



NEAR EAST UNIVERSITY



1988

**FACULTY OF ENGINEERING
B.Sc. COMPUTER ENGINEERING**

FINAL YEAR PROJECT

**TITLE: BOOKSTORE SYSTEM
USING FOXPRO
PROGRAMMING**

**ST-NAME:
HISHAM I.S ZAYED**

SUPERVISOR: Ms. BESIME ERIN

JUNE, 2000

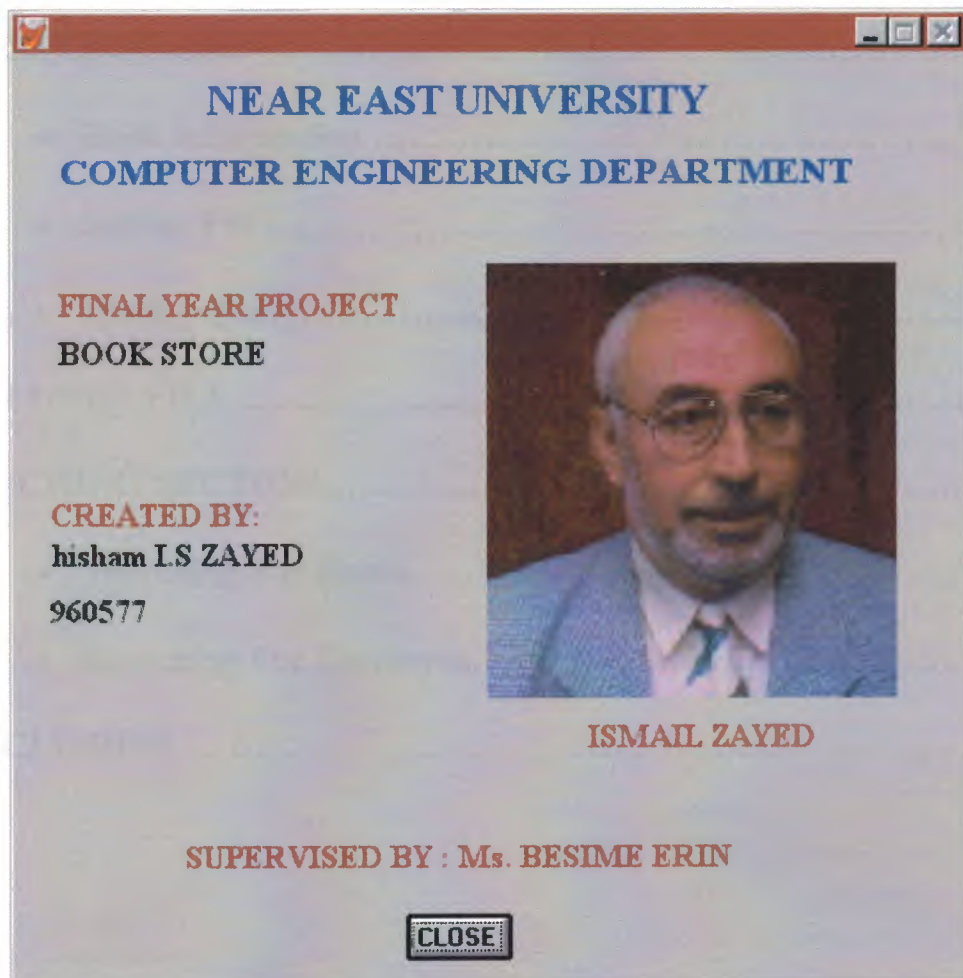




TABLE OF CONTENTS

INTRODUCTION	1
MAIN MENU	2
BOOKS SECTIONS	5
• Book Information	8
• Selling File	27
• Listing Query.....	46
EMPLOYEE FILE	47
SEARCHING SECTION.....	66
• Searching For Books.....	69
• Searching For Employee.....	76
CONCLUSION	82

AKNOWLEDGEMENT

All my thanks to those who endlessly encouraged me through out my undergraduate studies and provided me with all kind of supports financially or morality. Who stands beside me at every moment of my life, My Parents? Feeling proud to dedicate this project for them together with my respectful brothers ***Ramzi, Hamdi, Raid and Khaled.***

I would like to express my faithful thanks to my university with all its educational staff and my instructors who have been a good guide for me, specially my supervisor ***Miss. Besime Erin*** who provided me with Valuable advises and help to achieve my graduation project besides being an example of the responsible teacher.

Thanks to all my friends and school mates who joined me and shared my work sportingly.

Special thanks to my uncle ***Dr. Ibrahim ZAYED.***

ABSTRACT

As the 'information age' has affected every aspect of our life, the sake for computerizing many information systems became an important need. Consequently, one of the major branches that are contributed efficiently by information revolution is the 'database management systems'.

This project is concerned of using computer program in bookstore management system. It composed and written using FoxPro programming language, which is considered as one of, the best and strongest database languages.

This project is a complete bookstore management program, which covers all the services necessary for the bookstores, such as, costumers related information, units, employees, books arrangement and classifying.

INTRODUCTION

Now a day's, the computer science both hardware and software is being developed over the previous years, programming is always providing the scientists by a systematic development, in my project I did construct special programmed related to book store system. By using a computers every things become easier than before and it developed the humans in all its range, now you can do what ever you want in a short period. I'm advising every body to increase his knowledge about the computer.

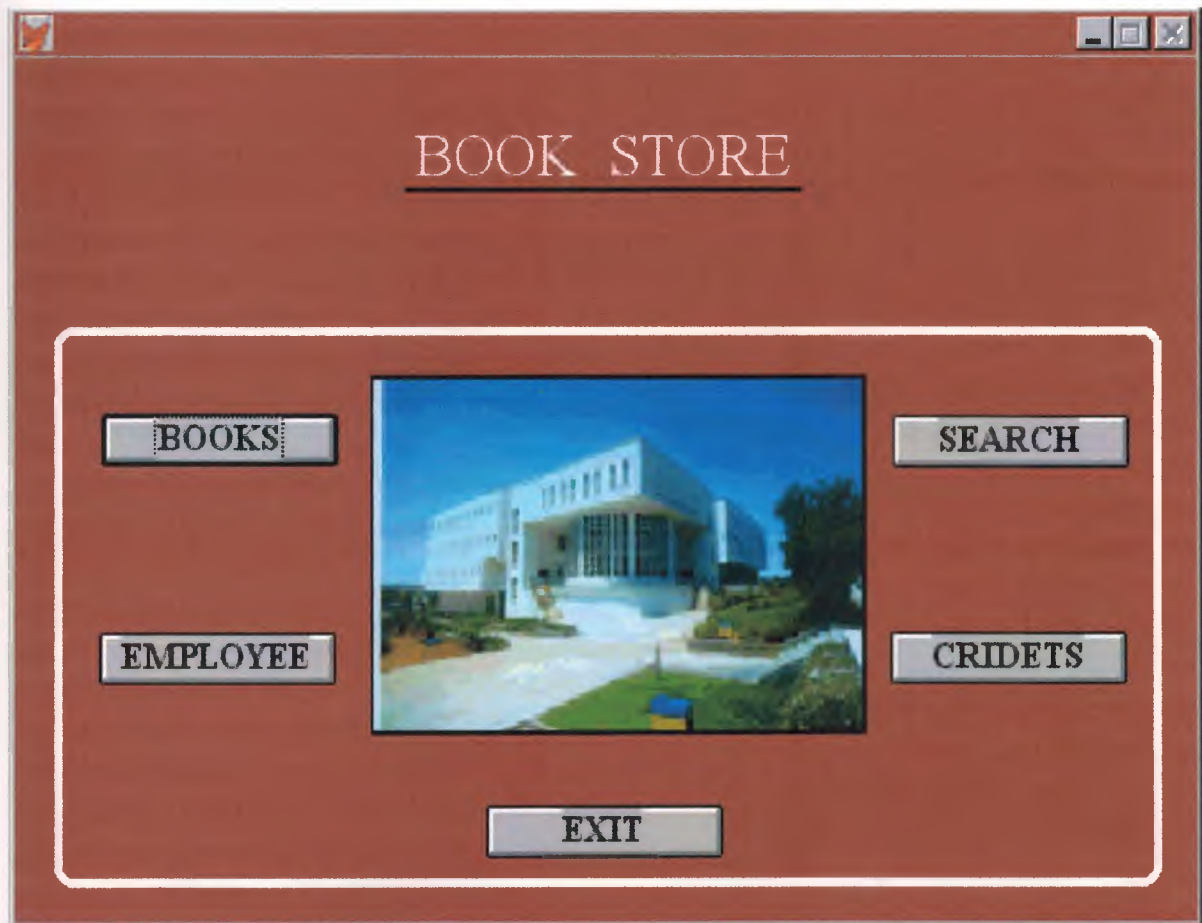
The aim of this project was to create a program that could be used for automation at circulation desk of any library. The reason behind this is to make the operation accurate, organized and faster which result in saving time and efforts.

The steps that I have followed in achieving these goals are summarized below:

First of all, I went to a bookstore and I met those who are in charge and I asked some questions about the system used in the bookstore. Such as, how to add a new books and how the system used in searching for the books performed individually book by book shelves or is it one of the computer's. By obtaining the answers for these questions I managed to built up an idea about how the operation runs over there, as a result I translated the operation into a software programming.

MAIN MENU

The aim of the main menu is to use the program easily, faster and use all the process screens or necessary programs in the same time. In order to use this menu, from the program menu select the do option and choose the *menu.prg* and run the program , after running the program we will see the menu as :



```

SET TALK OFF
IF NOT WEXIST("_0471f8w49")
    DEFINE WINDOW _0471f8w49 ;
        AT 0.000, 0.000 ;
        SIZE 15.750,67.875 ;
        FONT "Times New Roman", 16 ;
        FLOAT ;
        NOCLOSE ;
        MINIMIZE ;
        SYSTEM ;
        COLOR RGB(,,,128,0,0)
        MOVE WINDOW _0471f8w49 CENTER
ENDIF

IF WVISIBLE("_0471f8w49")
    ACTIVATE WINDOW _0471f8w49 SAME
ELSE
    ACTIVATE WINDOW _0471f8w49 NOSHOW
ENDIF

@ 6.500,5.000 GET A ;
    PICTURE "@*HN BOOKS" ;
    SIZE 1.150,15.571,0.571 ;
    DEFAULT 1 ;
    FONT "Times New Roman", 12 ;
    STYLE "B";
    VALID BT('PRO1')

@ 10.500,4.875 GET A ;
    PICTURE "@*HN EMPLOYEE" ;
    SIZE 1.150,15.571,0.571 ;
    DEFAULT 1 ;
    FONT "Times New Roman", 12 ;
    STYLE "B";
    VALID BT('PRO2')

@ 6.542,50.500 GET A ;
    PICTURE "@*HN SEARCH" ;
    SIZE 1.150,15.571,0.571 ;
    DEFAULT 1 ;
    FONT "Times New Roman", 12 ;
    STYLE "B";
    VALID BT('PRO3')

@ 10.500,50.375 GET A ;
    PICTURE "@*HN CRIDETS" ;
    SIZE 1.150,15.571,0.571 ;
    DEFAULT 1 ;
    FONT "Times New Roman", 12 ;
    STYLE "B";

