MENUUTIL LIB    38712 9-04-91 3:10p
MENUUTIL SC     16897 8-24-91 12:01p
NRA BAT         118 10-21-91 10:59a
NRADAT1 ZOO      277 7-30-91 2:20p
NRADAT10 ZOO     17976 9-13-91 9:43a
NRADAT11 ZOO     40026 10-16-91 1:57p
NRADAT12 ZOO    944376 10-11-91 10:12a
NRADAT2 ZOO      9790 10-09-91 4:35p
NRADAT3 ZOO     91347 10-18-91 1:02p
NRADAT4 ZOO      3519 4-24-91 4:20p
NRADAT5 ZOO     30076 10-18-91 12:57p
NRADAT6 ZOO      4757 7-12-91 2:16p
NRADAT7 ZOO     30410 10-18-91 12:57p
NRADAT8 ZOO     53929 10-18-91 1:06p
NRADAT9 ZOO    344685 10-17-91 12:05p
ZOO EXE        36856 1-31-87 3:51p
42 File(s) 18667520 bytes free

```

DOS BATCH PROCESSORS

COMPRESS.BAT

```
rem Compress NRA distribution disk - 24-Aug-91 JCP
rem revised 9/19/91 crw - text files now in HELP and TEXT
echo off
cls
echo COMPRESSING ENTIRE NRA APPLICATION AND DATA...
echo                                     (Except \maint subdirectory)
echo '
echo '
echo '
del maint\*.zoo
maint\zoo -add maint\nradat1.zoo util\NRA.bat
maint\zoo -add maint\nradat2.zoo s\v\*. *
maint\zoo -add maint\nradat3.zoo s\d\*. *
maint\zoo -add maint\nradat4.zoo b\v\*. *
maint\zoo -add maint\nradat5.zoo b\d\*. *
maint\zoo -add maint\nradat6.zoo i\v\*. *
maint\zoo -add maint\nradat7.zoo i\d\*. *
maint\zoo -add maint\nradat8.zoo w\*. *
maint\zoo -add maint\nradat9.zoo util\*.LIB
maint\zoo -add maint\nradat9.zoo util\MAIN.SC
maint\zoo -add maint\nradat10.zoo text\*. *
maint\zoo -add maint\nradat11.zoo HELP\*. *
maint\zoo -add maint\nradat12.zoo pdoxrun\*. *

rem
cls
echo NRA compression complete.
```

INSTALL.BAT

```
CLS
echo off
echo '24-Aug-91 JCP
rem revised 9/19/91 crw Text files now located in HELP and TEXT
echo '
echo '
echo '
echo                                     NRA
echo                                     NATIONAL RADIOBIOLOGY ARCHIVES
echo                                     Information Systems Demonstration

echo '
echo                                     Installation Procedure
echo '
echo                                     Pacific Northwest Laboratory
echo                                     Richland WA
echo '
echo                                     Contact: Sean K. Smith, (509) 376-5684
echo '
echo '
```

```

echo INSTRUCTIONS
echo Install command:  INSTALL (source drive) (destination drive)
echo           Example:  INSTALL A: C:
echo '
echo '
echo NRA will be decompressed from drive %1 to drive %2
echo Press CONTROL-BREAK if this is incorrect.
pause

%2

cd\

if exist \NRA\util\main.sc goto ask

goto makedirs

:ask
cls
echo '
echo '
echo '
echo A copy of NRA already exists on %2
echo '
echo Compressed NRA will overwrite existing NRA on %2
echo Press CONTROL-BREAK to cancel overwrite
pause

goto install

:makedirs
md NRA
cd NRA
md s
md s\v
md s\d
md b
md b\v
md b\d
md i
md i\v
md i\d
md w
md util
md text
md HELP
md pdoxrun
cd..
goto install

:install
cd NRA
%1\zoo e0 %1\nradat1.zoo
cd s\v
%1\zoo e0 %1\nradat2.zoo
cd..
cd d
%1\zoo e0 %1\nradat3.zoo
cd..
cd..
cd b\v

```

```

%1\zoo e0 %1\nradat4.zoo
cd..
cd d
%1\zoo e0 %1\nradat5.zoo
cd..
cd..
cd i\v
%1\zoo e0 %1\nradat6.zoo
cd..
cd d
%1\zoo e0 %1\nradat7.zoo
cd..
cd..
cd w
%1\zoo e0 %1\nradat8.zoo
cd..
cd util
%1\zoo e0 %1\nradat9.zoo
cd..
cd text
%1\zoo e0 %1\nradat10.zoo
cd ..
cd HELP
%1\zoo e0 %1\nradat11.zoo
cd..
cd pdoxrun

echo '
echo Enter NRA installation disk2 in %1
pause

%1\zoo e0 %1\nradat12.zoo
cd..
goto stop

:stop
cls
echo '
echo NRA successfully installed.
echo '
echo Type NRA to run application using RUNTIME.
pause
echo on

:end

```

MAKEDISK.BAT

```

cls
echo off
echo SEND COMPRESSED NRA TO DISK
rem
echo Insert 1st blank disk in drive %1
pause

copy maint\install.bat %1
copy maint\nradat1.zoo %1
copy maint\nradat2.zoo %1

```

```

copy maint\nradat3.zoo %1
copy maint\nradat4.zoo %1
copy maint\nradat5.zoo %1
copy maint\nradat6.zoo %1
copy maint\nradat7.zoo %1
copy maint\nradat8.zoo %1
copy maint\nradat9.zoo %1
copy maint\nradat10.zoo %1
copy maint\nradat11.zoo %1
copy maint\zoo.exe %1

echo '
echo Insert 2nd blank disk in drive %1
pause

copy maint\install.bat %1
copy maint\nradat12.zoo %1
copy maint\zoo.exe %1
cls
echo Operation completed.

```

NRA.BAT

```

echo off
echo One moment please...
path >util\oldpath.bat
path = pdoxrun
pdoxrun util\main
util\oldpath
cls

```

Paradox® SCRIPTS

MAINT\MAINTAIN.SC

```
;/*****
;* FILE: MAINT\MAINTAIN.SC
;* MODULE: 0.0
;* PURPOSE: Calls the first menu of the maintenance application.
;* CALLED BY:
;* CALLS:
;* LIBRARY CREATED: None
;* GLOBAL VARIABLES:
;* PROCEDURES IN MODULE: None
;*
;*
;* AUTHOR: Jonathan C. Prather, REVISED Chuck Watson
;* DATE: 8/11/91 REVISED 9/11/91
;*****/

;free memory
CLEAR
CURSOR OFF
RELEASE VARS ALL

;set autolibrary to run menu system
AUTOLIB = "MAINT\MENUUTIL,MAINT\MENUMADE"

;define error procedure from util\errorutil library
;
; also get procs for popup
;

READLIB "UTIL\DEMOUTIL" ERRORHANDLER, SETPOPUP, POPUP
ERRORPROC = "ERRORHANDLER"

; Call procedure to initialize global variables
INIT()

; Call the first menu of the application
MENUO()

;/*****
;* END MODULE: 0.0 MAINT\MAINTAIN.SC
;*****/
```

MAINT\MAINTUTL.SC

```

*****
;* FILE: MAINT\MAINTUTL.SC
;* MODULE:
;* PURPOSE: Holds utility procedures for NRADEMO maintenance
;* CALLED BY: MAINT\MAINTAIN.SC and the library made by MAKEMENU
;* CALLS:
;* LIBRARY CREATED: MAINT\MAINTUTL
;* GLOBAL VARIABLES:
;* PROCEDURES IN MODULE:  ADDTOSUM
;                          ADDTOBIB
;                          SEARCHBOX
;                          EMPTYSUM
;                          EMPTYINV
;                          EMPTYBIB
;                          COMPRESSDEMO
;                          MAKEDISK3
;                          MAKEDISK5
;*
;*
;* AUTHOR: Jonathan C. Prather, REVISED  Chuck Watson
;*
;* DATE: 8/23/91   REVISED  9/11/91, 9/18/91
;
;REVISION HISTORY:  1. Added F1 feature to ADDTOSUM.SC to allow user to select
;                    one or more animals from a list.
;                    2. Added logic to ADDTOSUM.SC to build a one line clinical
;                       history file for animals with no file on N:
;
;*****/

;put up working message
LIBNAME = "MAINT\MAINTUTL"
@24,0 CLEAR EOL ?? "CREATING "+LIBNAME+" LIBRARY"
STYLE BLINK ?? "... "
CREATELIB LIBNAME
STYLE

```

```

/*****
* PROCEDURE: ADDTOSUM()
* MODULE:
* PURPOSE: Move records from main NRA database on N drive to the demo
          version. User selects the summary database records to move
          by either:
            typing in an animal number
            typing in a lab_id and .. to get all animals from that lab
            choosing multiple animals within a lab using the F1 option
            selecting use of an existing table W\GETANI using F3
          NOTE
            File "NOFILE.TXT" must exist on N:\TEXT
* CALLED BY:
* CALLS:
* INPUT ASSERTION:
* OUTPUT ASSERTION:
* GLOBAL VARIABLES:
* AUTHOR: J. Prather
* DATE: 8/91
* REVISED 9/10/91 BY Chuck Watson to add F1 feature
* REVISED 9/18/91 BY Chuck Watson to create one line clinical text file
          for animals with out a file on N:\TEXT
*****/
PROC ADDTOSUM()

    PRIVATE SEARCHTITLE, SEARCHPROMPT, SEARCHWIDTH, SEARCHVAL, X, L, J, SHOWPRETTY1,
        SHOWPRETTY2, GC, GETLAB, SEARCHTABLE

    SEARCHTITLE = "ADD RECORDS TO NRADEMO SUMMARY"
    SEARCHPROMPT = "ANIMAL_ID"
    SEARCHWIDTH = 20
    SEARCHTABLE = "W\GETANI"

    WHILE TRUE
        SEARCHVAL = SEARCHBOX(SEARCHTITLE, SEARCHPROMPT, SEARCHWIDTH, SEARCHTABLE)

        CURSOR OFF
        SWITCH
            CASE SEARCHVAL = "" OR SEARCHVAL = " ": RETURN

            CASE SEARCHVAL = "F3": ; Use existing W\GETANI table
                ;***** put up message
                STYLE ATTRIBUTE HIGHLIGHT
                @ 24,59 ?? " ONE MOMENT PLEASE"
                STYLE ATTRIBUTE HIGHLIGHT+128
                @ 24,77 ?? "... "
                CANVAS ON
                CANVAS OFF
                ;****remove menu instructions and maintitle
                PAINTCANVAS ATTRIBUTE BKCOLOR 0,0,3,79
                PAINTCANVAS ATTRIBUTE BKCOLOR 24,0,24,79
                CLEARALL
                ;
                @18,10 ?? "You have chosen to use an existing table"
                ;
                @19,10 ?? "W\GETANI which is a subset of N:\S\D\ANI"
                ;
                @20,10 ?? "Hit any key to continue"
                ;
                GC = GETCHAR()
                QUITLOOP

```

```

CASE SEARCHVAL = "F1": ;GET POPUP OF CHOICES
;***** put up working message
STYLE ATTRIBUTE HIGHLIGHT
@ 24,59 ?? " ONE MOMENT PLEASE"
STYLE ATTRIBUTE HIGHLIGHT+128
@ 24,77 ?? "... "
CANVAS ON
CANVAS OFF
;****remove menu instructions and maintitle
PAINTCANVAS ATTRIBUTE BKCOLOR 0,0,3,79
PAINTCANVAS ATTRIBUTE BKCOLOR 24,0,24,79
CLEARALL
;initialize vars
ESCAPEFLAG = "N"
PRMPT1 = FORMAT("W80,AC","F2-Execute search for items selected
Esc-Return to menu")
PRMPT2 = FORMAT("W80,AC","ENTER-Select or un-select item      CURSOR
KEYS,PGUP,PGDN,HOME,END-Scroll list")
;GET POSSIBLE CHOICES TO ANSWER FOR POPUP
;
; First, get the LAB, then the animals
; because there are too many animals to fit in
; the array ITEM
;
(ASK)
TYPEIN "N:\\S\\D\\LAB"
ENTER
MOVETO FIELD "LAB_ID"
CHECK
DO IT!
;DO POPUP
SETPOPUP("ANSWER","LAB_ID")
PAINTCANVAS FILL " " ATTRIBUTE BKCOLOR 24,0,24,79
C=MAX(INT((80-WIDTH)/2)-3,0)
POPUP(4,C,13,1,"LAB ID ",PRMPT1,PRMPT2)
IF ESCAPEFLAG = "Y" THEN
RELEASE VARS ESCAPEFLAG,ITEM,WIDTH
RETURN
ELSE RELEASE VARS ESCAPEFLAG
ENDIF
;****GET the LAB code to use for next popup
FOR X FROM 1 TO ARRAYSIZE(ITEM)
IF SUBSTR(ITEM[X],1,1) = "*" THEN
GETLAB = SUBSTR(ITEM[X],2,1)
ENDIF
ENDFOR
;
;
*** POP UP the window with animals from that LAB
;
;***** put up working message
STYLE ATTRIBUTE HIGHLIGHT
@ 24,59 ?? " ONE MOMENT PLEASE"
STYLE ATTRIBUTE HIGHLIGHT+128
@ 24,77 ?? "... "
CANVAS ON
CANVAS OFF
;****remove menu instructions and maintitle
PAINTCANVAS ATTRIBUTE BKCOLOR 0,0,3,79
PAINTCANVAS ATTRIBUTE BKCOLOR 24,0,24,79
CLEARALL
;initialize vars

```