```



```

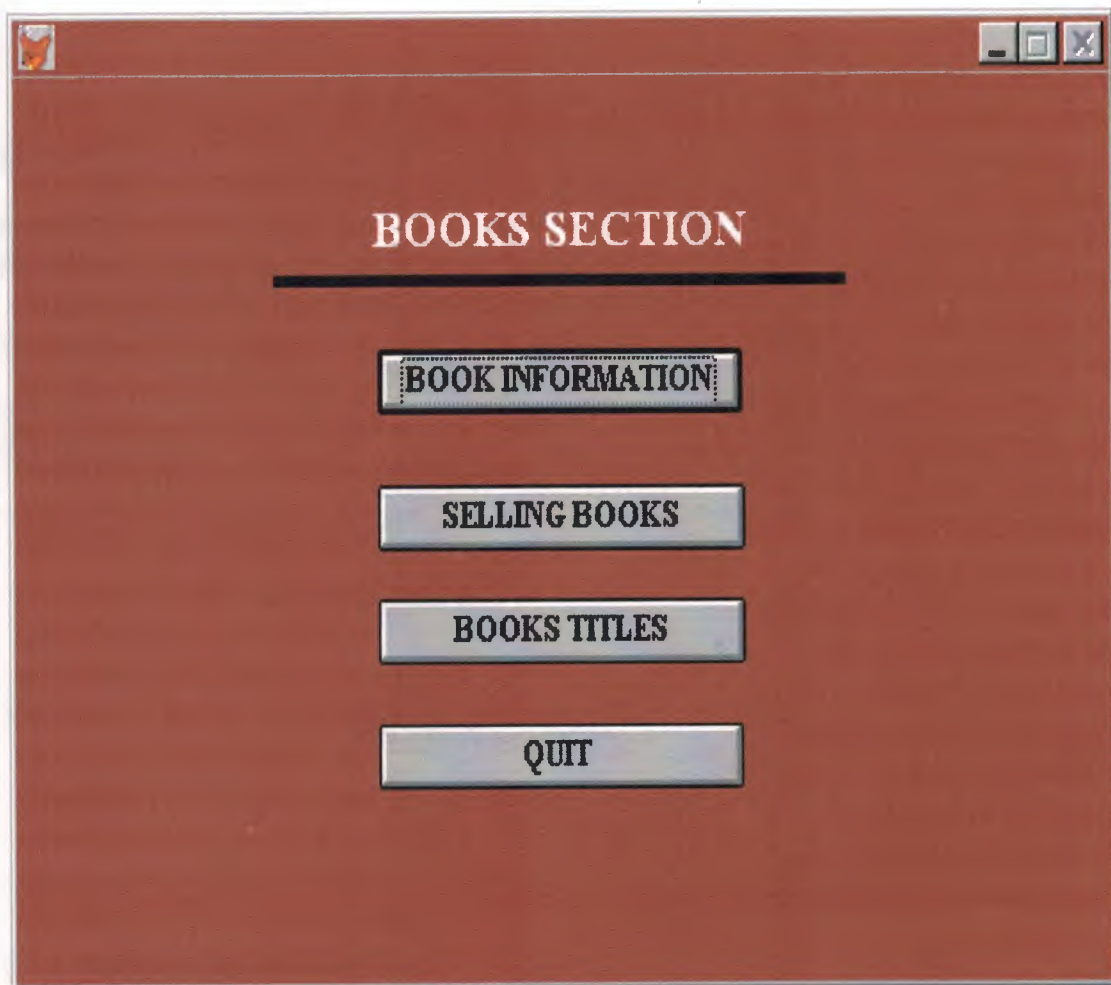
        VALID BT('PRO4')
@ 13.667,27.125 GET M.BT_EXIT ;
    PICTURE "@*HN EXIT" ;
    SIZE 1.150,15.571,0.571 ;
    DEFAULT 1 ;
    FONT "Times New Roman", 12 ;
    STYLE "B" ;
    VALID BT('EXIT')
@ 1.167,23.000 SAY "BOOK STORE" ;
    FONT "Times New Roman", 20 ;
    STYLE "T" ;
    COLOR RGB(255,255,255,,,)
@ 5.792,20.500 SAY (LOCFILE("C:\RAMIZ\2.bmp","BMP|ICO|PCT|ICN", "Where is
2?" )) BITMAP ;
    SIZE 6.625,28.500 ;
    STYLE "T"
@ 2.375,22.500 TO 2.375,45.250 ;
    PEN 2, 8 ;
    STYLE "1" ;
    COLOR RGB(0,0,0,0,0,0)
@ 4.917,2.250 TO 15.167,66.000 ;
    PEN 4, 8 ;
    STYLE "16" ;
    COLOR RGB(255,255,255,,,)

IF NOT WVISIBLE("_0471f8w49")
    ACTIVATE WINDOW _0471f8w49
ENDIF
READ CYCLE
PROCEDURE BT
    PARAMETER m.btnname
    DO CASE
CASE m.btnname='PRO1'
    Do c:\ramiz\book1.spr
CASE m.btnname='PRO2'
    DO c:\ramiz\employee.spr
CASE m.btnname='PRO3'
    DO c:\ramiz\search.spr
CASE m.btnname='PRO4'
    DO c:\ramiz\cridets.spr
CASE m.btnname='EXIT'
set sysm to defa
DEACTIVATE WINDOW _0471f8w49
RELEASE WINDOW _0471f8w49
    ENDCASE
RETURN

```

BOOKS SECTION

Book section menu is used to give us the books information, the selling books and books title, which give us the list of all books that we have in our bookstore.



```

SET TALK OFF
IF NOT WEXIST("_03yle7gxy")
    DEFINE WINDOW _03yle7gxy ;
        AT 0.000, 0.000 ;
        SIZE 15.900,77.833 ;
        FONT "Times New Roman", 12 ;
        FLOAT ;
        CLOSE ;
        MINIMIZE ;
        SYSTEM ;
        COLOR RGB(,,128,0,0)
        move window _03yle7gxy center
ENDIF

#REGION 1
IF WVISIBLE("_03yle7gxy")
    ACTIVATE WINDOW _03yle7gxy SAME
ELSE
    ACTIVATE WINDOW _03yle7gxy NOSHOW
ENDIF

@ 4.800,25.833 GET A ;
    PICTURE "@*HN BOOK INFORMATION" ;
    SIZE 1.438,26.000,0.667 ;
    DEFAULT 1 ;
    FONT "Times New Roman", 10 ;
    STYLE "B";
    valid bt('pro1')

@ 7.200,25.833 GET A ;
    PICTURE "@*HN SELLING BOOKS" ;
    SIZE 1.438,26.167,0.667 ;
    DEFAULT 1 ;
    FONT "Times New Roman", 10 ;
    STYLE "B";
    valid bt('pro2')

@ 11.400,25.833 GET A;
    PICTURE "@*HN QUIT " ;
    SIZE 1.438,26.000,0.667 ;
    DEFAULT 1 ;
    FONT "Times New Roman", 10 ;
    STYLE "B";
    valid bt('pro3')

@ 9.200,25.833 GET M.BT_EXIT ;
    PICTURE "@*HN BOOKS TITLES" ;
    SIZE 1.438,26.167,0.667 ;
    DEFAULT 1 ;
    FONT "Times New Roman", 10 ;

```

```

STYLE "B";
valid bt('EXIT')
@ 2.150,25.500 SAY "BOOKS SECTION" ;
FONT "Times New Roman", 14 ;
STYLE "BT" ;
COLOR RGB(255,255,255,,,)
@ 3.500,18.500 TO 3.500,59.167 ;
PEN 4, 8 ;
STYLE "1"

```

```

IF NOT WVISIBLE("_03y1e7gxy")
    ACTIVATE WINDOW _03y1e7gxy
ENDIF

```

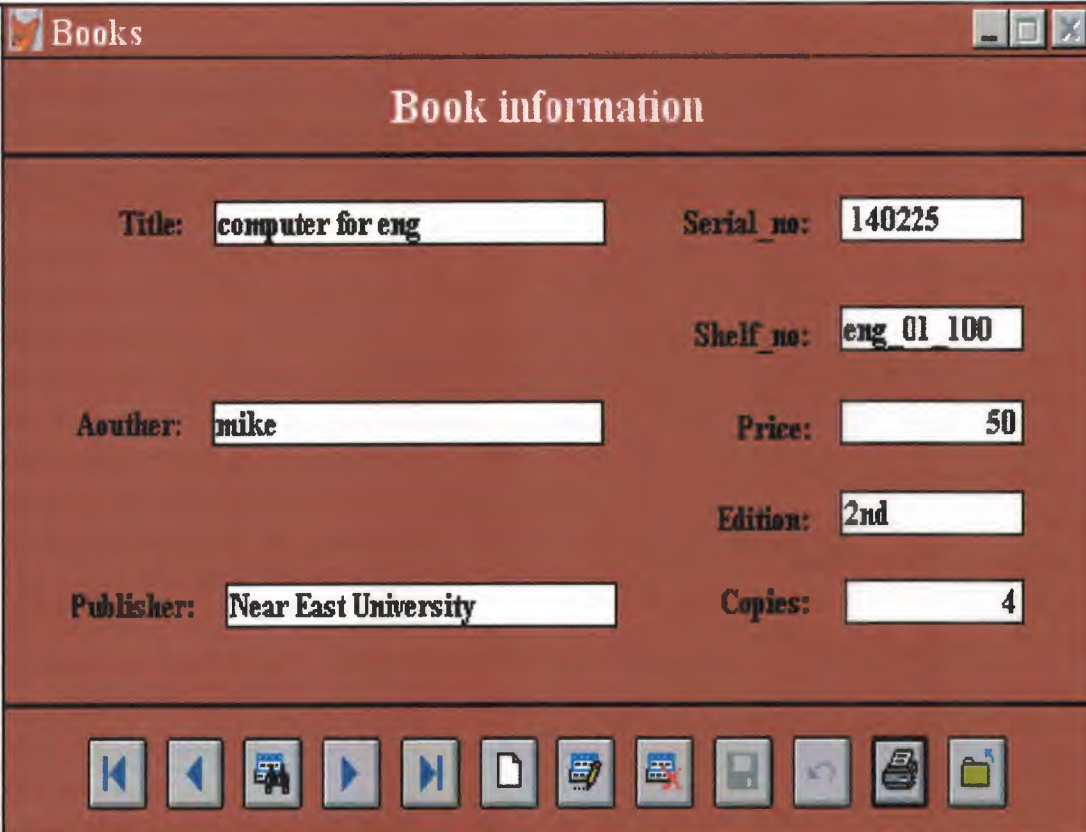
```

READ CYCLE
PROCEDURE BT
    PARAMETER m.btnname
    DO CASE
CASE m.btnname='pro1'
    Do c:\ramiz\books.spr
CASE m.btnname='pro2'
    DO c:\ramiz\sell.spr
    CASE m.btnname='EXIT'
    DO c:\ramiz\title.qpr
CASE m.btnname='pro3'
    set sysm to defa
DEACTIVATE WINDOW _03y1e7gxy
RELEASE WINDOW _03y1e7gxy
    ENDCASE
RETURN

```


BOOK INFORMATION

The book information menu is used for adding the new books or deleting the damaged ones and containing the information about the books like (title, serial number, author, shelf number, price, edition, publisher, and the number of copies) which is shown in the screen.