```

        RELEASE VARS ITEM, WIDTH
        ESCAPEFLAG = "N"
        PRMPT1 = FORMAT("W80,AC","F2-Execute search for items selected
Esc-Return to menu")
        PRMPT2 = FORMAT("W80,AC","ENTER-Select or un-select item      CURSOR
KEYS,PGUP,PGDN,HOME,END-Scroll list")
        {ASK}
        TYPEIN "N:\\S\\D\\ANI"
        ENTER
        MOVETO FIELD "ANIMAL_ID"
        CHECK
        TYPEIN STRVAL(GETLAB)+".."
        DO_IT!
;      @0,0 ?? STRVAL(GETLAB)+".."
;      GC = GETCHAR()
;      DO POPUP showing animal numbers
        SETPOPUP("ANSWER","ANIMAL_ID")
        PAINTCANVAS FILL " " ATTRIBUTE BKCOLOR 24,0,24,79
        C=MAX(INT((80-WIDTH)/2)-3,0)
        POPUP(4,C,13,1,"ANIMAL NUMBER",PRMPT1,PRMPT2)
        IF ESCAPEFLAG = "Y" THEN
            RELEASE VARS ESCAPEFLAG,ITEM,WIDTH
            RETURN
        ELSE RELEASE VARS ESCAPEFLAG
        ENDIF
;      *****DO SEARCH:
;      ***** put up working message
        STYLE ATTRIBUTE HIGHLIGHT
        @ 24,48 ?? " SEARCHING, O'E MOMENT PLEASE"
        STYLE ATTRIBUTE HIGHLIGHT+ 28
        @ 24,77 ?? "... "
        CLEARALL
        CANVAS ON
        CANVAS OFF
        CURSOR OFF
;      ***
        {ASK}
        TYPEIN "N:\\S\\D\\ANI"
        ENTER
        L=0
        FOR X FROM 1 TO ARRAYSIZE(ITEM)
            IF SUBSTR(ITEM[X],1,1) = "*" THEN
                LEFT
                CHECK
                MOVETO FIELD "ANIMAL_ID"
                TYPEIN SUBSTR(ITEM[X],2,LEN(ITEM[X])-1)+".."
                DOWN
            ENDIF
        ENDFOR
        UP
        IF ISBLANK([]) THEN RETURN
        ENDIF
        DO_IT!
;
; The answer table now contains the ANI records requested.
; These will be used to populate the other tables,
; so rename the table to save it.

        RENAME "ANSWER" "W\\GETANI"
        QUITLOOP

    OTHERWISE: ;SEARCH ON SEARCHVAL

```

```

;***** put up working message
STYLE ATTRIBUTE HIGHLIGHT
@ 24,48 ?? " SEARCHING, ONE MOMENT PLEASE"
STYLE ATTRIBUTE HIGHLIGHT-128
@ 24,77 ?? "... "
CLEARALL
CANVAS ON
CANVAS OFF
CURSOR OFF
(ASK)
;***
TYPEIN "N:\\S\\D\\ANI"
ENTER
CHECK
MOVETO FIELD "ANIMAL_ID"
TYPEIN SEARCHVAL
DO_IT!

;
; The answer table now contains the ANI records requested.
; These will be used to populate the other tables,
; so rename the table to save it.

RENAME "ANSWER" "W\\GETANI"
QUITLOOP

ENDSWITCH

ENDWHILE

;SHOW HOW MANY FOUND

STYLE ATTRIBUTE HIGHLIGHT
SHOWPRETTY1 = "MATCH"
SHOWPRETTY2 = "RECORD"

NUMBERFOUND = NRECORDS("W\\GETANI")

IF NUMBERFOUND > 1 THEN
    SHOWPRETTY1 = "MATCHES"
    SHOWPRETTY2 = "RECORDS"
ENDIF

;DRAW BOX FOR MESSAGE
PAINTCANVAS ATTRIBUTE BKCOLOR 24,0,24,79
@ 6,1 ?? " "+FILL("-",76)+" "
@ 7,1 ?? " "+FILL(" ",76)+" "
@ 8,1 ?? " "+FILL(" ",76)+" "
@ 9,1 ?? " "+FILL(" ",76)+" "
@ 10,1 ?? " "+FILL(" ",76)+" "
@ 11,1 ?? " "+FILL("-",76)+" "
CANVAS ON

IF IEMPTY("W\\GETANI") THEN
    @ 8,6 ?? FORMAT("W68,AC","NO MATCHES FOUND")
    @ 9,6 ?? FORMAT("W68,AC","PRESS ANY KEY TO CONTINUE...")
    GC = GETCHAR()
    RETURN

```

```

ELSE @ 8,6 ?? FORMAT("W68,AC",STRVAL(NUMBERFOUND)+" "+SHOWPRETTY1+" FOUND")
@ 9,6 ?? FORMAT("W68,AC"," PRESS ANY KEY TO CONTINUE...")
GC = GETCHAR()
ENDIF

CLEARALL
;***** put up working message
;DRAW BOX FOR MESSAGE
PAINTCANVAS ATTRIBUTE BKCOLOR 24,0,24,79
@ 6,5 ?? " "+FILL(" ",68)+" "
@ 7,5 ?? " "+FILL(" ",68)+" "
@ 8,5 ?? " "+FILL(" ",68)+" "
@ 9,5 ?? " "+FILL(" ",68)+" "
@ 10,5 ?? " "+FILL(" ",68)+" "
@ 11,5 ?? " "+FILL(" ",68)+" "
CANVAS ON
@ 8,6 ?? FORMAT("W68,AC","ADDING SUMMARY (&INV) RECORDS TO NRADEMO")
@ 9,6 ?? FORMAT("W68,AC","CURRENT STEP:")
CLEARALL
CANVAS ON
CURSOR OFF
;***
;FIRST, GET ANI RECORDS:
@ 10,6 ?? FORMAT("W68,AC","S\\D\\ANI")

ADD "W\\GETANI" "S\\D\\ANI"

;GET CLINICAL
@ 10,6 ?? FORMAT("W68,AC","CLINICAL TEXT")
CLEARALL
{ASK}
TYPEIN "W\\GETANI"
ENTER
MOVETO FIELD "ANIMAL_ID"
CHECK
MOVETO FIELD "LAB_ID"
CHECK
MOVETO FIELD "STUDY_ID"
CHECK
MOVETO FIELD "GROUP_ID"
CHECK
DO_IT!
ARRAY CLIN_PATH[NRECORDS("ANSWER")]
ARRAY CLIN_FILE[NRECORDS("ANSWER")]
VIEW "ANSWER"
SCAN
CLIN_PATH[[#]] = "TEXT\\LAB_"+[LAB_ID]+"\\STU_"+STRVAL([STUDY_ID])+
"\\GRO_"+STRVAL([GROUP_ID])+"\\\"
CLIN_FILE[[#]] = [ANIMAL_ID]+".TXT"
ENDSCAN
IF ISTABLE("W\\TEXT") AND NOT ISEMPY ("W\\TEXT") THEN
EMPTY "W\\TEXT"
ENDIF

EDIT "W\\TEXT"
[TEXT]= "CD E:\\\\NRADEMO"
ENTER
FOR X FROM 1 TO ARRAYSIZE(CLIN_PATH)
@15,15 ?? CLIN_PATH[X] + CLIN_FILE[X]
IF ISFILE("N:\\\\"+CLIN_PATH[X]+CLIN_FILE[X])
THEN

```

```

[TEXT] = "COPY N:\\\\"+CLIN_PATH[X]+CLIN_FILE[X]+" TEXT\\\\"+CLIN_FILE[X]
ENTER
ELSE
[TEXT] = "COPY N:\\TEXT\\NOFILE.DOC TEXT\\\\"+CLIN_FILE[X]
ENDIF
ENTER
ENDFOR
DO_IT!
IF ISFILE ("W\\CLINCOPY.BAT")
THEN
Menu {Tools} {ExportImport} {Export} {Ascii}
{Text} {W\\TEXT} {W\\CLINCOPY.BAT} {Replace}
ELSE
Menu {Tools} {ExportImport} {Export} {Ascii}
{Text} {W\\TEXT} {W\\CLINCOPY.BAT}
ENDIF
RUN "W\\CLINCOPY.BAT"

RELEASE VARS CLIN_PATH, CLIN_FILE

;GET CONTROL RECORDS:
@ 10,6 ?? FORMAT("W68,AC","S\\D\\CONTROL")
CLEARALL
{ASK}
TYPEIN "W\\GETANI"
ENTER
MOVETO FIELD "ANIMAL_ID"
EXAMPLE
TYPEIN "ID"
{ASK}
TYPEIN "N:\\S\\D\\CONTROL"
ENTER
CHECK
MOVETO FIELD "ANIMAL_ID"
EXAMPLE
TYPEIN "ID"
DO_IT!
ADD "ANSWER" "S\\D\\CONTROL"

;GET TDOSE RECORDS
@ 10,6 ?? FORMAT("W68,AC","S\\D\\TDOSE")
CLEARALL
{ASK}
TYPEIN "W\\GETANI"
ENTER
MOVETO FIELD "ANIMAL_ID"
EXAMPLE
TYPEIN "ID"
{ASK}
TYPEIN "N:\\S\\D\\TDOSE"
ENTER
CHECK
MOVETO FIELD "ANIMAL_ID"
EXAMPLE
TYPEIN "ID"
DO_IT!
ADD "ANSWER" "S\\D\\TDOSE"

; get tdose1 (with translations)
; GET W\\Tdose1 records with translations included
@ 10,6 ?? FORMAT("W68,AC","W\\TDOSE1")
CLEARALL

```

Query

```

S\d\tdose | ANIMAL_ID | TISSUE_CATEGORY | DOSE_UNITS | LAST_NORMAL |
           | _id      | _tcat      | _b          | _c          |

S\d\tdose | TYPE_LAST_NORMAL | DOSE_LAST_NORMAL | DOSE_ERR_LAST_NORMAL |
           | _d              | _e              | _f                    |

S\d\tdose | DATE_CLIN_DX | TYPE_CLIN_DX | DOSE_CLIN_DX | DOSE_ERR_CLIN_DX |
           | _g          | _h          | _i          | _j              |

S\d\tdose | DATE_TISS_DX | TYPE_TISS_DX | DOSE_TISS_DX | DOSE_ERR_TISS_DX |
           | _k          | _l          | _m          | _n              |

S\d\tdose | ACCESSION_DATE |
           | _o              |

```

```

N:s\v\tcat | CODE | TRANS |
           | _tcat | _trans |

```

```

W\tdose1 | ANIMAL_ID | TISSUE_CATEGORY | DOSE_UNITS | LAST_NORMAL |
insert   | _id      | _tcat          | _b          | _c          |

```

```

W\tdose1 | TYPE_LAST_NORMAL | DOSE_LAST_NORMAL | DOSE_ERR_LAST_NORMAL |
           | _d              | _e              | _f                    |

```

```

W\tdose1 | DATE_CLIN_DX | TYPE_CLIN_DX | DOSE_CLIN_DX | DOSE_ERR_CLIN_DX |
           | _g          | _h          | _i          | _j              |

```

```

W\tdose1 | DATE_TISS_DX | TYPE_TISS_DX | DOSE_TISS_DX | DOSE_ERR_TISS_DX |
           | _k          | _l          | _m          | _n              |

```

```

W\tdose1 | ACCESSION_DATE | TISSUE_TRANS |
           | _o              | _trans       |

```

Endquery
do_it!

```

;GET TEFFECT RECORDS
@ 10,6 ?? FORMAT("W68,AC","S\\D\\TEFFECT")
CLEARALL

```

Query

```

W\getani | ANIMAL_ID |
           | _ID        |

```

```

N:s\d\teffect | ANIMAL_ID | TISSUE_CATEGORY | MORP_CATEGORY | DX |
               | _ID        | _TOPO           | _MOR           | _DX |

```

```

N:s\d\teffect | ACCESSION_DATE |
               | _ADATE          |

s\d\teffect | ANIMAL_ID | TISSUE_CATEGORY | MORP_CATEGORY | DX |
INSERT      | _ID       | _TOPO          | _MOR          | _DX |

s\d\teffect | ACCESSION_DATE |
               | _ADATE          |

```

ENDQUERY
DO_IT!

```

; GET W\TEFFECT1 records with translations included
@ 10,6 ?? FORMAT("W68,AC","W\TEFFECT1")
CLEARALL

```

Query

```

s\d\teffect | ANIMAL_ID | TISSUE_CATEGORY | MORP_CATEGORY | DX |
             | _ID       | _TOPO          | _MOR          | _DX |

s\d\teffect | ACCESSION_DATE |
             | _ADATE          |

```

```

N:s\v\mcat | CODE | TRANS |
            | _MOR | _MORTAN |

```

```

W\teffect1 | ANIMAL_ID | TISSUE_CATEGORY | MORP_CATEGORY | DX |
INSERT      | _ID       | _TOPO          | _MOR          | _DX |

```

```

W\teffect1 | ACCESSION_DATE | MORP_TRANS |
            | _ADATE          | _MORTAN    |

```

Endquery
DO_IT!