The screenshot shows a window titled 'Books' with a standard Windows-style title bar (minimize, maximize, close buttons). The main area is titled 'Book information' in a large, bold, serif font. Below the title, there are several input fields arranged in two columns. The left column contains 'Title:', 'Aauthor:', and 'Publisher:'. The right column contains 'Serial_no:', 'Shelf_no:', 'Price:', 'Edition:', and 'Copies:'. Each label is followed by a text input box. The values entered in the boxes are: 'computer for eng', 'mike', 'Near East University', '140225', 'eng_01_100', '50', '2nd', and '4'. At the bottom of the window, there is a horizontal toolbar with 12 icons. From left to right, the icons represent: first page, previous page, search, next page, last page, a document with a magnifying glass, a document with a red 'X' (delete), a floppy disk (save), a circular arrow (refresh), a printer, and a folder with a plus sign (add).

Field	Value
Title:	computer for eng
Serial_no:	140225
Shelf_no:	eng_01_100
Aauthor:	mike
Price:	50
Edition:	2nd
Publisher:	Near East University
Copies:	4

```

* *****
*
* * 05/21/00          BOOKS.SPR          23:19:56
*
* *****
*
* * Author's Name
*
* * Copyright (c) 2000 Company Name
* * Address
* * City, Zip
*
* * Description:
* * This program was automatically generated by GENSCRN.
*
* *****

```

```

* *****
*
* *          BOOKS/Windows Setup Code - SECTION 1
*
* *****

```

```

#REGION 1
PRIVATE wzfields,wztalk
IF SET("TALK") = "ON"
    SET TALK OFF
    m.wztalk = "ON"
ELSE
    m.wztalk = "OFF"
ENDIF
m.wzfields=SET('FIELDS')
SET FIELDS OFF
IF m.wztalk = "ON"
    SET TALK ON
ENDIF
#REGION 0
REGIONAL m.currarea, m.talkstat, m.compstat

IF SET("TALK") = "ON"
    SET TALK OFF
    m.talkstat = "ON"
ELSE
    m.talkstat = "OFF"
ENDIF
m.compstat = SET("COMPATIBLE")

```

SET COMPATIBLE FOXPLUS

m.rborder = SET("READBORDER")
SET READBORDER ON

m.currarea = SELECT()

```
* *****
*
* *   S3997262/Windows Databases, Indexes, Relations
* *
* *****
```

IF USED("books")
 SELECT books
 SET ORDER TO TAG "_03y1drnzc"

ELSE
 SELECT 0
 USE (LOCFILE("books.dbf","DBF","Where is books?"));
 AGAIN ALIAS books ;
 ORDER TAG "_03y1drnzc"

ENDIF

```
* *****
*
* *   Windows Window definitions
* *
* *****
```

IF NOT WEXIST("_03y1e0cpa")
 DEFINE WINDOW _03y1e0cpa ;
 AT 0.000, 0.000 ;
 SIZE 18.188,83.333 ;
 TITLE "Books" ;
 FONT "Times New Roman", 10 ;
 STYLE "B" ;
 FLOAT ;
 CLOSE ;
 MINIMIZE ;
 COLOR RGB(,,128,0,0)
 MOVE WINDOW _03y1e0cpa CENTER

ENDIF

```
* *****
*
* *   BOOKS/Windows Setup Code - SECTION 2
* *
* *****
```

#REGION 1

```

#DEFINE C_DBFEMPTY      'Database is empty, add a record?'
#DEFINE C_EDITS          'Please finish your edits.'
#DEFINE C_TOPFILE        'Top of file.'
#DEFINE C_ENDFILE        'End of file.'
#DEFINE C_BRTITLE        'Locate Record'
#DEFINE C_NOLOCK         'Sorry, could not lock record -- try again later.'
#DEFINE C_ECANCEL        'Edits Canceled.'
#DEFINE C_DELREC         'Delete selected record?'
#DEFINE C_NOFEAT         'Feature not available yet.'
#DEFINE C_NOWIZ          'Wizard application is not available.'
#DEFINE C_MAKEREPO       'Creating report with Report Wizard.'
#DEFINE C_NOREPO         'Could not create report.'
#DEFINE C_DELNOTE        'Deleting records...'
#DEFINE C_READONLY       'Table is read-only. No editing allowed.'
#DEFINE C_NOTABLE        'No table selected. Open table or run query.'
#DEFINE C_BADEXP         'Invalid expression.'
#DEFINE C_LOCWIZ         'Locate WIZARD.APP:'
#DEFINE C_MULTITABLE     'You have multiple related tables. Adding records in not
allowed.'

```

```

MOVE WINDOW '_03y1e0cpa' CENTER
PRIVATE isediting,isadding,wztblarr
PRIVATE wzolddelete,wzolderror,wzoldesc
PRIVATE wzalias, tempcurs,wzlastrec
PRIVATE isreadonly,find_drop,is2table

```

```

IF EMPTY(ALIAS())
    WAIT WINDOW C_NOTABLE
    RETURN
ENDIF
m.wztblarr= "
m.wzalias=SELECT()
m.isediting=.F.
m.isadding=.F.
m.is2table = .F.
m.wzolddelete=SET('DELETE')
SET DELETED ON
m.tempcurs=SYS(2015) &&used if General field
m.wzlastrec = 1
m.wzolderror=ON('error')
ON ERROR DO wizerrorhandler
wzoldesc=ON('KEY','ESCAPE')
ON KEY LABEL ESCAPE
m.find_drop = IIF(_DOS,0,2)

m.isreadonly=IIF(ISREAD(),.T.,.F.)

```



```

IF m.isreadonly
    WAIT WINDOW C_READONLY TIMEOUT 1
ENDIF

IF RECCOUNT()=0 AND !m.isreadonly AND fox_alert(C_DBFEMPTY)
    APPEND BLANK
ENDIF

GOTO TOP
SCATTER MEMVAR MEMO

* *****
*
*      BOOKS/Windows Screen Layout
*
* *****
*

#REGION 1
IF WVISIBLE("_03y1e0cpa")
    ACTIVATE WINDOW _03y1e0cpa SAME
ELSE
    ACTIVATE WINDOW _03y1e0cpa NOSHOW
ENDIF
@ 2.188,0.000 TO 2.188,83.333 ;
    PEN 2, 8 ;
    STYLE "1"
@ 3.375,4.333 SAY "Title:" ;
    SIZE 0.938,9.500 ;
    FONT "Times New Roman", 10 ;
    STYLE "BT" ;
    PICTURE "@J" ;
    COLOR RGB(,,255,255,255)
@ 3.375,16.500 GET m.title ;
    SIZE 0.938,29.167 ;
    DEFAULT " " ;
    FONT "Times New Roman", 10 ;
    STYLE "B" ;
    PICTURE "@K XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX" ;
    WHEN isediting ;
    COLOR ,RGB(0,0,0,255,255,255)
@ 8.063,4.167 SAY "Aouther:" ;
    SIZE 0.938,9.500 ;
    FONT "Times New Roman", 10 ;
    STYLE "BT" ;

```

PICTURE "@J" ;
 COLOR RGB(,,255,255,255)
 @ 8.063,16.333 GET m.aouthier ;
 SIZE 0.938,29.167 ;
 DEFAULT " " ;
 FONT "Times New Roman", 10 ;
 STYLE "B" ;
 PICTURE "@K XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX" ;
 WHEN isediting ;
 COLOR ,RGB(0,0,0,255,255,255)
 @ 3.375,52.167 SAY "Serial_no:" ;
 SIZE 0.938,9.500 ;
 FONT "Times New Roman", 10 ;
 STYLE "BT" ;
 PICTURE "@J" ;
 COLOR RGB(,,255,255,255)
 @ 3.313,64.333 GET m.serial_no ;
 SIZE 0.938,13.333 ;
 DEFAULT " " ;
 FONT "Times New Roman", 10 ;
 STYLE "B" ;
 PICTURE "@K XXXXXXXXXXXX" ;
 WHEN isediting ;
 COLOR ,RGB(0,0,0,255,255,255)
 @ 6.000,52.167 SAY "Shelf_no:" ;
 SIZE 0.938,9.500 ;
 FONT "Times New Roman", 10 ;
 STYLE "BT" ;
 PICTURE "@J" ;
 COLOR RGB(,,255,255,255)
 @ 5.875,64.333 GET m.shelf_no ;
 SIZE 0.938,13.333 ;
 DEFAULT " " ;
 FONT "Times New Roman", 10 ;
 STYLE "B" ;
 PICTURE "@K XXXXXXXXXXXX" ;
 WHEN isediting ;
 COLOR ,RGB(0,0,0,255,255,255)
 @ 8.188,52.333 SAY "Price:" ;
 SIZE 0.938,9.500 ;
 FONT "Times New Roman", 10 ;
 STYLE "BT" ;
 PICTURE "@J" ;
 COLOR RGB(,,255,255,255)
 @ 8.063,64.333 GET m.price ;
 SIZE 0.938,13.333 ;

DEFAULT " " ;
 FONT "Times New Roman", 10 ;
 STYLE "B" ;
 PICTURE "@K 99,999" ;
 WHEN isediting ;
 COLOR ,RGB(0,0,0,255,255,255)
 @ 12.313,5.167 SAY "Publisher:" ;
 SIZE 0.938,9.500 ;
 FONT "Times New Roman", 10 ;
 STYLE "BT" ;
 PICTURE "@J" ;
 COLOR RGB(,,255,255,255)
 @ 12.313,17.333 GET m.publisher ;
 SIZE 0.938,29.167 ;
 DEFAULT " " ;
 FONT "Times New Roman", 10 ;
 STYLE "B" ;
 PICTURE "@K XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX" ;
 WHEN isediting ;
 COLOR ,RGB(0,0,0,255,255,255)
 @ 10.313,52.167 SAY "Edition:" ;
 SIZE 0.938,9.500 ;
 FONT "Times New Roman", 10 ;
 STYLE "BT" ;
 PICTURE "@J" ;
 COLOR RGB(,,255,255,255)
 @ 10.188,64.333 GET m.edition ;
 SIZE 0.938,13.333 ;
 DEFAULT " " ;
 FONT "Times New Roman", 10 ;
 STYLE "B" ;
 PICTURE "@K XXXXX" ;
 WHEN isediting ;
 COLOR ,RGB(0,0,0,255,255,255)
 @ 12.250,52.333 SAY "Copies:" ;
 SIZE 0.938,9.333 ;
 FONT "Times New Roman", 10 ;
 STYLE "BT" ;
 PICTURE "@J" ;
 COLOR RGB(,,255,255,255)
 @ 12.250,64.667 GET m.copies ;
 SIZE 0.938,13.000 ;
 DEFAULT " " ;
 FONT "Times New Roman", 10 ;
 STYLE "B" ;
 PICTURE "@K 99,999" ;

```

        WHEN isediting ;
        COLOR ,RGB(0,0,0,255,255,255)
@ 15.063,0.000 TO 15.063,83.333 ;
        PEN 2, 8 ;
        STYLE "1"
@ 15.875,6.333 GET m.top_btn ;
        PICTURE "@*BHN " + ;

        (LOCFILE("\fpw26\wizards\wizbmps\wztop.bmp", "BMP|ICO|PCT|ICN", "Where
is wztop?")) ;
        SIZE 1.688,4.500,0.667 ;
        DEFAULT 1 ;
        FONT "Times New Roman", 10 ;
        STYLE "B" ;
        VALID btn_val('TOP') ;
        MESSAGE 'Go to first record.'
@ 15.875,12.333 GET m.prev_btn ;
        PICTURE "@*BHN " + ;

        (LOCFILE("\fpw26\wizards\wizbmps\wzback.bmp", "BMP|ICO|PCT|ICN", "Where
is wzback?")) ;
        SIZE 1.688,4.500,0.667 ;
        DEFAULT 1 ;
        FONT "Times New Roman", 10 ;
        STYLE "B" ;
        VALID btn_val('PREV') ;
        MESSAGE 'Go to previous record.'
@ 15.875,18.333 GET m.loc_btn ;
        PICTURE "@*BHN " + ;

        (LOCFILE("\fpw26\wizards\wizbmps\wzlocate.bmp", "BMP|ICO|PCT|ICN", "Where
is wzlocate?")) ;
        SIZE 1.688,4.500,0.667 ;
        DEFAULT 1 ;
        FONT "Times New Roman", 10 ;
        STYLE "B" ;
        VALID btn_val('LOCATE') ;
        MESSAGE 'Locate a record.'
@ 15.875,24.333 GET m.next_btn ;
        PICTURE "@*BHN " + ;

        (LOCFILE("\fpw26\wizards\wizbmps\wznnext.bmp", "BMP|ICO|PCT|ICN", "Where
is wznext?")) ;
        SIZE 1.688,4.500,0.667 ;
        DEFAULT 1 ;
        FONT "Times New Roman", 10 ;

```



```

STYLE "B" ;
VALID btn_val('NEXT') ;
MESSAGE 'Go to next record.'
@ 15.875,30.333 GET m.end_btn ;
PICTURE "@*BHN " + ;

(LOCFILE("\fpw26\wizards\wizbmps\wzend.bmp", "BMP|ICO|PCT|ICN", "Where
is wzend?")) ;
SIZE 1.688,4.500,0.667 ;
DEFAULT 1 ;
FONT "Times New Roman", 10 ;
STYLE "B" ;
VALID btn_val('END') ;
MESSAGE 'Go to last record.'
@ 15.875,36.333 GET m.add_btn ;
PICTURE "@*BHN " + ;

(LOCFILE("\fpw26\wizards\wizbmps\wznew.bmp", "BMP|ICO|PCT|ICN", "Where
is wznew?")) ;
SIZE 1.688,4.500,0.667 ;
DEFAULT 1 ;
FONT "Times New Roman", 10 ;
STYLE "B" ;
VALID btn_val('ADD') ;
MESSAGE 'Add a new record.'
@ 15.875,42.333 GET m.edit_btn ;
PICTURE "@*BHN " + ;

(LOCFILE("\fpw26\wizards\wizbmps\wzedit.bmp", "BMP|ICO|PCT|ICN", "Where
is wzedit?")) ;
SIZE 1.688,4.500,0.667 ;
DEFAULT 1 ;
FONT "Times New Roman", 10 ;
STYLE "B" ;
VALID btn_val('EDIT') ;
MESSAGE 'Edit current record.'
@ 15.875,48.333 GET m.del_btn ;
PICTURE "@*BHN " + ;

(LOCFILE("\fpw26\wizards\wizbmps\wzdelete.bmp", "BMP|ICO|PCT|ICN", "Where
is wzdelete?")) ;
SIZE 1.688,4.500,0.667 ;
DEFAULT 1 ;
FONT "Times New Roman", 10 ;
STYLE "B" ;
VALID btn_val('DELETE') ;

```

```

MESSAGE 'Delete current record.'
@ 15.875,54.333 GET m.save_btn ;
    PICTURE "@*BHN " + ;

    (LOCFILE("\fpw26\wizards\wizbmps\wzsave.bmp", "BMP|ICO|PCT|ICN", "Where is wzsave?")) ;
    SIZE 1.688,4.500,0.667 ;
    DEFAULT 1 ;
    FONT "Times New Roman", 10 ;
    STYLE "B" ;
    VALID btn_val('SAVE') ;
    MESSAGE 'Save edits.'
@ 15.875,60.333 GET m.can_btn ;
    PICTURE "@*BHN " + ;

    (LOCFILE("\fpw26\wizards\wizbmps\wzundo.bmp", "BMP|ICO|PCT|ICN", "Where is wzundo?")) ;
    SIZE 1.688,4.500,0.667 ;
    DEFAULT 1 ;
    FONT "Times New Roman", 10 ;
    STYLE "B" ;
    VALID btn_val('CANCEL') ;
    MESSAGE 'Cancel edits.'
@ 15.875,66.333 GET m.prnt_btn ;
    PICTURE "@*BHN " + ;

    (LOCFILE("\fpw26\wizards\wizbmps\wzprint.bmp", "BMP|ICO|PCT|ICN", "Where is wzprint?")) ;
    SIZE 1.688,4.500,0.667 ;
    DEFAULT 1 ;
    FONT "Times New Roman", 10 ;
    STYLE "B" ;
    VALID btn_val('PRINT') ;
    MESSAGE 'Print report.'
@ 15.875,72.333 GET m.exit_btn ;
    PICTURE "@*BHN " + ;

    (LOCFILE("\fpw26\wizards\wizbmps\wzclose.bmp", "BMP|ICO|PCT|ICN", "Where is wzclose?")) ;
    SIZE 1.688,4.500,0.667 ;
    DEFAULT 1 ;
    FONT "Times New Roman", 10 ;
    STYLE "B" ;
    VALID btn_val('EXIT') ;
    MESSAGE 'Close screen.'
@ 0.438,29.667 SAY "Book information" ;

```

```

        FONT "Times New Roman", 14 ;
        STYLE "BT" ;
        COLOR RGB(255,255,255,,,)

IF NOT WVISIBLE("_03y1e0cpa")
    ACTIVATE WINDOW _03y1e0cpa
ENDIF

* *****
* *
* *   WindowsREAD contains clauses from SCREEN s3997262
* *
* *****
*

READ CYCLE ;
    ACTIVATE READACT() ;
    DEACTIVATE READDEAC() ;
    NOLOCK

RELEASE WINDOW _03y1e0cpa

* *****
* *
* *           Windows Closing Databases
* *
* *****
*

IF USED("books")
    SELECT books
    USE
ENDIF

SELECT (m.currarea)

#REGION 0

SET READBORDER &rborder

IF m.talkstat = "ON"
    SET TALK ON
ENDIF
IF m.compstat = "ON"

```

```
        SET COMPATIBLE ON
ENDIF
```

```
* *****
*
*      BOOKS/Windows Cleanup Code
*
* *****
*
```

```
#REGION 1
SET DELETED &wzolddelete
SET FIELDS &wzfields
ON ERROR &wzolderror
ON KEY LABEL ESCAPE &wzoldesc
DO CASE
CASE _DOS AND SET('DISPLAY')='VGA25'
    @24,0 CLEAR TO 24,79
CASE _DOS AND SET('DISPLAY')='VGA50'
    @49,0 CLEAR TO 49,79
CASE _DOS
    @24,0 CLEAR TO 24,79
ENDCASE
```

```
****Procedures****
```

```
* *****
*
*      BOOKS/Windows Supporting Procedures and Functions
*
* *****
*
```

```
#REGION 1
PROCEDURE readdeac
    IF isediting
        ACTIVATE WINDOW '_03y1e0cpa'
        WAIT WINDOW C_EDIT$ NOWAIT
    ENDIF
    IF !WVISIBLE(WOUTPUT())
        CLEAR READ
        RETURN .T.
    ENDIF
    RETURN .F.
```


PROCEDURE readact

IF !isediting

SELECT (m.wzalias)

SHOW GETS

ENDIF

DO REFRESH

RETURN

PROCEDURE wizerrorhandler

* This very simple error handler is primarily intended

* to trap for General field OLE errors which may occur

* during editing from the MODIFY GENERAL window.

WAIT WINDOW message()

RETURN

PROCEDURE printrec

PRIVATE sOldError,wizfname,saverec,savearea,tmpcurs,tmpstr

PRIVATE prnt_btn,p_recs,p_output,pr_out,pr_record

STORE 1 TO p_recs,p_output

STORE 0 TO prnt_btn

STORE RECNO() TO saverec

m.sOldError=ON('error')

DO pdialog

IF m.prnt_btn = 2

RETURN

ENDIF

IF !FILE(ALIAS()+'FRX')

m.wizfname=SYS(2004)+'WIZARDS\'+'WIZARD.APP'

IF !FILE(m.wizfname)

ON ERROR *

m.wizfname=LOCFILE('WIZARD.APP','APP',C_LOCWIZ)

ON ERROR &sOldError

IF !'WIZARD.APP'\$UPPER(m.wizfname)

WAIT WINDOW C_NOWIZ

RETURN

ENDIF

ENDIF

WAIT WINDOW C_MAKEREPO NOWAIT

m.savearea=SELECT()

m.tmpcurs='_'+LEFT(SYS(3),7)

CREATE CURSOR (m.tmpcurs) (comment m)

m.tmpstr = '* LAYOUT = COLUMNAR'+CHR(13)+CHR(10)

INSERT INTO (m.tmpcurs) VALUES(m.tmpstr)

SELECT (m.savearea)

```

        DO (m.wizfname) WITH
        'WZ_QREPO','NOSCRN/CREATE',ALIAS(),m.tmpcurs
        USE IN (m.tmpcurs)
    WAIT CLEAR
        IF !FILE(ALIAS()+'.FRX') &&wizard could not create report
        WAIT WINDOW C_NOREPO
        RETURN
    ENDIF
ENDIF

m.pr_out=IIF(m.p_output=1,'TO PRINT NOCONSOLE','PREVIEW')
m.pr_record=IIF(m.p_recs=1,'NEXT 1','ALL')
REPORT FORM (ALIAS()) &pr_out &pr_record
GO m.saverec

RETURN

```

```

PROCEDURE BTN_VAL
    PARAMETER m.btnname
    DO CASE
    CASE m.btnname='TOP'
        GO TOP
        WAIT WINDOW C_TOPFILE NOWAIT
    CASE m.btnname='PREV'
        IF !BOF()
            SKIP -1
        ENDIF
        IF BOF()
            WAIT WINDOW C_TOPFILE NOWAIT
            GO TOP
        ENDIF
    CASE m.btnname='NEXT'
        IF !EOF()
            SKIP 1
        ENDIF
        IF EOF()
            WAIT WINDOW C_ENDFILE NOWAIT
            GO BOTTOM
        ENDIF
    CASE m.btnname='END'
        GO BOTTOM
        WAIT WINDOW C_ENDFILE NOWAIT
    CASE m.btnname='LOCATE'
        DO loc_dlog
    CASE m.btnname='ADD' AND !isediting &&add record
        isediting=.T.
    
```

```

isadding=.T.
=edithand('ADD')
_curobj=1
DO refresh
SHOW GETS
RETURN
CASE m.btnname='EDIT' AND !isediting &&edit record
IF EOF() OR BOF()
    WAIT WINDOW C__ENDFILE NOWAIT
    RETURN
ENDIF
IF RLOCK()
    isediting=.T.
    _curobj=1
    DO refresh
    RETURN
ELSE
    WAIT WINDOW C__NOLOCK
    RETURN
ENDIF
CASE m.btnname='SAVE' AND isediting &&save record
IF isadding
    =edithand('SAVE')
ELSE
    GATHER MEMVAR MEMO
ENDIF
UNLOCK
isediting=.F.
isadding=.F.
DO refresh
CASE m.btnname='CANCEL' AND isediting &&cancel record
IF isadding
    =edithand('CANCEL')
ENDIF
isediting=.F.
isadding=.F.
UNLOCK
WAIT WINDOW C__ECANCEL NOWAIT
DO refresh
CASE m.btnname='DELETE'
IF EOF() OR BOF()
    WAIT WINDOW C__ENDFILE NOWAIT
    RETURN
ENDIF
IF fox_alert(C__DELREC)
    DELETE