```

QUERY
W\teffect1 | TISSUE_CATEGORY | TISSUE_TRANS |
            | _TOPO          | CHANGETO _TOPTRAN |

```

```

N:s\v\tcac | CODE | TRANS |
            | _TOPO | _TOPTRAN |

```

ENDQUERY
DO_IT!

```

;GET DETAIL RECORDS
@ 10,6 ?? FORMAT("W68,AC","I\D\DETAIL")
CLEARALL
(ASK)
TYPEIN "W\GETANI"
ENTER
MOVETO FIELD "ANIMAL_ID"
EXAMPLE
TYPEIN "ID"

```

```

(ASK)
TYPEIN "N:\\I\\D\\DETAIL"
ENTER
CHECK
MOVETO FIELD "ANIMAL_ID"
EXAMPLE
TYPEIN "ID"
DO_IT!
ADD "ANSWER" "I\\D\\DETAIL"

;GET LOCATION RECORDS
@ 10,6 ?? FORMAT("W68,AC","I\\D\\LOCATION")
CLEARALL
(ASK)
TYPEIN "W\\GETANI"
ENTER
MOVETO FIELD "ANIMAL_ID"
EXAMPLE
TYPEIN "ID"
(ASK)
TYPEIN "N:\\I\\D\\DETAIL"
ENTER
MOVETO FIELD "BAR_CODE"
EXAMPLE
TYPEIN "A"
MOVETO FIELD "ANIMAL_ID"
EXAMPLE
TYPEIN "ID"
(ASK)
TYPEIN "N:\\I\\D\\LOCATION"
ENTER
CHECK
MOVETO FIELD "BAR_CODE"
EXAMPLE
TYPEIN "A"
DO_IT!
ADD "ANSWER" "I\\D\\LOCATION"

;GET MASTER RECORDS
@ 10,6 ?? FORMAT("W68,AC","I\\D\\MASTER")
CLEARALL
(ASK)
TYPEIN "W\\GETANI"
ENTER
MOVETO FIELD "ANIMAL_ID"
EXAMPLE
TYPEIN "ID"
(ASK)
TYPEIN "N:\\I\\D\\DETAIL"
ENTER
MOVETO FIELD "BAR_CODE"
EXAMPLE
TYPEIN "A"
MOVETO FIELD "ANIMAL_ID"
EXAMPLE
TYPEIN "ID"
(ASK)
TYPEIN "N:\\I\\D\\MASTER"
ENTER
CHECK
MOVETO FIELD "BAR_CODE"
EXAMPLE

```

```

        TYPEIN "A"
        DO_IT!
        ADD "ANSWER" "I\\D\\MASTER"

;ENDSWITCH
;ENDWHILE

;SHOW HOW MANY FOUND

;DRAW BOX FOR MESSAGE
PAINTCANVAS ATTRIBUTE BKCOLOR 24,0,24,79
a 6,5 ?? " "+FILL("-",68)+" "
a 7,5 ?? " "+FILL(" ",68)+" "
a 8,5 ?? " "+FILL(" ",68)+" "
a 9,5 ?? " "+FILL(" ",68)+" "
a 10,5 ?? " "+FILL(" ",68)+" "
a 11,5 ?? " "+FILL("-",68)+" "
CANVAS ON

        a 8,6 ?? FORMAT("W68,AC",STRVAL(NUMBERFOUND)+" ANIMAL RECORD(S) ADDED")
        a 9,6 ?? FORMAT("W68,AC","PRESS ANY KEY TO CONTINUE")
        GC = GETCHAR()

CLEARALL

ENDPROC
WRITELIB LIBNAME ADDTOSUM
RELEASE PROCS ADDTOSUM
;*****
;* END MODULE: x.x ADDTOSUM()
;*****/

```

```

;*****
; * PROCEDURE: ADDTOBIB()
; * MODULE:
; * PURPOSE: Move records from main NRA database on N drive to the demo
;            version. User selects the Bibliography database records to move
;            by either:
;                typing in a bar code
;                typing in a bar code prefix and .. to get all records in that series
;                choosing multiple bar codes using the F1 option
;                F1 IS NOT IMPLEMENTED
;                selecting use of an existing table W\GETBOOK using F3
;
;
; * CALLED BY:
; * CALLS:
; * INPUT ASSERTION:
; * OUTPUT ASSERTION:
; * GLOBAL VARIABLES:
; * AUTHOR: J Prather, modified by Chuck Watson
; * DATE: 8/91, modified 9/12/91 to allow F3 and follow style of ADDTOSUM
;*****/
PROC ADDTOBIB()
    PRIVATE SEARCHTITLE, SEARCHPROMPT, SEARCHWIDTH, SEARCHVAL, X, L, J, SHOWPRETTY1,
        SHOWPRETTY2, GC, C, NUMBERFOUND, SEARCHTABLE

    SEARCHTITLE = "ADD RECORDS TO NRADEMO BIBLIOGRAPHY"
    SEARCHPROMPT = "BAR_CODE:"
    SEARCHWIDTH = 20
    SEARCHTABLE = "W\GETBOOK"

    WHILE TRUE
        SEARCHVAL = SEARCHBOX(SEARCHTITLE, SEARCHPROMPT, SEARCHWIDTH, SEARCHTABLE)

        CURSOR OFF
        SWITCH
            CASE SEARCHVAL = "" OR SEARCHVAL = " ": RETURN

            CASE SEARCHVAL = "F1":

                @15,10 ?? "F1 option not implemented, hit any key to continue"
                GC = GETCHAR()
                RETURN

            CASE SEARCHVAL = "F3":

                QUITLOOP

        OTHERWISE: ;SEARCH ON SEARCHVAL
            ;***** put up working message
            STYLE ATTRIBUTE HIGHLIGHT
            @ 24,48 ?? " SEARCHING, ONE MOMENT PLEASE"
            STYLE ATTRIBUTE HIGHLIGHT+128
            @ 24,77 ?? "..."
            CLEARALL
            CANVAS ON
            CANVAS OFF
            CURSOR OFF
            {ASK}
            TYPEIN "N:\\B\\D\\BOOK"
            ENTER
            CHECK

```

```

        MOVETO FIELD "BAR_CODE"
        TYPEIN SEARCHVAL
        DO_IT!
;
; The answer table now contains the BOOK records requested.
; These will be used to populate the other tables,
; so rename the table to save it.

        RENAME "ANSWER" "W\\GETBOOK"
        QUITLOOP

ENDSWITCH

ENDWHILE

;SHOW HOW MANY FOUND

STYLE ATTRIBUTE HIGHLIGHT
SHOWPRETTY1 = "MATCH"
SHOWPRETTY2 = "RECORD"

NUMBERFOUND = NRECORDS("W\\GETBOOK")

IF NUMBERFOUND > 1 THEN
    SHOWPRETTY1 = "MATCHES"
    SHOWPRETTY2 = "RECORDS"
ENDIF

;DRAW BOX FOR MESSAGE
PAINTCANVAS ATTRIBUTE BKCOLOR 24,0,24,79
@ 6,1 ?? " "+FILL(" ",76)+" "
@ 7,1 ?? " "+FILL(" ",76)+" "
@ 8,1 ?? " "+FILL(" ",76)+" "
@ 9,1 ?? " "+FILL(" ",76)+" "
@ 10,1 ?? " "+FILL(" ",76)+" "
@ 11,1 ?? " "+FILL(" ",76)+" "
CANVAS ON

IF IEMPTY("W\\GETBOOK") THEN
    @ 8,6 ?? FORMAT("W68,AC","NO MATCHES FOUND")
    @ 9,6 ?? FORMAT("W68,AC","  PRESS ANY KEY TO CONTINUE...")
    GC = GETCHAR()
    RETURN
ELSE @ 8,6 ?? FORMAT("W68,AC",STRVAL(NUMBERFOUND)+" "+SHOWPRETTY1+" FOUND")
    @ 9,6 ?? FORMAT("W68,AC","  PRESS ANY KEY TO CONTINUE...")
    GC = GETCHAR()
ENDIF

CLEARALL
;***** put up working message
;DRAW BOX FOR MESSAGE
PAINTCANVAS ATTRIBUTE BKCOLOR 24,0,24,79
@ 6,5 ?? " "+FILL(" ",68)+" "
@ 7,5 ?? " "+FILL(" ",68)+" "
@ 8,5 ?? " "+FILL(" ",68)+" "
@ 9,5 ?? " "+FILL(" ",68)+" "
@ 10,5 ?? " "+FILL(" ",68)+" "
@ 11,5 ?? " "+FILL(" ",68)+" "
CANVAS ON

```

```

@ 8,6 ?? FORMAT("W68,AC","ADDING BIBLIOGRAPHY (&INV) RECORDS TO NRADEMO")
@ 9,6 ?? FORMAT("W68,AC","CURRENT STEP:")
      CLEARALL
      CANVAS ON
      CURSOR OFF
      ;***
;FIRST, MOVE BOOK RECORDS .
@ 10,6 ?? FORMAT("W68,AC","B\\D\\BOOK")

      ADD "W\\GETBOOK" "B\\D\\BOOK"


; GET AUTHOR RECORDS
@ 10,6 ?? FORMAT("W68,AC","B\\D\\AUTHOR")
      CLEARALL
      {ASK}
      TYPEIN "W\\GETBOOK"
      ENTER
      MOVETO FIELD "BAR_CODE"
      EXAMPLE
      TYPEIN "BAR"
      {ASK}
      TYPEIN "N:\\B\\D\\AUTHOR"
      ENTER
      CHECK
      MOVETO FIELD "BAR_CODE"
      EXAMPLE
      TYPEIN "BAR"
      DO_IT!
      ADD "ANSWER" "B\\D\\AUTHOR"


;GET LOCATION RECORDS
@ 10,6 ?? FORMAT("W68,AC","I\\D\\LOCATION")
      CLEARALL
      {ASK}
      TYPEIN "W\\GETBOOK"
      ENTER
      MOVETO FIELD "BAR_CODE"
      EXAMPLE
      TYPEIN "BAR"
      {ASK}
      TYPEIN "N:\\I\\D\\LOCATION"
      ENTER
      CHECK
      MOVETO FIELD "BAR_CODE"
      EXAMPLE
      TYPEIN "BAR"
      DO_IT!
      ADD "ANSWER" "I\\D\\LOCATION"


;GET MASTER RECORDS
@ 10,6 ?? FORMAT("W68,AC","I\\D\\MASTER")
      CLEARALL
      {ASK}
      TYPEIN "W\\GETBOOK"
      ENTER
      MOVETO FIELD "BAR_CODE"
      EXAMPLE
      TYPEIN "BAR"

```

```

(ASK)
TYPEIN "N:\\I\\D\\MASTER"
ENTER
CHECK
MOVETO FIELD "BAR_CODE"
EXAMPLE
TYPEIN "BAR"
DO_IT!
ADD "ANSWER" "I\\D\\MASTER"

```

```

;SHOW HOW MANY FOUND

```

```

;DRAW BOX FOR MESSAGE
PAINTCANVAS ATTRIBUTE BKCOLOR 24,0,24,79
@ 6,5 ?? "+"FILL("-",68)+"|"
@ 7,5 ?? "+"FILL(" ",68)+"|"
@ 8,5 ?? "+"FILL(" ",68)+"|"
@ 9,5 ?? "+"FILL(" ",68)+"|"
@ 10,5 ?? "+"FILL(" ",68)+"|"
@ 11,5 ?? "+"FILL("-",68)+"|"
CANVAS ON

@ 8,6 ?? FORMAT("W68,AC",STRVAL(NUMBERFOUND)+" BIBLIOGRAPHY RECORD(S) ADDED")
@ 9,6 ?? FORMAT("W68,AC","PRESS ANY KEY TO CONTINUE")
GC = GETCHAR()

```

```

CLEARALL

```

```

ENDPROC
WRITELIB LIBNAME ADDTOBIB
RELEASE PROCS ADDTOBIB
;*****
;* END MODULE: x.x ADDTOBIB()
;*****/

```

```

;*****
;* PROCEDURE: SEARCHBOX()
;* MODULE:
;* PURPOSE: General purpose proc to display a search box. This one accepts
;           several values from the user:
;           F1 - pops up help windows
;           F3 - uses SEARCHTABLE rather than performing a search
;           SEARCHVAL - searches for value entered by user
;           ESCAPE - erases window and returns
;           else - continues to prompt
;* CALLED BY:
;* CALLS:
;* INPUT ASSERTION:
;* OUTPUT ASSERTION:
;* GLOBAL VARIABLES:
;* AUTHOR: J Prather, modified by Chuck Watson
;* DATE: 8/91, modified 9/12/91 to allow F3 options BOX is one line larger
;*****/
PROC SEARCHBOX(SEARCHTITLE,SEARCHPROMPT,SEARCHWIDTH,SEARCHTABLE)
PRIVATE SEARCHVAL,MESSAGE1,MESSAGE2,BOXWIDTH,STARTCOLUMN,THECOLUMN,TITLECOLUMN,
PROMPTCOLUMN,MESSAGE1COLUMN,MESSAGE2COLUMN,THEROW,NEXTCHAR, MESSAGE3