```

```

        IF !EOF() AND DELETED()
            SKIP 1
        ENDIF
        IF EOF()
            WAIT WINDOW C__ENDFILE NOWAIT
            GO BOTTOM
        ENDIF
    ELSE
        RETURN
    ENDIF
CASE m.btnname='PRINT'
    DO printrec
    RETURN
CASE m.btnname='EXIT'
    m.bailout=.T. &&this is needed if used with FoxApp
    CLEAR READ
    RETURN
OTHERWISE
    RETURN
ENDCASE
SCATTER MEMVAR MEMO
SHOW GETS
RETURN

```

PROCEDURE REFRESH

```

DO CASE
CASE m.isreadonly AND RECCOUNT()=0
    SHOW GETS DISABLE
    SHOW GET exit_btn ENABLE
CASE m.isreadonly
    SHOW GET add_btn DISABLE
    SHOW GET edit_btn DISABLE
    SHOW GET del_btn DISABLE
    SHOW GET save_btn DISABLE
    SHOW GET can_btn DISABLE
CASE (RECCOUNT()=0 OR EOF()) AND !m.isediting
    SHOW GETS DISABLE
    SHOW GET add_btn ENABLE
    SHOW GET exit_btn ENABLE
CASE m.isediting
    SHOW GET find_drop DISABLE
    SHOW GET top_btn DISABLE
    SHOW GET prev_btn DISABLE
    SHOW GET loc_btn DISABLE
    SHOW GET next_btn DISABLE

```



```

    SHOW GET end_btn DISABLE
    SHOW GET add_btn DISABLE
    SHOW GET edit_btn,1 DISABLE
    SHOW GET del_btn,1 DISABLE
    SHOW GET prnt_btn DISABLE
    SHOW GET exit_btn DISABLE
    SHOW GET save_btn ENABLE
    SHOW GET can_btn ENABLE
    ON KEY LABEL ESCAPE DO BTN_VAL WITH 'CANCEL'
    RETURN
OTHERWISE
    SHOW GET find_drop ENABLE
    SHOW GET top_btn ENABLE
    SHOW GET prev_btn ENABLE
    SHOW GET loc_btn ENABLE
    SHOW GET next_btn ENABLE
    SHOW GET end_btn ENABLE
    SHOW GET add_btn ENABLE
    SHOW GET edit_btn,1 ENABLE
    SHOW GET del_btn,1 ENABLE
    SHOW GET prnt_btn ENABLE
    SHOW GET exit_btn ENABLE
    SHOW GET save_btn DISABLE
    SHOW GET can_btn DISABLE
ENDCASE
IF m.is2table
    SHOW GET add_btn DISABLE
ENDIF
ON KEY LABEL ESCAPE
RETURN

PROCEDURE edithand
    PARAMETER m.paction
    * procedure handles edits
    DO CASE
    CASE m.paction = 'ADD'
        SCATTER MEMVAR MEMO BLANK
    CASE m.paction = 'SAVE'
        INSERT INTO (ALIAS()) FROM MEMVAR
    CASE m.paction = 'CANCEL'
        * nothing here
    ENDCASE
RETURN

PROCEDURE fox_alert
    PARAMETER wzalrtmess

```

```

PRIVATE alrtbtn
m.alrtbtn=2
  DEFINE WINDOW _qec1ij2t7 AT 0,0 SIZE 8,50 ;
  FONT "MS Sans Serif",10 STYLE 'B' ;
  FLOAT NOCLOSE NOMINIMIZE DOUBLE TITLE WTITLE()
  MOVE WINDOW _qec1ij2t7 CENTER
  ACTIVATE WINDOW _qec1ij2t7 NOSHOW
  @ 2,(50-txtwidth(wzalrtmess))/2 SAY wzalrtmess;
  FONT "MS Sans Serif", 10 STYLE "B"
  @ 6,18 GET m.alrtbtn ;
  PICTURE "@*HT \<OK;\?! \<Cancel" ;
  SIZE 1.769,8.667,1.333 ;
  FONT "MS Sans Serif", 8 STYLE "B"
  ACTIVATE WINDOW _qec1ij2t7
  READ CYCLE MODAL
  RELEASE WINDOW _qec1ij2t7
RETURN m.alrtbtn=1

```

```

PROCEDURE pdialog
  DEFINE WINDOW _qjn12zbvh ;
  AT 0.000, 0.000 ;
  SIZE 13.231,54.800 ;
  TITLE "Microsoft FoxPro" ;
  FONT "MS Sans Serif", 8 ;
  FLOAT NOCLOSE MINIMIZE SYSTEM
  MOVE WINDOW _qjn12zbvh CENTER
  ACTIVATE WINDOW _qjn12zbvh NOSHOW
  @ 2.846,33.600 SAY "Output:" ;
  FONT "MS Sans Serif", 8 ;
  STYLE "BT"
  @ 2.846,4.800 SAY "Print:" ;
  FONT "MS Sans Serif", 8 ;
  STYLE "BT"
  @ 4.692,7.200 GET m.p_recs ;
  PICTURE "@*RVN \<Current Record;\<All Records" ;
  SIZE 1.308,18.500,0.308 ;
  DEFAULT 1 ;
  FONT "MS Sans Serif", 8 ;
  STYLE "BT"
  @ 4.692,36.000 GET m.p_output ;
  PICTURE "@*RVN \<Printer;Pre\<view" ;
  SIZE 1.308,12.000,0.308 ;
  DEFAULT 1 ;
  FONT "MS Sans Serif", 8 ;
  STYLE "BT"

```

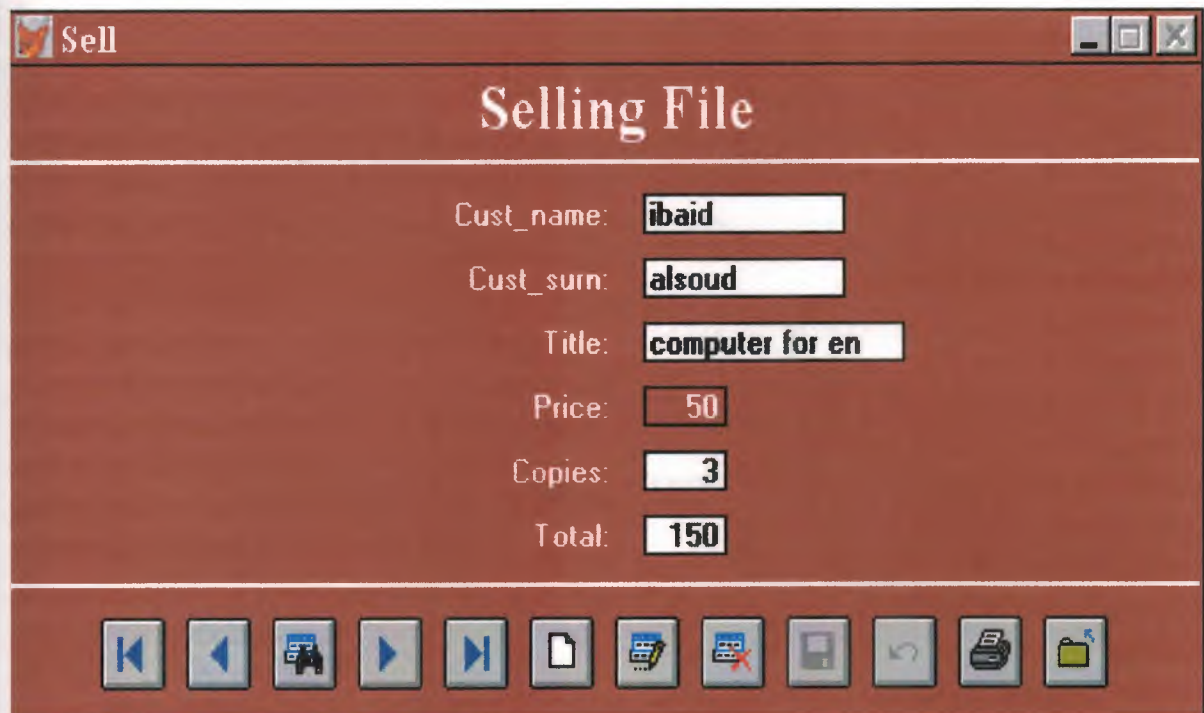
```

@ 10.154,16.600 GET m.prnt_btn ;
    PICTURE "@*HT P\<rint;Ca\<ncel" ;
    SIZE 1.769,8.667,0.667 ;
    DEFAULT 1 ;
    FONT "MS Sans Serif", 8 ;
    STYLE "B"
ACTIVATE WINDOW _qjn12zbvh
READ CYCLE MODAL
RELEASE WINDOW _qjn12zbvh
RETURN
PROCEDURE loc_dlog
    PRIVATE gfields,i
    DEFINE WINDOW wzlocate FROM 1,1 TO 20,40;
        SYSTEM GROW CLOSE ZOOM FLOAT FONT "MS Sans Serif",8
    MOVE WINDOW wzlocate CENTER
    m.gfields=SET('FIELDS',2)
    IF !EMPTY(RELATION(1))
        SET FIELDS ON
        IF m.gfields # 'GLOBAL'
            SET FIELDS GLOBAL
        ENDIF
        IF EMPTY(FLDLIST())
            m.i=1
            DO WHILE !EMPTY(OBJVAR(m.i))
                IF ATC('M.',OBJVAR(m.i))=0
                    SET FIELDS TO (OBJVAR(m.i))
                ENDIF
                m.i = m.i + 1
            ENDDO
        ENDIF
    ENDIF
    BROWSE WINDOW wzlocate NOEDIT NODELETE ;
        NOMENU TITLE C_BRTITLE
    SET FIELDS &gfields
    SET FIELDS OFF
    RELEASE WINDOW wzlocate
RETURN

```

SELLING FILE

The selling file menu is used for giving the all information about the book store sales daily, which contains the (customer name, his/her surname, title of the book, price, the number of the copies, and the total money), in this you take the full information about the book that you sold it .