PAINTCANVAS ATTRIBUTE BKCOLOR 24,0,24,79

SEARCHVAL = ""
MESSAGE1="Type in search string (wildcard = ..)"
MESSAGE2="[F1]-view list of choices [F2],[ENTER]-execute search"
MESSAGE3="[F3] to use existing table "+STRVAL(SEARCHTABLE)
BOXWIDTH = MAX(65,MAX((SEARCHWIDTH)+6,LEN(SEARCHTITLE)+2))
STARTCOLUMN = INT((80-BOXWIDTH)/2)

THECOLUMN = STARTCOLUMN+INT((BOXWIDTH-SEARCHWIDTH)/2)
TITLECOLUMN = STARTCOLUMN+INT((BOXWIDTH-LEN(SEARCHTITLE))/2)
PROMPTCOLUMN = STARTCOLUMN+INT((BOXWIDTH-LEN(SEARCHPROMPT))/2)
MESSAGE1COLUMN = STARTCOLUMN+INT((BOXWIDTH-LEN(MESSAGE1))/2)
MESSAGE2COLUMN = STARTCOLUMN+INT((BOXWIDTH-LEN(MESSAGE2))/2)
THEROW = 8

STYLE ATTRIBUTE HIGHLIGHT
@ 6,STARTCOLUMN ?? " "+FILL("-",BOXWIDTH-2)+" "
@ 7,STARTCOLUMN ?? " "+FILL(" ",BOXWIDTH-2)+" "
@ 8,STARTCOLUMN ?? " "+FILL(" ",BOXWIDTH-2)+" "
@ 9,STARTCOLUMN ?? " "+FILL(" ",BOXWIDTH-2)+" "
@ 10,STARTCOLUMN ?? " "+FILL(" ",BOXWIDTH-2)+" "
@ 11,STARTCOLUMN ?? " "+FILL(" ",BOXWIDTH-2)+" "
@ 12,STARTCOLUMN ?? " "+FILL("-",BOXWIDTH-2)+" "
@ 6,TITLECOLUMN ?? SEARCHTITLE
@ 7,PROMPTCOLUMN ?? SEARCHPROMPT
@ 9,MESSAGE1COLUMN ?? MESSAGE1
@ 10,MESSAGE2COLUMN ?? MESSAGE2
@ 11,MESSAGE2COLUMN ?? MESSAGE3
STYLE ATTRIBUTE 31
@ 8,THECOLUMN ?? FILL(" ",SEARCHWIDTH)

WHILE TRUE
@ THEROW,THECOLUMN ?? SEARCHVAL
CURSOR NORMAL
NEXTCHAR = GETCHAR()

```

```

SWITCH
CASE NEXTCHAR = 27 :      ;ESC
    RETURN ""

CASE NEXTCHAR = -59 :    ;F1
    RETURN "F1"

CASE NEXTCHAR = -61 :    ;F3
    RETURN "F3"

CASE NEXTCHAR = -60 OR NEXTCHAR = 13:    ;F2,ENTER
    RETURN SEARCHVAL

CASE 31 < NEXTCHAR AND NEXTCHAR < 127:
    IF LEN(SEARCHVAL) < SEARCHWIDTH THEN
        SEARCHVAL = SEARCHVAL + CHR(NEXTCHAR)
    ELSE BEEP
    ENDIF

CASE NEXTCHAR = 8 OR NEXTCHAR = -75:    ;BKSPACE OR LEFT
    SEARCHVAL = SUBSTR(SEARCHVAL,1,LEN(SEARCHVAL)-1)
    @ THEROW,THECOLUMN ?? SEARCHVAL+" "

ENDSWITCH
ENDWHILE

ENDPROC
WRITELIB !LIBNAME SEARCHBOX
RELEASE PROCS SEARCHBOX
; /*****
; * END MODULE: X.X SEARCHBOX()
; *****/

```

```

; /*****
; * PROCEDURE: EMPTYSUM()
; * MODULE:
; * PURPOSE: empty NRADEMO's summary tables, and replace contents of
; *           LAB, STU, and GRO
; * CALLED BY:
; * CALLS:
; * INPUT ASSERTION:
; * OUTPUT ASSERTION:
; * GLOBAL VARIABLES:
; * AUTHOR: JC Prather
; * DATE: August 1991
; *****/
PROC EMPTYSUM()

STYLE ATTRIBUTE HIGHLIGHT
;DRAW BOX FOR MESSAGE
PAINTCANVAS ATTRIBUTE BKCOLOR 24,0,24,79
@ 6,5 ?? " "+FILL(" ",68)+" "
@ 7,5 ?? " "+FILL(" ",68)+" "
@ 8,5 ?? " "+FILL(" ",68)+" "
@ 9,5 ?? " "+FILL(" ",68)+" "
@ 10,5 ?? " "+FILL(" ",68)+" "
@ 11,5 ?? " "+FILL(" ",68)+" "

@ 8,6 ?? FORMAT("W68,AC","EMPTYING SUMMARY TABLES, CURRENT STEP:")
@ 9,6 ?? FORMAT("W68,AC","")

CANVAS ON

@ 9,6 ?? FORMAT("W68,AC","emptying S\\D\\ANI table")
EMPTY "S\\D\\ANI"
RESET

@ 9,6 ?? FORMAT("W68,AC","emptying S\\D\\CONTROL table")
EMPTY "S\\D\\CONTROL"
RESET

@ 9,6 ?? FORMAT("W68,AC","emptying S\\D\\TDOSE table")
EMPTY "S\\D\\TDOSE"
RESET

@ 9,6 ?? FORMAT("W68,AC","emptying S\\D\\TEFFECT table")
EMPTY "S\\D\\TEFFECT"
RESET

@ 9,6 ?? FORMAT("W68,AC","emptying S\\D\\GRO table")
EMPTY "S\\D\\GRO"
RESET

@ 9,6 ?? FORMAT("W68,AC","adding contents of N:\\S\\D\\GRO")
ADD "N:\\S\\D\\GRO" "S\\D\\GRO"

@ 9,6 ?? FORMAT("W68,AC","emptying S\\D\\LAB table")
EMPTY "S\\D\\LAB"
RESET

@ 9,6 ?? FORMAT("W68,AC","adding contents of N:\\S\\D\\LAB")

```

```

ADD "N:\\S\\D\\LAB" "S\\D\\LAB"

@ 9,6 ?? FORMAT("W68,AC","emptying S\\D\\STU table")
EMPTY "S\\D\\STU"
RESET

@ 9,6 ?? FORMAT("W68,AC","adding contents of N:\\S\\D\\STU")
ADD "N:\\S\\D\\STU" "S\\D\\STU"

@ 9,6 ?? FORMAT("W68,AC","emptying W\\TEFFECT1 table")
EMPTY "W\\TEFFECT1"
RESET

@ 9,6 ?? FORMAT("W68,AC","emptying W\\TDOSE1 table")
EMPTY "W\\TDOSE1"
RESET

@ 9,6 ?? FORMAT("W68,AC","deleting clinical text files.")
RUN "DEL E:\\NRADEMO\\TEXT\\*.TXT "

@ 8,6 ?? FORMAT("W68,AC","SUMMARY TABLES HAVE BEEN EMPTIFD")
@ 9,6 ?? FORMAT("W68,AC","Press any key to continue...")

GC = GETCHAR()

ENDPROC
WRITELIB LIBNAME EMPTYSUM
RELEASE PROCS EMPTYSUM
;*****
; * END MODULE: x.x EMPTYSUM()
;*****/

```

```

; /*****
; * PROCEDURE: EMPTYINV()
; * MODULE:
; * PURPOSE: Empty NRADEMO's inventory tables
; * CALLED BY:
; * CALLS:
; * INPUT ASSERTION:
; * OUTPUT ASSERTION:
; * GLOBAL VARIABLES:
; * AUTHOR: JCP
; * DATE: August 1991
; *****/
PROC EMPTYINV()

STYLE ATTRIBUTE HIGHLIGHT
;DRAW BOX FOR MESSAGE
PAINTCANVAS ATTRIBUTE BKCOLOR 24,0,24,79
@ 6,5 ?? " "+FILL("-",68)+" "
@ 7,5 ?? " "+FILL(" ",68)+" "
@ 8,5 ?? " "+FILL(" ",68)+" "
@ 9,5 ?? " "+FILL(" ",68)+" "
@ 10,5 ?? " "+FILL(" ",68)+" "
@ 11,5 ?? " "+FILL("-",68)+" "

@ 8,6 ?? FORMAT("W68,AC","EMPTYING INVENTORY TABLES, CURRENT STEP:")
@ 9,6 ?? FORMAT("W68,AC","")

CANVAS ON

@ 9,6 ?? FORMAT("W68,AC","emptying I\\D\\DETAIL table")
EMPTY "I\\D\\DETAIL"
RESET

@ 9,6 ?? FORMAT("W68,AC","emptying I\\D\\LOCATION table")
EMPTY "I\\D\\LOCATION"
RESET

@ 9,6 ?? FORMAT("W68,AC","emptying I\\D\\MASTER table")
EMPTY "I\\D\\MASTER"

RESET

@ 8,6 ?? FORMAT("W68,AC","INVENTORY TABLES HAVE BEEN EMPTIED")
@ 9,6 ?? FORMAT("W68,AC","Press any key to continue...")

GC = GETCHAR()

ENDPROC
WRITELIB LIBNAME EMPTYINV
RELEASE PROCS EMPTYINV
; /*****
; * END MODULE: x.x EMPTYINV()
; *****/

```

```

; /*****
; * PROCEDURE: EMPTYBIB()
; * MODULE:
; * PURPOSE: Empty NRADEMO's bibliography tables
; * CALLED BY:
; * CALLS:
; * INPUT ASSERTION:
; * OUTPUT ASSERTION:
; * GLOBAL VARIABLES:
; * AUTHOR: JC Prather
; * DATE: August 1991
; *****/
PROC EMPTYBIB()

STYLE ATTRIBUTE HIGHLIGHT
;DRAW BOX FOR MESSAGE
PAINTCANVAS ATTRIBUTE BKCOLOR 24,0,24,79
@ 6,5 ?? " "+FILL(" ",68)+" "
@ 7,5 ?? " "+FILL(" ",68)+" "
@ 8,5 ?? " "+FILL(" ",68)+" "
@ 9,5 ?? " "+FILL(" ",68)+" "
@ 10,5 ?? " "+FILL(" ",68)+" "
@ 11,5 ?? " "+FILL(" ",68)+" "

@ 8,6 ?? FORMAT("W68,AC","EMPTYING BIBLIOGRAPHY TABLES, CURRENT STEP:")
@ 9,6 ?? FORMAT("W68,AC","")

CANVAS ON

@ 9,6 ?? FORMAT("W68,AC","emptying B\\D\\BOOK table")
EMPTY "B\\D\\BOOK"
RESET

@ 9,6 ?? FORMAT("W68,AC","emptying B\\D\\AUTHOR table")
EMPTY "B\\D\\AUTHOR"
RESET

@ 8,6 ?? FORMAT("W68,AC","BIBLIOGRAPHY TABLES HAVE BEEN EMPTIED")
@ 9,6 ?? FORMAT("W68,AC","Press any key to continue...")

GC = GETCHAR()

ENDPROC
WRITELIB LIBNAME EMPTYBIB
RELEASE PROCS EMPTYBIB
; /*****
; * END MODULE: x.x EMPTYBIB()
; *****/

```

```

;*****
;* PROCEDURE: COMPRESSDEMO()
;* MODULE:
;* PURPOSE: Compresses the Demo and its data tables for export
;* CALLED BY:
;* CALLS:
;* INPUT ASSERTION:
;* OUTPUT ASSERTION:
;* GLOBAL VARIABLES:
;* AUTHOR: JC Prather
;* DATE: August 1991
;*****/
PROC COMPRESSDEMO()

    ;show pop box
    CANVAS OFF
    PAINTCANVAS ATTRIBUTE BKCOLOR 24,0,24,79
    STYLE ATTRIBUTE HIGHLIGHT

    @ 6,5 ?? " " + FILL(" ",70) + " "
    @ 7,5 ?? " " + FILL(" ",70) + " "
    @ 8,5 ?? " " + FILL(" ",70) + " "
    @ 9,5 ?? " " + FILL(" ",70) + " "
    @ 10,5 ?? " " + FILL(" ",70) + " "

    STYLE ATTRIBUTE 78
    @ 8,6 ?? FORMAT ("W69,AC","COMPRESSING NRADemo, PLACING NRADATA.ZOO FILES IN
\Maint")
    @ 9,6 ?? FORMAT ("W69,AC"," One moment please...")
    CANVAS ON

    RUN "MAINT\COMPRESS"

    CURSOR OFF
    STYLE ATTRIBUTE 78
    @ 7,6 ?? FORMAT("W69,AC","COMPRESSION COMPLETE, NRADATA.ZOO FILES UNDER \MAINT")
    @ 8,6 ?? FORMAT("W69,AC","(Either make disk from menu option, or creat your own
install.bat")
    @ 9,6 ?? FORMAT("W69,AC","and copy NRADATA.ZOO files to disk.) Press any key to
continue...")
    GC = GETCHAR()

ENDPROC
WRITELIB LIBNAME COMPRESSDEMO
RELEASE PROCS COMPRESSDEMO
;*****
;* END MODULE: x.x COMPRESSDEMO()
;*****/

```

```

;*****
;* PROCEDURE: MAKEDISK3()
;* MODULE:
;* PURPOSE: Writes compressed application to 3.5 inch diskette
;* CALLED BY:
;* CALLS:
;* INPUT ASSERTION:
;* OUTPUT ASSERTION:
;* GLOBAL VARIABLES:
;* AUTHOR: JC Prather
;* DATE: August 1991
;*****/
PROC MAKEDISK3()
  PRIVATE ANS,GC
  ;show pop box
  CANVAS OFF
  PAINTCANVAS ATTRIBUTE BKCOLOR 24,0,24,79
  STYLE ATTRIBUTE HIGHLIGHT

  @ 6,5 ?? " " + FILL(" ",70) + " "
  @ 7,5 ?? " " + FILL(" ",70) + " "
  @ 8,5 ?? " " + FILL(" ",70) + " "
  @ 9,5 ?? " " + FILL(" ",70) + " "
  @ 10,5 ?? " " + FILL(" ",70) + " "

  STYLE ATTRIBUTE 78
  @ 7,6 ?? FORMAT ("W69,AC","Please enter 3 1/2 destination drive for compressed
NRADemo:")
  @ 9,6 ?? FORMAT ("W69,AC"," Format: (Drive name) + \":\" Example: A:")
  CANVAS ON
  CURSOR NORMAL
  @ 8,38
  STYLE ATTRIBUTE 31
  ACCEPT "A2" TO ANS

  IF ISASSIGNED(ANS) = FALSE THEN
    RETURN
  ENDIF

  IF ANS = "" THEN
    RETURN
  ENDIF

  RUN "MAINT\MAKEDISK "+ANS

  CURSOR OFF
  STYLE ATTRIBUTE 78
  @ 7,6 ?? FORMAT("W69,AC","COMPRESSION COMPLETE, Tc install from disk in drive
"+ans+" ---")
  @ 8,6 ?? FORMAT("W69,AC","type INSTALL (source drive) (destination drive), ie:
INSTALL A: C:")
  STYLE ATTRIBUTE HIGHLIGHT
  @ 9,6 ?? FORMAT("W69,AC","Press any key to continue...")
  GC = GETCHAR()
ENDPROC
WRITELIB LIBNAME MAKEDISK3
RELEASE PROCS MAKEDISK3
;*****
;* END MODULE: x.x MAKEDISK3()
;*****/

```

```

;*****
;* PROCEDURE: MAKEDISK5()
;* MODULE:
;* PURPOSE: Writes compressed application to 5.25 inch diskette
;* CALLED BY:
;* CALLS:
;* INPUT ASSERTION:
;* OUTPUT ASSERTION:
;* GLOBAL VARIABLES:
;* AUTHOR: JC Prather
;* DATE: August 1991
;*****/
PROC MAKEDISK5()
    PRIVATE ANS,GC
    ;show pop box
    CANVAS OFF
    PAINTCANVAS ATTRIBUTE BKCOLOR 24,0,24,79
    STYLE ATTRIBUTE HIGHLIGHT

    @ 6,5 ?? " " + FILL(" ",70) + " "
    @ 7,5 ?? " " + FILL(" ",70) + " "
    @ 8,5 ?? " " + FILL(" ",70) + " "
    @ 9,5 ?? " " + FILL(" ",70) + " "
    @ 10,5 ?? " " + FILL(" ",70) + " "

    STYLE ATTRIBUTE 78
    @ 7,6 ?? FORMAT ("W69,AC","Please enter 5 1/4 destination drive for compressed
NRADemo:")
    @ 9,6 ?? FORMAT ("W69,AC"," Format: (Drive name) + \":\" Example: A:")
    CANVAS ON
    CURSOR NORMAL
    @ 8,38
    STYLE ATTRIBUTE 31
    ACCEPT "A2" TO ANS

    IF ISASSIGNED(ANS) = FALSE THEN
        RETURN
    ENDIF

    IF ANS = "" THEN
        RETURN
    ENDIF

    RUN "MAINT\MAKEDISK "+ANS

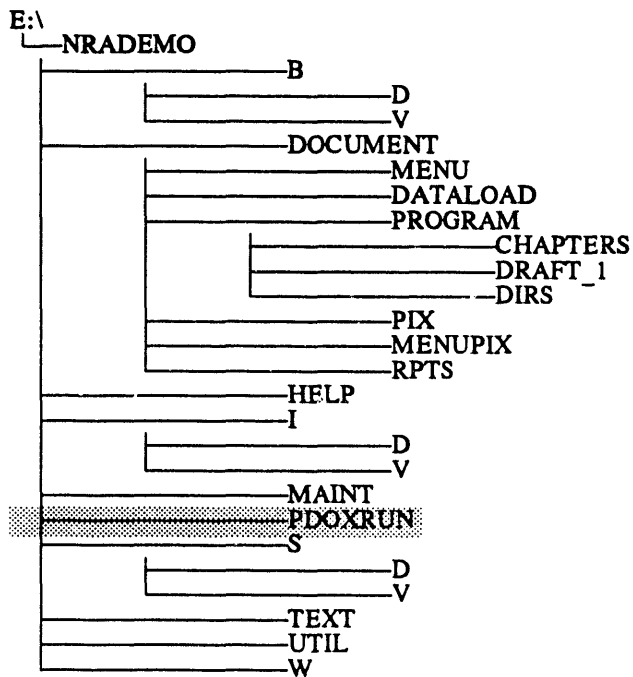
    CURSOR OFF
    STYLE ATTRIBUTE 78
    @ 7,6 ?? FORMAT("W69,AC","COMPRESSION COMPLETE, To install from disk in drive
"+ans+" ---")
    @ 8,6 ?? FORMAT("W69,AC","type INSTALL (source drive) (destination drive), ie:
INSTALL A: C:")
    STYLE ATTRIBUTE HIGHLIGHT
    @ 9,6 ?? FORMAT("W69,AC","Press any key to continue...")
    GC = GETCHAR()
ENDPROC
WRITELIB LIBNAME MAKEDISK5
RELEASE PROCS MAKEDISK5
;*****
;* END MODULE: x.x MAKEDISK5()
;*****/

```

CHAPTER 5 - \PDOXRUN

PDOXRUN directory tree

Subdirectory PDOXRUN contains the runtime executable modules distributed with Paradox®. The files in this subdirectory are static.



Directory of E:\NRADEMO\PDOXRUN

```
Volume in drive E is Jon
Directory of E:\NRADEMO\PDOXRUN

.                <DIR>          8-25-91   3:49p
```

```

..          <DIR>          8-25-91   3:49p
PDOXRUN  AUX   131707      2-25-91   3:51a
PDOXRUN  CFG    2578      10-11-91  10:12a
PDOXRUN  EXE    6856      2-25-91   3:51a
PDOXRUN  MSG   67274      2-25-91   3:51a
PDOXRUN  NTP    1101      2-25-91   3:51a
PDOXRUN  OV1   85717      2-25-91   3:51a
PDOXRUN  OV2   61727      2-25-91   3:51a
PDOXRUN  SOM    1113      6-17-91   1:53p
PDOXRUN  SOR    1637      2-25-91   3:51a
PDOXRUNK EXE  102739      2-25-91   3:51a
PDOXRUNK EXE  985728      2-25-91   3:51a
PDOXRUNT EXE   22606      2-25-91   3:51a
      14 File(s) 18636800 bytes free

```

These files must be present in this subdirectory in order that the stand alone version of NRADEMO will operate. See the Paradox® manuals for more information.

What is Paradox® Runtime?

The following definition is given on page 1 of "Paradox® Relational Database Version 3.5 Runtime Guide", published in 1990 by Borland International, Inc.

Paradox® Runtime is specially designed to allow users to run applications developed with Paradox®. Paradox® Runtime is non-interactive--its users can perform only the operations included in a Paradox® script and not any of the ad hoc interactions included in Paradox® itself. Paradox® Runtime users have access to all Paradox® capabilities and operations that are built into the application they are using.

Paradox® Runtime gives application developers a cost-effective way to distribute applications to users. This can be useful because under the terms of your Paradox® license, you cannot make copies of Paradox® itself, even for the purpose of distributing applications. However, with Paradox® Runtime, you can distribute unlimited copies of your Paradox® application programs with no licensing fees.

Distribution Diskette Labeling Requirement

All distribution diskettes must be labeled with the following copyright notice:

Paradox® Runtime Copyright 1990 Borland International. All Rights Reserved.

Paradox® Runtime Disclaimer

The following disclaimer is required at the start of each application built using Paradox® Runtime.

This software package is running under Paradox® Runtime (Copyright 1990, Borland International, all rights reserved). By pressing a key to continue, you agree to:

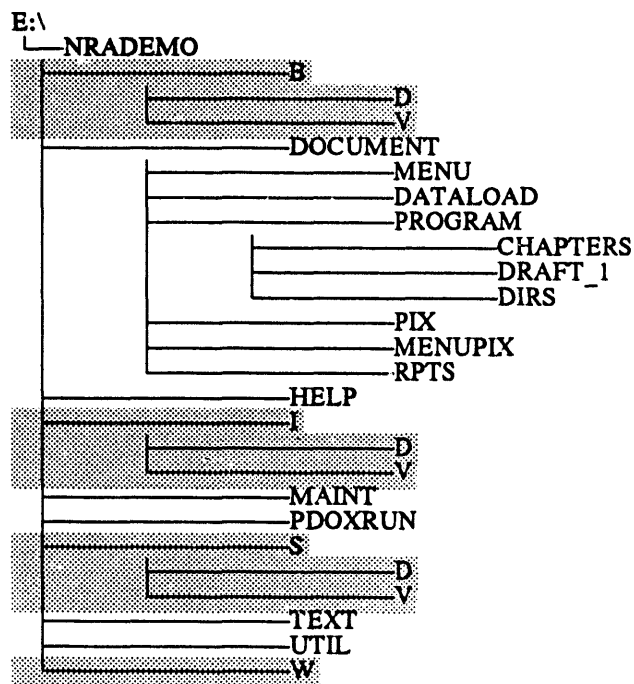
- acknowledge that Paradox® Runtime is owned by Borland International and may not be copied,
- look to PNL, the application developer, and not Borland International, for any support services required by this application,
- take no action against Borland International for any damages resulting from the use of this application, it being understood that Paradox® Runtime is provided by Borland International "AS IS" and without warranties or liabilities for any damages.

Press any key to continue...

CHAPTER 6 - DATABASE DIRECTORIES

Database directory trees

The database directories, **B**ibliography, **I**nventory, and **S**ummary are organized in three nodes. These contain copies of the files in the same nodes in the main NRA database. The file structures were copied from the main NRA databases through the Paradox® {F10} {Tools} {Copy} {JustFamily} option. These database tables are described in "National Radiobiology Archives Database Documentation C. Table Structures".



\NRADEMO\B

The \B subdirectory is the root for the Data and Validation tables associated with the bibliography database. No files are stored in \B.

\NRADEMO\B\D

Subdirectory \NRADEMO\B\D contains the Author and Book tables and associated indexes and report formats.

Volume in drive E is Jon
Directory of E:\NRADEMO\B\D

.	<DIR>	8-25-91	3:48p
..	<DIR>	8-25-91	3:48p
AUTHOR	DB	12288	10-18-91 12:57p
AUTHOR	F	1012	10-10-91 9:51a
AUTHOR	F1	880	10-10-91 9:51a
AUTHOR	F2	1080	10-10-91 9:51a
AUTHOR	F3	1036	10-10-91 9:51a
AUTHOR	F4	1073	10-10-91 9:51a
AUTHOR	PX	4096	10-18-91 12:57p
AUTHOR	R10	2415	10-10-91 9:51a
AUTHOR	VAL	365	10-10-91 9:51a
BOOK	DB	104448	10-18-91 12:57p
BOOK	F	1413	10-09-91 11:36a
BOOK	F1	2291	10-09-91 11:36a
BOOK	F2	1668	10-09-91 11:36a
BOOK	F4	1316	10-09-91 11:36a
BOOK	R1	3490	10-09-91 11:36a
BOOK	R10	3782	10-09-91 11:36a
BOOK	R2	3490	10-09-91 11:36a
BOOK	VAL	473	10-09-91 11:36a
20 File(s) 18759680 bytes free			

\NRADEMO\B\V

Three validation tables, AUTHTYPE, DOCTYPE, and DONOR, (with indexes and reports) are stored in E:\NRADEMO\B\V:

```
Volume in drive E is Jon
Directory of E:\NRADEMO\B\V

.                <DIR>          8-25-91   3:49p
..               <DIR>          8-25-91   3:49p
AUTHTYPE DB      1218      4-24-91   4:20p
AUTHTYPE PX      1197      4-24-91   4:20p
AUTHTYPE R       1941      3-02-91  10:07a
DOCTYPE DB       1218     10-23-90   9:14a
DOCTYPE PX       1197     10-23-90   9:14a
DOCTYPE R        3541      3-02-91  10:07a
DONOR DB          1218     10-23-90   7:33a
DONOR PX          1197     10-23-90   7:33a
DONOR R           1941      3-02-91  10:06a
11 File(s) 18728960 bytes free
```

\NRADEMO\I

The \I subdirectory is the root for the Data and Validation tables associated with the Inventory database. No files are stored in \I.

\NRADEMO\ID

The subdirectory \NRADEMO\ID contains copies of the three Paradox® tables of the Inventory database. MASTER defines each barcode, LOCATION deals with the physical location of the item, and DETAIL relates individual animals (if any) with the item.