The screenshot shows a window titled "Sell" with a sub-header "Selling File". The window contains several input fields for recording a sale:

Field	Value
Cust_name:	ibaid
Cust_surn:	alsoud
Title:	computer for en
Price:	50
Copies:	3
Total:	150

At the bottom of the window is a toolbar with icons for navigation (back, forward, search, etc.), file operations (open, save, print), and other functions.


```

* *****
*
* * 05/30/00          SEL.SPR          22:29:49
*
* *****
*
* * Author's Name
*
* * Copyright (c) 2000 Company Name
* * Address
* * City, Zip
*
* * Description:
* * This program was automatically generated by GENSCRN.
*
* *****

```

```

* *****
*
* *          SEL/Windows Setup Code - SECTION 1
*
* *****

```

```

#REGION 1
PRIVATE wzfields,wztalk
IF SET("TALK") = "ON"
    SET TALK OFF
    m.wztalk = "ON"
ELSE
    m.wztalk = "OFF"
ENDIF
m.wzfields=SET('FIELDS')
SET FIELDS OFF
IF m.wztalk = "ON"
    SET TALK ON
ENDIF

```

```

#REGION 0
REGIONAL m.currarea, m.talkstat, m.compstat

IF SET("TALK") = "ON"
    SET TALK OFF

```

```

        m.talkstat = "ON"
ELSE
        m.talkstat = "OFF"
ENDIF
m.compstat = SET("COMPATIBLE")
SET COMPATIBLE FOXPLUS

```

```

m.rborder = SET("READBORDER")
SET READBORDER ON

```

```

m.currarea = SELECT()

```

```

*      *****
*      *
*      *      SEL/Windows Databases, Indexes, Relations
*      *
*      *****
*

```

```

IF USED("sell")
        SELECT sell
        SET ORDER TO TAG "_04717qix2"
ELSE
        SELECT 0
        USE (LOCFILE("sell.dbf","DBF","Where is sell?"));
        AGAIN ALIAS sell ;
        ORDER TAG "_04717qix2"
ENDIF

```

```

*      *****
*      *
*      *      Windows Window definitions
*      *
*      *****
*

```

```

IF NOT WEXIST("_0471c7wft")
        DEFINE WINDOW _0471c7wft ;
                AT 0.000,0.000 ;
                SIZE 18.692,83.333 ;
                TITLE "Sell" ;
                FONT "MS Sans Serif", 8 ;
                STYLE "B" ;
                FLOAT ;

```

```

CLOSE ;
MINIMIZE ;
COLOR RGB(,,128,0,0)
MOVE WINDOW _0471c7wft CENTER
ENDIF

```

```

* *****
* *
* *      SEL/Windows Setup Code - SECTION 2
* *
* *****
*

```

```
#REGION 1
```

```

#DEFINE C_DBFEMPTY      'Database is empty, add a record?'
#DEFINE C_EDITS          'Please finish your edits.'
#DEFINE C_TOPFILE        'Top of file.'
#DEFINE C_ENDFILE        'End of file.'
#DEFINE C_BRTITLE        'Locate Record'
#DEFINE C_NOLOCK         'Sorry, could not lock record -- try again later.'
#DEFINE C_ECANCEL        'Edits Canceled.'
#DEFINE C_DELREC         'Delete selected record?'
#DEFINE C_NOFEAT         'Feature not available yet.'
#DEFINE C_NOWIZ          'Wizard application is not available.'
#DEFINE C_MAKEREPO       'Creating report with Report Wizard.'
#DEFINE C_NOREPO         'Could not create report.'
#DEFINE C_DELNOTE        'Deleting records...'
#DEFINE C_READONLY       'Table is read-only. No editing allowed.'
#DEFINE C_NOTABLE        'No table selected. Open table or run query.'
#DEFINE C_BADEXP         'Invalid expression.'
#DEFINE C_LOCWIZ         'Locate WIZARD.APP:'
#DEFINE C_MULTITABLE     'You have multiple related tables. Adding records in not
allowed.'

```

```

MOVE WINDOW ' _0471c7wft' CENTER
PRIVATE isediting,isadding,wztblarr
PRIVATE wzolddelete,wzolderror,wzoldesc
PRIVATE wzalias, tempcurs,wzlastrec
PRIVATE isreadonly,find_drop,is2table

```

```

IF EMPTY(ALIAS())
    WAIT WINDOW C_NOTABLE
    RETURN
ENDIF

```

```

m.wztblarr= "
m.wzalias=SELECT()
m.isediting=.F.
m.isadding=.F.
m.is2table = .F.
m.wzolddelete=SET('DELETE')
SET DELETED ON
m.tempcurs=SYS(2015) &&used if General field
m.wzlastrec = 1
m.wzolderror=ON('error')
ON ERROR DO wizerrorhandler
wzoldesc=ON('KEY','ESCAPE')
ON KEY LABEL ESCAPE
m.find_drop = IIF(_DOS,0,2)

m.isreadonly=IIF(ISREAD(),.T.,.F.)
IF m.isreadonly
    WAIT WINDOW C_READONLY TIMEOUT 1
ENDIF

IF RECCOUNT()=0 AND !m.isreadonly AND fox_alert(C_DBFEMPTY)
    APPEND BLANK
ENDIF

GOTO TOP
SCATTER MEMVAR MEMO

*      *****
*      *
*      *      SEL/Windows Screen Layout
*      *
*      *****
*

#REGION 1
IF WVISIBLE("_0471c7wft")
    ACTIVATE WINDOW _0471c7wft SAME
ELSE
    ACTIVATE WINDOW _0471c7wft NOSHOW
ENDIF
@ 0.231,32.833 SAY "Selling File" ;
    FONT "Times New Roman", 18 ;
    STYLE "BT" ;
    COLOR RGB(255,255,255,255,255,255)

```


@ 2.692,0.000 TO 2.692,83.333 ;
 PEN 2, 8 ;
 STYLE "1" ;
 COLOR RGB(255,255,255,255,255,255)
 @ 3.769,31.000 SAY "Cust_name:" ;
 SIZE 1.000,11.000 ;
 FONT "MS Sans Serif", 8 ;
 STYLE "BT" ;
 PICTURE "@J" ;
 COLOR RGB(255,255,255,255,255,255)
 @ 3.769,44.667 GET m.cust_name ;
 SIZE 1.000,13.333 ;
 DEFAULT " " ;
 FONT "MS Sans Serif", 8 ;
 STYLE "B" ;
 PICTURE "@K XXXXXXXXXXXX" ;
 WHEN isediting ;
 COLOR ,RGB(0,0,0,255,255,255)
 @ 5.615,31.000 SAY "Cust_surn:" ;
 SIZE 1.000,11.000 ;
 FONT "MS Sans Serif", 8 ;
 STYLE "BT" ;
 PICTURE "@J" ;
 COLOR RGB(255,255,255,255,255,255)
 @ 5.615,44.667 GET m.cust_surn ;
 SIZE 1.000,13.333 ;
 DEFAULT " " ;
 FONT "MS Sans Serif", 8 ;
 STYLE "B" ;
 PICTURE "@K XXXXXXXXXXXX" ;
 WHEN isediting ;
 COLOR ,RGB(0,0,0,255,255,255)
 @ 7.462,31.000 SAY "Title:" ;
 SIZE 1.000,11.000 ;
 FONT "MS Sans Serif", 8 ;
 STYLE "BT" ;
 PICTURE "@J" ;
 COLOR RGB(255,255,255,255,255,255)
 @ 7.462,44.667 GET m.title ;
 SIZE 1.000,17.500 ;
 DEFAULT " " ;
 FONT "MS Sans Serif", 8 ;
 STYLE "B" ;
 PICTURE "@K XXXXXXXXXXXXXXXXXXXX" ;
 WHEN isediting ;
 COLOR ,RGB(0,0,0,255,255,255)

@ 9.308,31.000 SAY "Price:" ;
 SIZE 1.000,11.000 ;
 FONT "MS Sans Serif", 8 ;
 STYLE "BT" ;
 PICTURE "@J" ;
 COLOR RGB(255,255,255,255,255,255)
 @ 9.308,44.667 GET m.price ;
 SIZE 1.000,5.000 ;
 DEFAULT " " ;
 FONT "MS Sans Serif", 8 ;
 STYLE "B" ;
 PICTURE "@K 9999" ;
 WHEN price() ;
 COLOR ,RGB(0,0,0,255,255,255)
 @ 11.154,31.000 SAY "Copies:" ;
 SIZE 1.000,11.000 ;
 FONT "MS Sans Serif", 8 ;
 STYLE "BT" ;
 PICTURE "@J" ;
 COLOR RGB(255,255,255,255,255,255)
 @ 11.154,44.667 GET m.copies ;
 SIZE 1.000,5.000 ;
 DEFAULT " " ;
 FONT "MS Sans Serif", 8 ;
 STYLE "B" ;
 PICTURE "@K 9999" ;
 WHEN isediting ;
 COLOR ,RGB(0,0,0,255,255,255)
 @ 13.000,31.000 SAY "Total:" ;
 SIZE 1.000,11.000 ;
 FONT "MS Sans Serif", 8 ;
 STYLE "BT" ;
 PICTURE "@J" ;
 COLOR RGB(255,255,255,255,255,255)
 @ 13.000,44.667 GET m.total ;
 SIZE 1.000,5.000 ;
 DEFAULT " " ;
 FONT "MS Sans Serif", 8 ;
 STYLE "B" ;
 PICTURE "@K 9999" ;
 WHEN AA() ;
 COLOR ,RGB(0,0,0,255,255,255)
 @ 14.846,0.000 TO 14.846,83.333 ;
 PEN 2, 8 ;
 STYLE "1" ;
 COLOR RGB(255,255,255,255,255,255)

@ 15.846,6.333 GET m.top_btn ;
PICTURE "@*BHN " + ;

(LOCFILE("\fpw26\wizards\wizbmps\wztop.bmp", "BMP|ICO|PCT|ICN", "Where
is wztop?")) ;

SIZE 2.077,4.500,0.667 ;
DEFAULT 1 ;
FONT "MS Sans Serif", 8 ;
STYLE "B" ;
VALID btn_val("TOP") ;
MESSAGE 'Go to first record.'

@ 15.846,12.333 GET m.prev_btn ;
PICTURE "@*BHN " + ;

(LOCFILE("\fpw26\wizards\wizbmps\wzback.bmp", "BMP|ICO|PCT|ICN", "Where
is wzback?")) ;

SIZE 2.077,4.500,0.667 ;
DEFAULT 1 ;
FONT "MS Sans Serif", 8 ;
STYLE "B" ;
VALID btn_val("PREV") ;
MESSAGE 'Go to previous record.'

@ 15.846,18.333 GET m.loc_btn ;
PICTURE "@*BHN " + ;

(LOCFILE("\fpw26\wizards\wizbmps\wzlocate.bmp", "BMP|ICO|PCT|ICN", "Where
is wzlocate?")) ;

SIZE 2.077,4.500,0.667 ;
DEFAULT 1 ;
FONT "MS Sans Serif", 8 ;
STYLE "B" ;
VALID btn_val("LOCATE") ;
MESSAGE 'Locate a record.'

@ 15.846,24.333 GET m.next_btn ;
PICTURE "@*BHN " + ;

(LOCFILE("\fpw26\wizards\wizbmps\wznnext.bmp", "BMP|ICO|PCT|ICN", "Where
is wznnext?")) ;

SIZE 2.077,4.500,0.667 ;
DEFAULT 1 ;
FONT "MS Sans Serif", 8 ;
STYLE "B" ;
VALID btn_val("NEXT") ;
MESSAGE 'Go to next record.'