```
Volume in drive E is Jon
Directory of E:\NRADEMO\I\ID

.                <DIR>          8-25-91   3:48p
..               <DIR>          8-25-91   3:48p
DETAIL DB        8192  10-18-91  12:57p
DETAIL F         940   7-18-91   6:33p
DETAIL F1        772   10-16-91  10:24a
DETAIL F2       1207   10-16-91  10:24a
DETAIL F7        772   10-16-91  10:24a
DETAIL FX       4096   10-18-91  12:57p
DETAIL R1       4840   10-16-91  10:24a
DETAIL R10      2199   10-16-91  10:24a
DETAIL VAL       161   10-16-91  10:24a
DETAIL X02     10240   10-18-91  12:57p
DETAIL Y02       4096   10-18-91  12:57p
LOCATION DB      18432   10-18-91  12:57p
LOCATION F1      1048   10-09-91  11:12a
LOCATION F2       984   10-09-91  11:12a
LOCATION F3       984   10-09-91  11:12a
LOCATION F7      1048   10-09-91  11:13a
LOCATION FX       4096   10-18-91  12:57p
LOCATION R10     2801   10-09-91  11:13a
LOCATION VAL       269   10-09-91  11:12a
MASTER DB     28672   10-18-91  12:57p
MASTER F      1242   10-09-91  10:54a
MASTER F1     1304   10-09-91  10:54a
MASTER F2     1354   10-09-91  10:54a
MASTER F3     1140   10-09-91  10:54a
MASTER F7     1646   10-09-91  10:54a
MASTER FX     4096   10-18-91  12:57p
MASTER R10    2744   10-09-91  10:54a
MASTER VAL      269   10-09-91  10:54a
30 File(s)  18755584 bytes free
```

\NRADEMO\I\V

Subdirectory \NRADEMO\I\V contains the look up and validation tables associated with the inventory database.

```
Volume in drive E is Jon
Directory of E:\NRADEMO\I\V

.                <DIR>          8-25-91   3:48p
..               <DIR>          8-25-91   3:48p
CONTAIN DB       1218       7-12-91   2:16p
CONTAIN FX       1197       7-12-91   2:16p
CONTAIN R        1841      11-09-90   3:38p
ITEM DB          1218       6-25-91  12:53p
ITEM F1          500        6-07-91   7:26a
ITEM FX          1197       6-25-91  12:53p
ITEM R           1841      11-09-90   3:37p
POS DB           1218       4-15-91  12:33p
POS FX           1197       4-15-91  12:33p
QUAL DB          4096       7-08-91   3:08p
QUAL FX          4096       7-08-91   3:08p
QUAL R           1941       3-03-91   9:46a
14 File(s) 18755584 bytes free
```

\NRADEMO\S

The \S subdirectory is the root for the Data and Validation tables associated with the Summary database. No files are stored in \S.

\NRADEMO\S\D

Subdirectory S\D contains the database tables, forms, and reports which comprise the Summary database system.

Volume in drive E is Jon
Directory of E:\NRADEMO\S\D

	<DIR>	8-25-91	3:48p
..	<DIR>	8-25-91	3:48p
ANI	DB	53248	10-18-91 12:50p
ANI	F	2882	10-09-91 10:19a
ANI	F1	2666	10-09-91 10:19a
ANI	F10	2998	10-09-91 10:19a
ANI	F11	3105	10-09-91 10:19a
ANI	F12	3298	10-09-91 10:19a
ANI	F14	3314	10-09-91 10:19a
ANI	F2	992	10-09-91 10:19a
ANI	PX	4096	10-18-91 12:50p
ANI	R1	6994	10-09-91 10:19a
ANI	R10	4536	10-09-91 10:19a
ANI	R2	5453	10-09-91 10:19a
ANI	R3	3737	10-09-91 10:19a
ANI	R4	229	10-09-91 10:19a
ANI	R5	8080	10-09-91 10:19a
ANI	SET	596	10-09-91 10:19a
ANI	VAL	870	10-09-91 10:19a
CONTROL	DB	4096	10-18-91 12:59p
CONTROL	PX	4096	10-18-91 12:59p
CONTROL	R10	1773	7-16-91 8:26p
CONTROL	VAL	269	7-16-91 8:26p
GRO	DB	104448	10-18-91 12:44p
GRO	F	1580	10-10-91 9:56a
GRO	F1	1184	10-10-91 9:56a
GRO	F10	560	10-10-91 9:56a
GRO	F11	496	10-10-91 9:56a
GRO	F12	492	10-10-91 9:56a
GRO	PX	4096	10-18-91 12:44p
GRO	R1	3018	10-10-91 9:56a
GRO	R10	2831	10-10-91 9:56a
GRO	VAL	377	10-10-91 9:56a
LAB	DB	6144	10-18-91 12:44p
LAB	F	1522	10-09-91 9:41a
LAB	F1	1094	10-09-91 9:41a
LAB	F10	568	10-09-91 9:41a
LAB	F11	492	10-09-91 9:41a
LAB	F12	492	10-09-91 9:41a
LAB	PX	4096	10-18-91 12:44p
LAB	R1	2933	10-09-91 9:41a
LAB	R10	2789	10-09-91 9:41a
STU	DB	45056	10-18-91 12:44p
STU	F	1746	10-09-91 9:39a
STU	F1	1208	10-09-91 9:39a
STU	F10	544	10-09-91 9:39a
STU	F11	496	10-09-91 9:39a
STU	F12	492	10-09-91 9:39a
STU	PX	4096	10-18-91 12:44p
STU	R1	6345	10-09-91 9:39a
STU	R10	4156	10-09-91 9:39a
STU	VAL	701	10-09-91 9:39a
TDOSE	DB	14336	10-18-91 1:02p
TDOSE	F	1876	7-16-91 8:12p
TDOSE	F1	560	7-16-91 8:12p
TDOSE	F2	616	7-16-91 8:12p
TDOSE	F3	2255	7-16-91 8:12p
TDOSE	PX	4096	10-18-91 1:02p

```

TDOSE   R10      2817   7-16-91   8:12p
TDOSE   VAL      593   7-16-91   8:12p
TEFFECT DB     26624  10-18-91  12:57p
TEFFECT F       927  10-09-91  10:43a
TEFFECT F1      768  10-09-91  10:43a
TEFFECT F2      560  10-09-91  10:43a
TEFFECT PX     4096  10-18-91  12:57p
TEFFECT R1     4669  10-09-91  10:43a
TEFFECT R10    2435  10-09-91  10:43a
TEFFECT VAL     377  10-09-91  10:43a
68 File(s) 18606080 bytes free

```

\NRADEMO\S\V

The validation, conversion and lookup tables for the Summary database are stored in subdirectory S\V.

```

Volume in drive E is Jon
Directory of E:\NRADEMO\S\V

.           <DIR>      8-25-91   3:48p
..          <DIR>      8-25-91   3:48p
DISP       DB      4096   7-18-91   6:39p
DISP       F1      492   7-18-91   6:39p
DISP      F12      488   7-19-91  10:33p
DISP       PX     4096   7-18-91   6:39p
DISP       R1     8079   7-18-91   6:39p
DISP      R10     1073   7-18-91   6:39p
DX         DB      4096  10-09-91   4:35p
DX         F1      492  10-09-91   4:35p
DX         PX     4096  10-09-91   4:35p
DX         R1     1775  10-09-91   4:35p
DX        R10     1097  10-09-91   4:35p
SPECIES   DB     1218   7-18-91   6:48p
SPECIES   F1      526   7-18-91   6:48p
SPECIES  F12      492   7-19-91  10:32p
SPECIES   F2      492   7-18-91   6:48p
SPECIES   PX     1197   7-18-91   6:48p
SPECIES   R      1941   7-18-91   6:48p
SPECIES   R1     1779   7-18-91   6:48p
SPECIES  R10     1073   7-18-91   6:48p
UNITS     DB      4096  10-09-91   4:32p
UNITS     PX      4096  10-09-91   4:32p
UNITS     R1     1525  10-09-91   4:32p
UNITS    R10     1073  10-09-91   4:32p
25 File(s) 18575360 bytes free

```

\NRADEMO\W

The WORK directory is used for database tables which are not fully documented. These work tables are part of the specific application and are temporary tables which change with every distribution. In the Distributed Access application, there are five tables in \W.

GETANI is a copy of S\D\ANI which defines the subset of animals to be distributed. It is created by the DATALOAD process if the user chooses F1 or manually enters animal numbers. It is used by the DATALOAD process if the user chooses F3. This file could be created externally through interactive Paradox® and copied to this directory before invoking the DATALOAD process. See the DATALOAD manual for additional details.

GETBOOK is a copy of B\D\BOOK which defines the subset of the bibliography to be distributed. Use of GETBOOK is similar to that of GETANI.

TDOSE1 is a copy of TDOSE with a TRANS field appended so that the name of the tissue category can be shown on the screen. This is to overcome a limitation of the ability of Paradox® to display multiple look-up translations.

TEFFECT1 is a copy of TEFFECT with two TRANS fields appended so that the name of the tissue and morphology categories can be shown on the screen. This is to overcome a limitation of the ability of Paradox® to display multiple look-up translations.

TEXT is a table used for display of the help and clinical text window. The file is converted to a temporary Paradox® table "TEMP", which the script then shows on the screen.

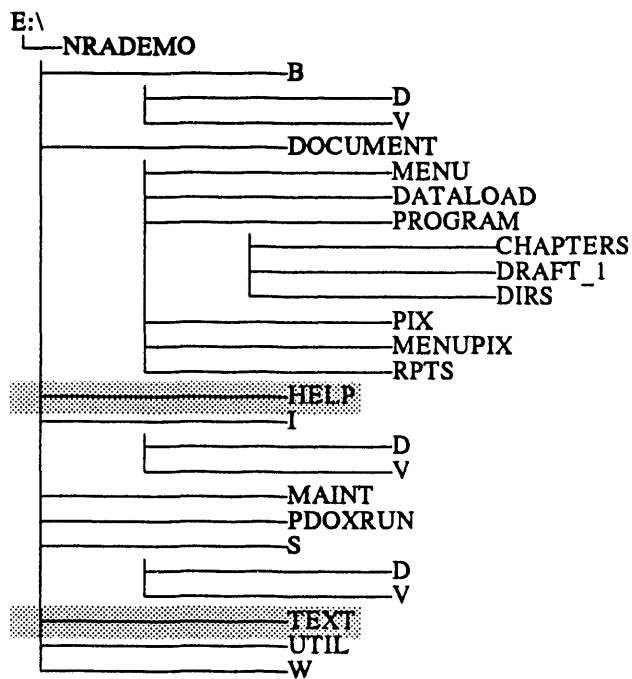
```
Volume in drive E is Jon
Directory of E:\NRADEMO\W

.                <DIR>          8-25-91   3:48p
..               <DIR>          8-25-91   3:48p
CLINCOPY BAT      4330  10-18-91  12:49p
GETANI  DB        53248 10-18-91  12:49p
GETANI  X01        4096 10-18-91  12:51p
GETANI  Y01        4096 10-18-91  12:51p
GETBOOK DB      104448 10-08-91   7:37p
TDOSE1  DB        16384 10-18-91   1:06p
TDOSE1  F12        1038  7-30-91   7:49p
TDOSE1  PX         4096 10-18-91   1:06p
TEFFECT1 DB       90112 10-18-91  12:57p
TEFFECT1 F        1158  8-24-91   2:55a
TEFFECT1 F12       888  8-24-91   9:09a
TEFFECT1 PX       4096 10-18-91  12:57p
TEXT    DB       18432 10-18-91  12:50p
TEXT    F         750  8-24-91   6:24p
TEXT    R1        1589 10-12-91   7:32p
17 File(s) 18544640 bytes free
```

CHAPTER 7 - TEXT FILE DIRECTORIES

HELP and TEXT directory trees

This chapter is about the two subdirectories used to store ASCII text files. Subdirectory \HELP contains the text files associated with the F1 key. Subdirectory \TEXT contains one text file per animal.



\NRADEMO\TEXT

Subdirectory \NRADEMO\TEXT contains one ASCII text file for each animal in \S\D\ANI. These files are retrieved and shown on the screen when the user presses F5 while viewing the animal browse screen.

Eventually, there will be a complete narrative description of each animal. These will be provided by the donating laboratories. If no description is available, the DATALOAD process generates one line temporary files, e.g. A000202.TXT. An example of a clinical summary is U000002.TXT.

```
Volume in drive E is Jon
Directory of E:\NRADEMO\TEXT

.<DIR>          8-25-91   3:49p
..<DIR>          8-25-91   3:49p
A000202  TXT         45   9-13-91   9:43a
A000220  TXT         45   9-13-91   9:43a
A000238  TXT         45   9-13-91   9:43a
A000301  TXT         45   9-13-91   9:43a
A000315  TXT         45   9-13-91   9:43a
A000340  TXT         45   9-13-91   9:43a
A000389  TXT         45   9-13-91   9:43a
A000400  TXT         45   9-13-91   9:43a
A000421  TXT         45   9-13-91   9:43a
DD00F01  TXT         45   9-13-91   9:43a
DD00F02  TXT         45   9-13-91   9:43a
DD00F03  TXT         45   9-13-91   9:43a
DD00F05  TXT         45   9-13-91   9:43a
DD00F08  TXT         45   9-13-91   9:43a
DD00F11  TXT         45   9-13-91   9:43a
DD00F12  TXT         45   9-13-91   9:43a
DD00F16  TXT         45   9-13-91   9:43a
DD00F19  TXT         45   9-13-91   9:43a
DD00F24  TXT         45   9-13-91   9:43a
DD00F29  TXT         45   9-13-91   9:43a
DD00M88  TXT         45   9-13-91   9:43a
DD00X23  TXT         45   9-13-91   9:43a
DR00M87  TXT         45   9-13-91   9:43a
DR00Y41  TXT         45   9-13-91   9:43a
DR00Y45  TXT         45   9-13-91   9:43a
I00002C  TXT         45   9-13-91   9:43a
I00009E  TXT         45   9-13-91   9:43a
I00010C  TXT         45   9-13-91   9:43a
I00012A  TXT         45   9-13-91   9:43a
I00021C  TXT         45   9-13-91   9:43a
I00024E  TXT         45   9-13-91   9:43a
I00032A  TXT         45   9-13-91   9:43a
I00053D  TXT         45   9-13-91   9:43a
I00059C  TXT         45   9-13-91   9:43a
I00121A  TXT         45   9-13-91   9:43a
I00165E  TXT         45   9-13-91   9:43a
P000738  TXT         45   9-13-91   9:43a
P000749  TXT         45   9-13-91   9:43a
P000775  TXT         45   9-13-91   9:43a
P000785  TXT         45   9-13-91   9:43a
P000789  TXT         45   9-13-91   9:43a
P000792  TXT         45   9-13-91   9:43a
P000800  TXT         45   9-13-91   9:43a
P000846  TXT         45   9-13-91   9:43a
P000868  TXT         45   9-13-91   9:43a
P000882  TXT         45   9-13-91   9:43a
P000939  TXT         45   9-13-91   9:43a
P000949  TXT         45   9-13-91   9:43a
P001197  TXT         45   9-13-91   9:43a
R000701  TXT         45   9-13-91   9:43a
R000702  TXT         45   9-13-91   9:43a
R000703  TXT         45   9-13-91   9:43a
R000704  TXT         45   9-13-91   9:43a
R000705  TXT         45   9-13-91   9:43a
R000706  TXT         45   9-13-91   9:43a
R000707  TXT         45   9-13-91   9:43a
R000708  TXT         45   9-13-91   9:43a
R000709  TXT         45   9-13-91   9:43a
R000710  TXT         45   9-13-91   9:43a
```

R000711	TXT	45	9-13-91	9:43a
R000712	TXT	45	9-13-91	9:43a
R000713	TXT	45	9-13-91	9:43a
R000714	TXT	45	9-13-91	9:43a
R000715	TXT	45	9-13-91	9:43a
R000716	TXT	45	9-13-91	9:43a
R000717	TXT	45	9-13-91	9:43a
R000718	TXT	45	9-13-91	9:43a
R000719	TXT	45	9-13-91	9:43a
R000720	TXT	45	9-13-91	9:43a
R000721	TXT	45	9-13-91	9:43a
R000722	TXT	45	9-13-91	9:43a
R000723	TXT	45	9-13-91	9:43a
R000724	TXT	45	9-13-91	9:43a
R000725	TXT	45	9-13-91	9:43a
R000726	TXT	45	9-13-91	9:43a
R000727	TXT	45	9-13-91	9:43a
R000728	TXT	45	9-13-91	9:43a
R000729	TXT	45	9-13-91	9:43a
R000730	TXT	45	9-13-91	9:43a
R000731	TXT	45	9-13-91	9:43a
R000732	TXT	45	9-13-91	9:43a
R000733	TXT	45	9-13-91	9:43a
R000734	TXT	45	9-13-91	9:43a
R000735	TXT	45	9-13-91	9:43a
U000002	TXT	1096	11-10-90	12:03p
U000019	TXT	45	9-13-91	9:43a
U000027	TXT	2520	11-10-90	12:05p
U000030	TXT	2355	11-10-90	10:54a
U000076	TXT	45	9-13-91	9:43a
U000116	TXT	45	9-13-91	9:43a
U000138	TXT	45	9-13-91	9:43a
U000145	TXT	45	9-13-91	9:43a
U000246	TXT	45	9-13-91	9:43a
U000312	TXT	45	9-13-91	9:43a
U000386	TXT	45	9-13-91	9:43a
U000441	TXT	2772	11-10-90	8:52a
U000508	TXT	2295	11-10-90	8:58a
U001639	TXT	45	9-13-91	9:43a
U001640	TXT	45	9-13-91	9:43a
U002085	TXT	45	9-13-91	9:43a

102 File(s) 18483200 bytes free

\NRADEMO\HELP

Subdirectory \NRADEMO\HELP contains two types of ASCII text files. Those with the name extension "HLP" are field related help which is shown when the user presses F1 while viewing any of the browse screens. Those files with the name extension, "TXT", are shown when the user presses F1 while viewing a menu.

Extreme care should be used when editing these files because of line length considerations. The text is word wrapped by the Distributed Access software when it is shown on the screen or printed from the browse screen. These files also serve as subdocuments to the User's Guide, without word wrap.

Volume in drive E is Jon
Directory of E:\NRADEMO\HELP

```

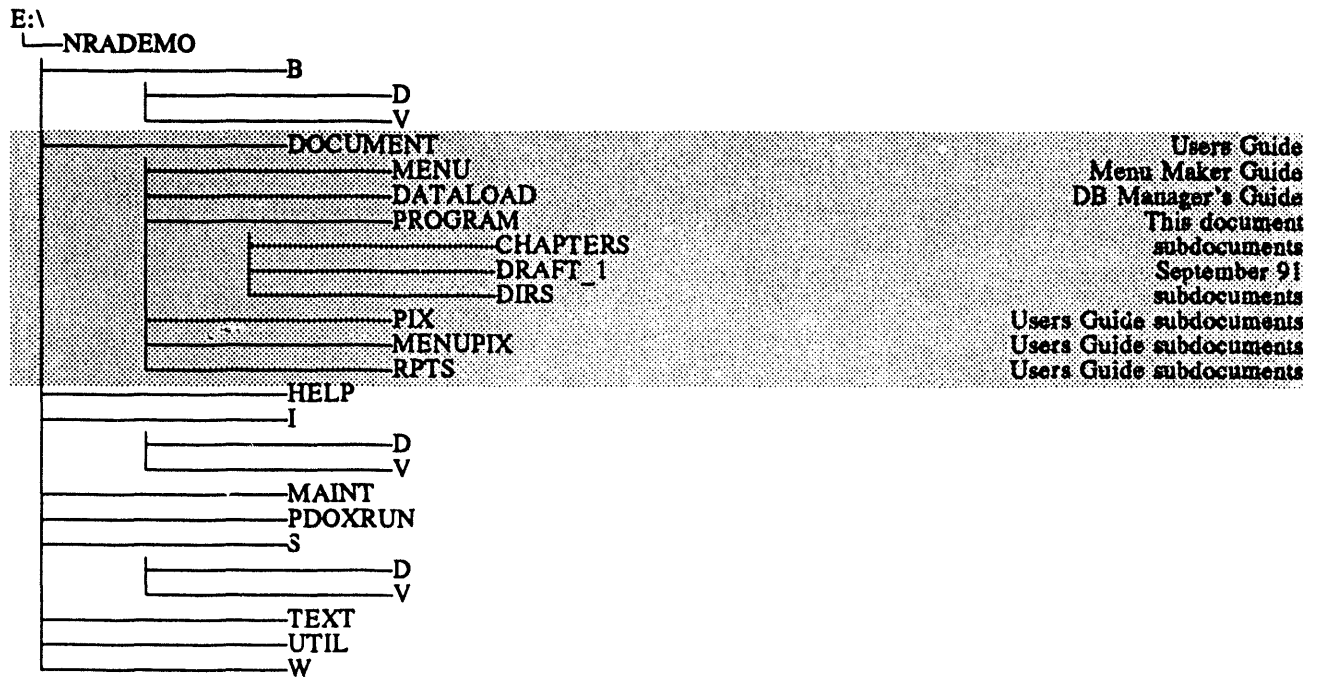
      <DIR>          9-13-91 11:32a
      <DIR>          9-13-91 11:32a
ACCESSIO HLP      377 10-15-91 12:09p
ANIMAL I HLP     1788 10-15-91 12:19p
ANI_QTY HLP      158 10-15-91 11:06a
ASSIGNED HLP     1850 10-14-91 10:04a
BAR CODE HLP      801 10-15-91 12:29p
BIBLIOGR TXT     1509 10-16-91 12:29p
BIRTH DA HLP      334 10-14-91 9:50a
BUILDING HLP      157 10-14-91 9:43a
BURDEN T HLP      577 10-14-91 9:34a
CABINET HLP      196 10-11-91 5:24p
CAUSE OF HLP     2587 10-15-91 12:26p
COMMENTS HLP      177 10-11-91 5:25p
CONTAIN HLP      495 10-11-91 5:27p
CONTROL HLP      402 10-11-91 5:20p
CURSOR TXT     2604 10-15-91 10:49a
DAM ID HLP       601 10-15-91 12:33p
DEATH DA HLP      926 10-15-91 12:45p
DETAIL TXT     3047 10-16-91 12:30p
DISPOSIT HLP      583 10-11-91 5:13p
DOC TYPE HLP      381 10-11-91 5:12p
DONOR_ID HLP      381 10-11-91 5:11p
DX HLP          1222 10-11-91 4:26p
FINAL IN HLP      510 10-11-91 5:09p
FINDINGS HLP     1042 10-11-91 4:53p
FIRST IN HLP      639 10-11-91 4:49p
FIRST_NA HLP      119 10-11-91 4:46p
FUNCTION TXT     1477 10-16-91 8:34a
GROUP ID HLP      774 10-11-91 4:43p
INSULT U HLP      971 10-16-91 8:16a
INVENTOR TXT     2977 10-16-91 12:47p
ITEM HLP         793 10-11-91 4:35p
ITEM COD HLP      796 10-02-91 5:24p
ITM_QTY HLP      162 10-15-91 11:07a
JOURNAL HLP      138 10-11-91 4:32p
LAB HLP         196 10-11-91 4:32p
LAB ID HLP       671 10-11-91 4:30p
LAST NAM HLP      117 10-11-91 4:28p
LITTER N HLP      232 10-11-91 4:28p
MIDDLE HLP        117 10-11-91 4:26p
MORE CAT HLP     1222 10-15-91 12:50p
NOMELP HLP        107 10-11-91 4:22p
NRA TXT         4266 10-16-91 12:58p
PAGES HLP        148 10-11-91 4:22p
POS HLP          166 10-11-91 4:21p
PRIMARY HLP       791 10-15-91 12:50p
PUBLISHE HLP     165 10-11-91 4:16p
PUB YEAR HLP     126 10-11-91 4:15p
PURPOSE TXT     1135 10-16-91 8:22a
QTY INSU HLP     219 10-11-91 4:14p
QUALIFIE HLP     277 10-15-91 12:52p
QUANTITY HLP     251 10-11-91 4:08p
RANK HLP         164 10-15-91 12:53p
REMOVAL HLP      455 10-02-91 4:50p
REMOVAL HLP      470 10-15-91 12:54p
REPORT N HLP      270 10-11-91 3:50p
RESULTS HLP      303 10-11-91 3:47p
ROOM HLP         142 10-11-91 3:46p
SEX HLP          283 10-11-91 3:45p
SHELF HLP        248 10-11-91 3:44p
SIRE ID HLP      605 10-15-91 12:57p
SNODOG HLP       908 10-16-91 10:42a
SPECIES HLP      672 10-15-91 1:01p
STATUS HLP       230 10-11-91 1:15p
STUDY ID HLP     585 10-11-91 1:12p
SUBTITLE HLP     147 10-11-91 1:06p
SUMMARY TXT     3757 10-14-91 1:25p
TISSUE C HLP     933 10-11-91 1:05p
TISSUE T HLP     933 10-11-91 1:05p
TISS_QUA HLP     722 10-15-91 1:03p
TITLE HLP        141 10-11-91 12:55p
VOLUME HLP       156 10-11-91 12:57p
WGT_FIRS HLP     214 10-11-91 12:56p
74 File(s) 18513920 bytes free

```

CHAPTER 8 - \DOCUMENT

DOCUMENT directory tree

This document is stored in the subdirectories associated with NRADEMO\DOCUMENT. The document was written in WordPerfect® 5.1. Both the WordPerfect® version, *.W51, and an ASCII text file, *.DOC, are provided.



Users Guide
Menu Maker Guide
DB Manager's Guide
This document
subdocuments
September 91
subdocuments
Users Guide subdocuments
Users Guide subdocuments
Users Guide subdocuments

Document File Naming Convention

<i>name</i>	<i>extension</i>	<i>conventional meaning</i>
*	W51	WordPerfect® V5.1 input file (usually a subdocument)
*	DIR	ASCII file produced by DOS DIR command, used as a subdocument by PROG_WP.W51
*	RPT	ASCII file produced by Paradox®. These examples of the use of Distributed Access F6 option are used as subdocuments by USER_WP.W51
*	TXT	ASCII text file (subdocument)
*	DOC	ASCII version of document (w/o fonts or pagination). This is the file to print from DOS.
*_WP	W51	Condensed master document e.g. PROG_WP.W51 is the WordPerfect® master document for this publication.
*_MAN	W51	Expanded master document, e.g. PROG_MAN.W51 is the WordPerfect® expansion of PROG_WP.W51 and all its associated subdocuments. This is the file to print from WordPerfect®.

\DOCUMENT

The subdirectory NRADEMO\DOCUMENT contains the User's Guide, USER_WP.W51, and the associated subdocuments, CHAP_H2.W51, CHAP_IS1.W51, *etc.*