@ 15.846,30.333 GET m.end_btn ;
PICTURE "@*BHN " + ;

```
(LOCFILE("\fpw26\wizards\wizbmps\wzend.bmp", "BMP|ICO|PCT|ICN", "Where
is wzend?"));
SIZE 2.077,4.500,0.667 ;
DEFAULT 1 ;
FONT "MS Sans Serif", 8 ;
STYLE "B" ;
VALID btn_val('END') ;
MESSAGE 'Go to last record.'
@ 15.846,36.333 GET m.add_btn ;
PICTURE "@*BHN " + ;
```

```
(LOCFILE("\fpw26\wizards\wizbmps\wznew.bmp", "BMP|ICO|PCT|ICN", "Where
is wznew?"));
SIZE 2.077,4.500,0.667 ;
DEFAULT 1 ;
FONT "MS Sans Serif", 8 ;
STYLE "B" ;
VALID btn_val('ADD') ;
MESSAGE 'Add a new record.'
@ 15.846,42.333 GET m.edit_btn ;
PICTURE "@*BHN " + ;
```

```
(LOCFILE("\fpw26\wizards\wizbmps\wzedit.bmp", "BMP|ICO|PCT|ICN", "Where
is wzedit?"));
SIZE 2.077,4.500,0.667 ;
DEFAULT 1 ;
FONT "MS Sans Serif", 8 ;
STYLE "B" ;
VALID btn_val('EDIT') ;
MESSAGE 'Edit current record.'
@ 15.846,48.333 GET m.del_btn ;
PICTURE "@*BHN " + ;
```

```
(LOCFILE("\fpw26\wizards\wizbmps\wzdelete.bmp", "BMP|ICO|PCT|ICN", "Where
is wzdelete?"));
SIZE 2.077,4.500,0.667 ;
DEFAULT 1 ;
FONT "MS Sans Serif", 8 ;
STYLE "B" ;
VALID btn_val('DELETE') ;
MESSAGE 'Delete current record.'
@ 15.846,54.333 GET m.save_btn ;
PICTURE "@*BHN " + ;
```



```

        (LOCFILE("\fpw26\wizards\wizbmps\wzsave.bmp", "BMP|ICO|PCT|ICN", "Where is wzsav?"));
        SIZE 2.077,4.500,0.667 ;
        DEFAULT 1 ;
        FONT "MS Sans Serif", 8 ;
        STYLE "B" ;
        VALID btn_val('SAVE') ;
        MESSAGE 'Save edits.'
@ 15.846,60.333 GET m.can_btn ;
        PICTURE "@*BHN " + ;

```

```

        (LOCFILE("\fpw26\wizards\wizbmps\wzundo.bmp", "BMP|ICO|PCT|ICN", "Where is wzundo?"));
        SIZE 2.077,4.500,0.667 ;
        DEFAULT 1 ;
        FONT "MS Sans Serif", 8 ;
        STYLE "B" ;
        VALID btn_val('CANCEL') ;
        MESSAGE 'Cancel edits.'
@ 15.846,66.333 GET m.prnt_btn ;
        PICTURE "@*BHN " + ;

```

```

        (LOCFILE("\fpw26\wizards\wizbmps\wzprint.bmp", "BMP|ICO|PCT|ICN", "Where is wzprint?"));
        SIZE 2.077,4.500,0.667 ;
        DEFAULT 1 ;
        FONT "MS Sans Serif", 8 ;
        STYLE "B" ;
        VALID btn_val('PRINT') ;
        MESSAGE 'Print report.'
@ 15.846,72.333 GET m.exit_btn ;
        PICTURE "@*BHN " + ;

```

```

        (LOCFILE("\fpw26\wizards\wizbmps\wzclose.bmp", "BMP|ICO|PCT|ICN", "Where is wzclose?"));
        SIZE 2.077,4.500,0.667 ;
        DEFAULT 1 ;
        FONT "MS Sans Serif", 8 ;
        STYLE "B" ;
        VALID btn_val('EXIT') ;
        MESSAGE 'Close screen.'

```

```

IF NOT WVISIBLE("_0471c7wft")
        ACTIVATE WINDOW _0471c7wft
ENDIF

```

```

* *****
*
* *   WindowsREAD contains clauses from SCREEN sel
*
* *****
*

```

```

READ CYCLE ;
    ACTIVATE READACT() ;
    DEACTIVATE READDEAC() ;
    NOLOCK

```

```

RELEASE WINDOW _0471c7wft

```

```

* *****
*
* *   Windows Closing Databases
*
* *****
*

```

```

IF USED("sell")
    SELECT sell
    USE
ENDIF

```

```

SELECT (m.currarea)

```

```

#REGION 0

```

```

SET READBORDER &rborder

```

```

IF m.talkstat = "ON"
    SET TALK ON
ENDIF
IF m.compstat = "ON"
    SET COMPATIBLE ON
ENDIF

```

```

* *****
*
* *   SEL/Windows Cleanup Code

```

```

*      *
*      *****
*

```

```

#REGION 1
SET DELETED &wzolddelete
SET FIELDS &wzfields
ON ERROR &wzolderror
ON KEY LABEL ESCAPE &wzoldesc
DO CASE
CASE _DOS AND SET('DISPLAY')='VGA25'
    @24,0 CLEAR TO 24,79
CASE _DOS AND SET('DISPLAY')='VGA50'
    @49,0 CLEAR TO 49,79
CASE _DOS
    @24,0 CLEAR TO 24,79
ENDCASE

```

****Procedures****

```

*      *****
*      *
*      *   SEL/Windows Supporting Procedures and Functions
*      *
*      *****
*

```

```

#REGION 1
PROCEDURE readdeac
    IF isediting
        ACTIVATE WINDOW '_0471c7wft'
        WAIT WINDOW C_EDIT$ NOWAIT
    ENDIF
    IF !WVISIBLE(WOUTPUT())
        CLEAR READ
        RETURN .T.
    ENDIF
    RETURN .F.

```

```

PROCEDURE readact
    IF !isediting
        SELECT (m.wzalias)
        SHOW GETS
    ENDIF
    DO REFRESH

```

RETURN

PROCEDURE wizerrorhandler

- * This very simple error handler is primarily intended
- * to trap for General field OLE errors which may occur
- * during editing from the MODIFY GENERAL window.

WAIT WINDOW message()

RETURN

PROCEDURE printrec

```
PRIVATE sOldError,wizfname,saverec,savearea,tmpcurs,tmpstr
PRIVATE prnt_btn,p_recs,p_output,pr_out,pr_record
STORE 1 TO p_recs,p_output
STORE 0 TO prnt_btn
STORE RECNO() TO saverec
m.sOldError=ON('error')
DO pdialog
IF m.prnt_btn = 2
  RETURN
ENDIF
IF !FILE(ALIAS()+'FRX')
  m.wizfname=SYS(2004)+'WIZARDS\'+'WIZARD.APP'
  IF !FILE(m.wizfname)
    ON ERROR *
    m.wizfname=LOCFILE('WIZARD.APP','APP',C_LOCWIZ)
    ON ERROR &sOldError
    IF !'WIZARD.APP'$UPPER(m.wizfname)
      WAIT WINDOW C_NOWIZ
      RETURN
    ENDIF
  ENDIF
ENDIF
WAIT WINDOW C_MAKEREPO NOWAIT
m.savearea=SELECT()
m.tmpcurs='_'+LEFT(SYS(3),7)
CREATE CURSOR (m.tmpcurs) (comment m)
m.tmpstr = '* LAYOUT = COLUMNAR'+CHR(13)+CHR(10)
INSERT INTO (m.tmpcurs) VALUES(m.tmpstr)
SELECT (m.savearea)
DO (m.wizfname) WITH
", 'WZ_QREPO', 'NOSCRN/CREATE', ALIAS(), m.tmpcurs
USE IN (m.tmpcurs)
WAIT CLEAR
IF !FILE(ALIAS()+'FRX') &&wizarc could not create report
WAIT WINDOW C_NOREPO
RETURN
```



```
ENDIF  
ENDIF
```

```
m.pr_out=IIF(m.p_output=1,'TO PRINT NOCONSOLE','PREVIEW')  
m.pr_record=IIF(m.p_recs=1,'NEXT 1','ALL')  
REPORT FORM (ALIAS()) &pr_out &pr_record  
GO m.saverec
```

```
RETURN
```

```
PROCEDURE BTN_VAL
```

```
PARAMETER m.btnname  
DO CASE  
CASE m.btnname='TOP'  
GO TOP  
WAIT WINDOW C_TOPFILE NOWAIT  
CASE m.btnname='PREV'  
IF !BOF()  
SKIP -1  
ENDIF  
IF BOF()  
WAIT WINDOW C_TOPFILE NOWAIT  
GO TOP  
ENDIF  
CASE m.btnname='NEXT'  
IF !EOF()  
SKIP 1  
ENDIF  
IF EOF()  
WAIT WINDOW C_ENDFILE NOWAIT  
GO BOTTOM  
ENDIF  
CASE m.btnname='END'  
GO BOTTOM  
WAIT WINDOW C_ENDFILE NOWAIT  
CASE m.btnname='LOCATE'  
DO loc_dlog  
CASE m.btnname='ADD' AND !isediting &&add record  
isediting=.T.  
isadding=.T.  
=edithand('ADD')  
_curobj=1  
DO refresh  
SHOW GETS  
RETURN  
CASE m.btnname='EDIT' AND !isediting &&edit record
```

```

IF EOF() OR BOF()
    WAIT WINDOW C_ENDFILE NOWAIT
    RETURN
ENDIF
IF RLOCK()
    isediting=.T.
    _curobj=1
    DO refresh
    RETURN
ELSE
    WAIT WINDOW C_NOLOCK
    RETURN
ENDIF
CASE m.btnname='SAVE' AND isediting &&save record
    IF isadding
        =edithand('SAVE')
    ELSE
        GATHER MEMVAR MEMO
    ENDIF
    UNLOCK
    isediting=.F.
    isadding=.F.
    DO refresh
CASE m.btnname='CANCEL' AND isediting &&cancel record
    IF isadding
        =edithand('CANCEL')
    ENDIF
    isediting=.F.
    isadding=.F.
    UNLOCK
    WAIT WINDOW C_ECANCEL NOWAIT
    DO refresh
CASE m.btnname='DELETE'
    IF EOF() OR BOF()
        WAIT WINDOW C_ENDFILE NOWAIT
        RETURN
    ENDIF
    IF fox_alert(C_DELREC)
        DELETE
        IF !EOF() AND DELETED()
            SKIP 1
        ENDIF
    IF EOF()
        WAIT WINDOW C_ENDFILE NOWAIT
        GO BOTTOM
    ENDIF