```
Volume in drive E is Jon
Directory of E:\NRADEMO\DOCUMENT

.                <DIR>          8-25-91    3:49p
..               <DIR>          8-25-91    3:49p
DATALOAD         <DIR>        10-18-91    9:25a
MENU             <DIR>        10-18-91    9:22a
MENUPIX         <DIR>        10-14-91    1:21p
PIX             <DIR>        10-01-91   10:52a
PROGRAM         <DIR>        9-30-91   12:17p
RPTS            <DIR>        10-09-91    1:39p
CHAP_H1.W51      22698  10-16-91    1:51p
CHAP_H2.W51      54668  10-16-91    2:35p
CHAP_IS1.W51     37455  10-16-91    1:55p
CHAP_IS2.W51     27362  10-16-91    2:04p
CHAP_M1.W51     26866  10-17-91   12:26p
CHAP_P1.W51     25280  10-16-91    2:00p
LICENSE.DOC      6622  10-17-91   12:28p
LICENSE.TXT      16741  10-16-91    1:00p
RUNTIME.TXT       833  10-16-91   12:45p
USER_MAN.W51     335855  10-17-91    8:49a
USER_WP.W51      101515  10-17-91    8:56a
               19 File(s) 19124224 bytes free
```

\DOCUMENT\MENUPIX

Subdirectory NRADEMO\DOCUMENT\MENUPIX contains screen capture images of the menus. The screens were captured with the Demo II software, then converted to ASCII text files through it's Print option.

```
Volume in drive E is Jon
Directory of E:\NRADEMO\DOCUMENT\MENUPIX

.                <DIR>        10-14-91    1:21p
..               <DIR>        10-14-91    1:21p
A_SEAR_1.TXT     1555  10-16-91    9:07a
A_SEAR_2.TXT     1558  10-16-91    9:08a
A_SEAR_3.TXT     1559  10-16-91    9:09a
A_SEAR_4.TXT     1559  10-16-91    9:10a
BIB.TXT          1156  10-15-91    1:55p
INVENTOR.TXT     1156  10-15-91    1:56p
MAIN.TXT         1140  10-15-91    1:50p
OCT_14.DBD      10922  10-14-91    1:31p
OCT_15.DBD       2871  10-15-91    1:46p
SUMMARY.TXT      1156  10-15-91    1:57p
TITLE.TXT        2050  10-15-91    1:45p
               13 File(s) 18235392 bytes free
```

\DOCUMENT\PIX

Subdirectory NRADEMO\DOCUMENT\PIX contains screen capture images of the search and print options of NRA Distributed Access.

```
Volume in drive E is Jon
Directory of E:\NRADEMO\DOCUMENT\PIX

.<DIR>          10-01-91  10:52a
..<DIR>          10-01-91  10:52a
LAB      RPT      839  10-16-91  8:52a
P_A_1    TXT    10677  10-16-91  2:22p
P_A_2    TXT    1393  10-16-91  8:56a
P_A_3    TXT    1427  10-16-91  8:56a
P_A_4    TXT    1410  10-16-91  8:57a
P_A_5    TXT    1407  10-16-91  8:58a
P_A_RPT  TXT    3707  10-16-91  9:41a
P_B_1    TXT    3802  10-16-91  2:19p
P_B_3    TXT    1202  10-14-91  10:24a
P_B_RPT  TXT    1387  10-07-91  2:02p
P_I_1    TXT    4621  10-16-91  2:19p
P_I_2    TXT    1534  10-16-91  9:14a
P_I_3    TXT    1553  10-14-91  10:26a
P_I_RPT  TXT    1833  10-16-91  9:15a
16 File(s) 18759680 bytes free
```

\DOCUMENT\RPTS

Subdirectory NRADEMO\DOCUMENT\RPTS contains example report files produced by the F6 option of NRA Distributed Access.

```
Volume in drive E is Jon
Directory of E:\NRADEMO\DOCUMENT\RPTS

.<DIR>          10-09-91  1:39p
..<DIR>          10-09-91  1:39p
ANI      RPT      6064  10-09-91  2:43p
AUTHOR   RPT      490  10-09-91  2:44p
BOOK     RPT     1464  10-17-91  9:10a
DETAIL   RPT      812  10-09-91  2:45p
DOCTYPE  RPT     2245  10-09-91  1:53p
GRO      RPT      709  10-09-91  2:45p
LAB      RPT      942  10-09-91  9:12a
LAB      RPT      706  10-17-91  7:48a
LOCATION   RPT     1272  10-16-91  1:56p
MASTER   RPT     2053  10-09-91  2:46p
NEWLAB   RPT    37962  10-17-91  7:46a
STU      RPT     2541  10-09-91  2:47p
TDOSE    RPT     2316  10-09-91  4:38p
TEFFECT  RPT     2045  10-17-91  9:09a
16 File(s) 18755584 bytes free
```

\DOCUMENT\MENU

Subdirectory NRADEMO\DOCUMENT\MENU contains an internal users's guide to the NRA MAKEMENU application generator.

```
Volume in drive E is Jon
Directory of E:\NRADEMO\DOCUMENT\MENU

.                <DIR>          10-18-91   9:22a
..               <DIR>          10-18-91   9:22a
MENUDOC  W51      82889    8-28-91  10:50a
          3 File(s) 18470912 bytes free
```

\DOCUMENT\DATALOAD

Subdirectory NRADEMO\DOCUMENT\DATALOAD contains an internal users's guide to the data maintenance menus for creating a new distributed subset.

```
Volume in drive E is Jon
Directory of E:\NRADEMO\DOCUMENT\DATALOAD

.                <DIR>          10-18-91   9:25a
..               <DIR>          10-18-91   9:25a
DATALOAD W51      2046    10-10-91   9:01a
GETANI  DB        53248   10-08-91   6:53p
GETANI  X01       4096    10-08-91   7:20p
GETANI  Y01       4096    10-08-91   7:20p
GETBOOK DB       104448   10-08-91   7:37p
          7 File(s) 18468864 bytes free
```

\DOCUMENT\PROGRAM

Subdirectory NRADEMO\DOCUMENT\PROGRAM contains this publication, PROG_WP.W51.

```
Volume in drive E is Jon
Directory of E:\NRADEMO\DOCUMENT\PROGRAM

.                <DIR>          9-30-91  12:17p
..               <DIR>          9-30-91  12:17p
CHAPTERS        <DIR>         10-21-91   7:46a
DIRS            <DIR>         10-18-91   9:12a
DRAFT 1         <DIR>         10-18-91   9:21a
PROG_WP  W51      56129   10-23-91   8:20a
          6 File(s) 19124224 bytes free
```

\DOCUMENT\PROGRAM\CHAPTERS

Subdirectory NRADEMO\DOCUMENT\PROGRAM\CHAPTERS contains the subdocuments of PROG_WP.W51.

```
Volume in drive E is Jon
Directory of E:\NRADEMO\DOCUMENT\PROGRAM\CHAPTERS

.                <DIR>          10-21-91   7:46a
..               <DIR>          10-21-91   7:46a
CH 1            W51      41993   10-22-91   8:23p
CH 2            W51     16687   10-22-91   9:25a
CH 3A           W51     29278   10-22-91   8:56p
CH 3B           W51    210725   10-22-91   8:02p
CH 3C           W51     82704   10-21-91  10:39a
CH 4A           W51     15826   10-22-91  11:16a
CH 4B           W51     69294   10-21-91  10:40a
CH 5            W51     15095   10-22-91   1:12p
CH 6            W51     27427   10-22-91   8:51p
CH 7            W51     12513   10-22-91   1:33p
CH 8            W51     17824   10-23-91  12:17p
MENUCHOI RPT    2366   10-22-91  10:20a
MENULEVE RPT     746   10-22-91  10:00a
15 File(s) 19124224 bytes free
```

\DOCUMENT\PROGRAM\DRAFT_1

Subdirectory NRADEMO\DOCUMENT\DRAFT_1 contains an internal version of this publication.

```
Volume in drive E is Jon
Directory of E:\NRADEMO\DOCUMENT\PROGRAM\DRAFT_1

.                <DIR>          10-18-91   9:21a
..               <DIR>          10-18-91   9:21a
DEMOPROG W51    296931   9-19-91   8:04a
3 File(s) 18454528 bytes free
```

\DOCUMENT\PROGRAM\DIRS

Subdirectory NRADEMO\DOCUMENT\PROGRAM\DIRS contains ASCII files produced by the DOS DIR command. These serve as subdocuments to PROG_WP.W51.

```
Volume in drive E is Jon
Directory of E:\NRADEMO\DOCUMENT\PROGRAM\DIRS

.                <DIR>          10-18-91   9:12a
..               <DIR>          10-18-91   9:12a
BD              DIR           930  10-22-91   3:00p
BV              DIR           561  10-22-91   2:59p
DOCDATA        DIR           411  10-22-91   2:31p
DOCMENU        DIR           243  10-22-91   2:31p
DOCMPTX        DIR           656  10-23-91  12:20p
DOCFCHAP       DIR           747  10-23-91  12:37p
DOCFDIRS       DIR            0  10-23-91  12:38p
DOCFDRAP       DIR           254  10-22-91   2:44p
DOCFIX         DIR           775  10-22-91   3:08p
DOCFPROC       DIR           369  10-23-91  12:34p
DOCFPTS        DIR           776  10-22-91   2:54p
DOCFPTS        DIR           776  10-22-91   3:09p
DOCUMENT       DIR           894  10-23-91  12:32p
HELP           DIR          3145  10-22-91   2:48p
ID             DIR          1340  10-22-91   3:10p
IV             DIR           684  10-22-91   3:10p
MAINT          DIR          1834  10-22-91   2:52p
NRADEMO        DIR           639  10-22-91   2:52p
PDOXRUN        DIR           688  10-22-91   2:51p
SD             DIR          2898  10-22-91   2:51p
SV             DIR          1135  10-22-91   2:51p
TEXT          DIR          4293  10-22-91   2:45p
UTIL          DIR          1218  10-22-91   3:07p
W              DIR           805  10-22-91   2:50p
26 File(s)  19124224 bytes free
```

DISTRIBUTION

No. of
Copies

No. of
Copies

OFFSITE:

2 DOE/Scientific and Technical
 Information

 David A. Smith, Ph.D.
 Acting Director of HERD
 OHER-ER72
 U.S. Department of Energy (GTN)
 Washington, D.C. 20545

 Marvin E. Frazier, Ph.D.
 U.S. Department of Energy (GTN)
 Office of Health and Environmental
 Research
 ER-72
 Washington, DC 20545

 Michael E. Ginevan, Ph.D.
 Deputy Director
 Office of Epidemiology and Health
 Surveillance
 US Department of Energy
 Office of Health, EH-42 (GTN)
 Washington, D.C. 20545

 Otto G. Raabe, Ph.D.
 Laboratory for Energy Related
 Health Research
 University of California at Davis
 Old Davis Road
 Davis, CA 95616

Stephen A. Benjamin, Ph.D.
Director, CRHL
Foothills Campus
Colorado State University
Fort Collins, CO 80523

Scott C. Miller, Ph.D.
Department of Radiobiology
University of Utah
Salt Lake City, UT 84112

Tom E. Fritz, Ph.D.
Argonne National Laboratory
9700 South Cass Avenue
Argonne, IL 60439

Mr. Don E. Doyle
Argonne National Laboratory
9700 South Cass Avenue
Argonne, IL 60439

Bruce B. Boecker, Ph.D.
Inhalation Toxicology Research
Institute
P.O. Box 5890
Albuquerque, NM 87185

Mr. Jonathan C. Prather
Linfield College
Linfield Box 3094 Room 303
McMinnville, OR 97128-6894

Ms. Anna M. Martinez
Northern Arizona University
700 South Blackbird RST Apt. #71
Flagstaff, AZ 86001

No. of
Copies

ONSITE:

DOE Richland Field Office

PW Kruger

Hanford Environmental Health
Foundation

R Filipy

13 Pacific Northwest Laboratory

JF Park

SK Smith (5)

CR Watson

Publication Coordination

Technical Report Files (5)

END

**DATE
FILMED**

3 / 27 / 92