```

```

        ELSE
            RETURN
        ENDIF
CASE m.btnname='PRINT'
    DO printrec
    RETURN
CASE m.btnname='EXIT'
    m.bailout=.T. &&this is needed if used with FoxApp
    CLEAR READ
    RETURN
OTHERWISE
    RETURN
ENDCASE
SCATTER MEMVAR MEMO
SHOW GETS
RETURN
PROCEDURE REFRESH
DO CASE
CASE m.isreadonly AND RECCOUNT()=0
    SHOW GETS DISABLE
    SHOW GET exit_btn ENABLE
CASE m.isreadonly
    SHOW GET add_btn DISABLE
    SHOW GET edit_btn DISABLE
    SHOW GET del_btn DISABLE
    SHOW GET save_btn DISABLE
    SHOW GET can_btn DISABLE
CASE (RECCOUNT()=0 OR EOF()) AND !m.isediting
    SHOW GETS DISABLE
    SHOW GET add_btn ENABLE
    SHOW GET exit_btn ENABLE
CASE m.isediting
    SHOW GET find_drop DISABLE
    SHOW GET top_btn DISABLE
    SHOW GET prev_btn DISABLE
    SHOW GET loc_btn DISABLE
    SHOW GET next_btn DISABLE
    SHOW GET end_btn DISABLE
    SHOW GET add_btn DISABLE
    SHOW GET edit_btn,1 DISABLE
    SHOW GET del_btn,1 DISABLE
    SHOW GET prnt_btn DISABLE
    SHOW GET exit_btn DISABLE
    SHOW GET save_btn ENABLE
    SHOW GET can_btn ENABLE
ON KEY LABEL ESCAPE DO BTN_VAL WITH 'CANCEL'

```

```

    RETURN
OTHERWISE
    SHOW GET find_drop ENABLE
    SHOW GET top_btn ENABLE
    SHOW GET prev_btn ENABLE
    SHOW GET loc_btn ENABLE
    SHOW GET next_btn ENABLE
    SHOW GET end_btn ENABLE
    SHOW GET add_btn ENABLE
    SHOW GET edit_btn,1 ENABLE
    SHOW GET del_btn,1 ENABLE
    SHOW GET prnt_btn ENABLE
    SHOW GET exit_btn ENABLE
    SHOW GET save_btn DISABLE
    SHOW GET can_btn DISABLE
ENDCASE
IF m.is2table
    SHOW GET add_btn DISABLE
ENDIF
ON KEY LABEL ESCAPE
RETURN
PROCEDURE edithand
    PARAMETER m.paction
    DO CASE
    CASE m.paction = 'ADD'
        SCATTER MEMVAR MEMO BLANK
    CASE m.paction = 'SAVE'
        INSERT INTO (ALIAS()) FROM MEMVAR
    CASE m.paction = 'CANCEL'
        * nothing here
    ENDCASE
RETURN

PROCEDURE fox_alert
    PARAMETER wzalrtmess
    PRIVATE alrtbtn
    m.alrtbtn=2
    DEFINE WINDOW _qec1ij2t7 AT 0,0 SIZE 8,50 ;
    FONT "MS Sans Serif",10 STYLE 'B' ;
    FLOAT NOCLOSE NOMINIMIZE DOUBLE TITLE WTITLE()
    MOVE WINDOW _qec1ij2t7 CENTER
    ACTIVATE WINDOW _qec1ij2t7 NOSHOW
    @ 2,(50-txtwidth(wzalrtmess))/2 SAY wzalrtmess;
    FONT "MS Sans Serif", 10 STYLE "B"
    @ 6,18 GET m.alrtbtn ;
    PICTURE "@*HT \<OK;\?!\<Cancel" ;

```



```

    SIZE 1.769,8.667,1.333 ;
    FONT "MS Sans Serif", 8 STYLE "B"
    ACTIVATE WINDOW _qec1ij2t7
    READ CYCLE MODAL
    RELEASE WINDOW _qec1ij2t7
RETURN m.alrtbtn=1
PROCEDURE pdialog
    DEFINE WINDOW _qjn12zbvh ;
        AT 0.000, 0.000 ;
        SIZE 13.231,54.800 ;
        TITLE "Microsoft FoxPro" ;
        FONT "MS Sans Serif", 8 ;
        FLOAT NOCLOSE MINIMIZE SYSTEM
    MOVE WINDOW _qjn12zbvh CENTER
    ACTIVATE WINDOW _qjn12zbvh NOSHOW
    @ 2.846,33.600 SAY "Output:" ;
        FONT "MS Sans Serif", 8 ;
        STYLE "BT"
    @ 2.846,4.800 SAY "Print:" ;
        FONT "MS Sans Serif", 8 ;
        STYLE "BT"
    @ 4.692,7.200 GET m.p_recs ;
        PICTURE "@*RVN \<Current Record;\<All Records" ;
        SIZE 1.308,18.500,0.308 ;
        DEFAULT 1 ;
        FONT "MS Sans Serif", 8 ;
        STYLE "BT"
    @ 4.692,36.000 GET m.p_output ;
        PICTURE "@*RVN \<Printer;Pre\<view" ;
        SIZE 1.308,12.000,0.308 ;
        DEFAULT 1 ;
        FONT "MS Sans Serif", 8 ;
        STYLE "BT"
    @ 10.154,16.600 GET m.prnt_btn ;
        PICTURE "@*HT P\<rint;Ca\<ncel" ;
        SIZE 1.769,8.667,0.667 ;
        DEFAULT 1 ;
        FONT "MS Sans Serif", 8 ;
        STYLE "B"
    ACTIVATE WINDOW _qjn12zbvh
    READ CYCLE MODAL
    RELEASE WINDOW _qjn12zbvh
RETURN
PROCEDURE loc_dlog
    PRIVATE gfields,i
    DEFINE WINDOW wzlocate FROM 1,1 TO 20,40;

```

```

SYSTEM GROW CLOSE ZOOM FLOAT FONT "MS Sans Serif",8
MOVE WINDOW wzlocate CENTER
m.gfields=SET('FIELDS',2)
IF !EMPTY(RELATION(1))
    SET FIELDS ON
    IF m.gfields # 'GLOBAL'
        SET FIELDS GLOBAL
    ENDIF
    IF EMPTY(FLDLIST())
        m.i=1
        DO WHILE !EMPTY(OBJVAR(m.i))
            IF ATC('M.',OBJVAR(m.i))=0
                SET FIELDS TO (OBJVAR(m.i))
            ENDIF
            m.i = m.i + 1
        ENDDO
    ENDIF
ENDIF
BROWSE WINDOW wzlocate NOEDIT NODELETE ;
    NOMENU TITLE C_BRTITLE
SET FIELDS &gfields
SET FIELDS OFF
RELEASE WINDOW wzlocate
RETURN

```

```

FUNCTION price
CLOSE DATABASES
USE (LOCFILE("c:\ramiz\books.dbf","DBF","Where is books?"));
    SHARE;
    AGAIN ALIAS books ;
    ORDER TAG "_03y1drnzc"

SEEK m.title
IF FOUND()
m.price=price
ENDIF
show gets
CLOSE DATABASES
USE (LOCFILE("c:\ramiz\sell.dbf","dbf","WHERE IS sell ?"));
share ;
AGAIN ALIAS sell;
ORDER TAG "_04717qix2"

```

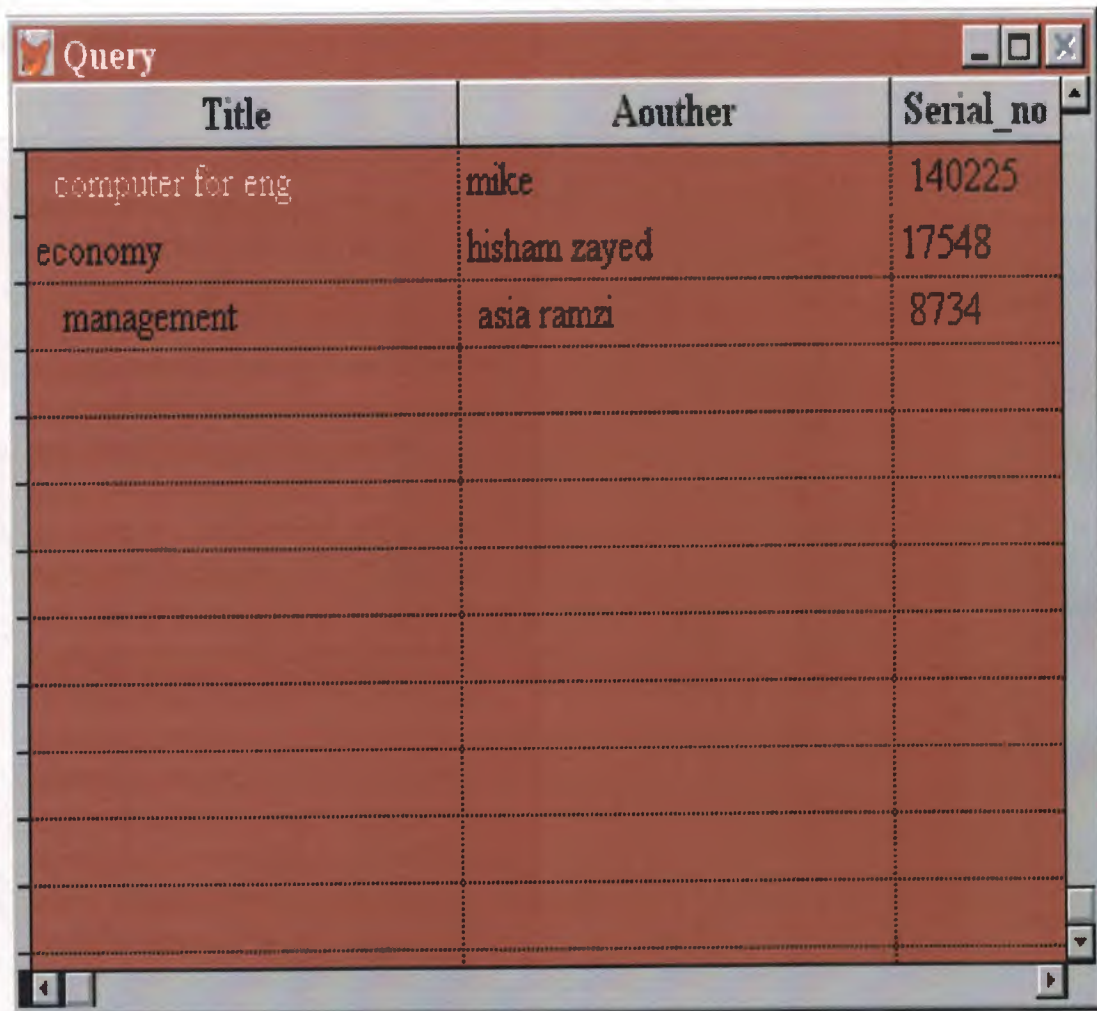
```

FUNCTION AA
m.total=m.price*m.copies
RETURN

```

LISTING QUERY

In this query we will have the general list of the bookstore, which allow the user to see what we have in the bookstore from books.



Title	Aauther	Serial_no
computer for eng	mike	140225
economy	hisham zayed	17548
management	asia ramzi	8734

```
SELECT *;  
FROM Books;  
WHERE Books.title = Books.title
```


EMPLOYEE FILE

This screen gives the necessary information about the employee that is enough in order to be a worker in our book store .It can add anew worker who want to work or delete the record that contains the fields like (name, surname, ID No, location, period, work hours sex, and the status of the employer).



The image shows a software window titled "Employee" with a sub-header "Employee File". The form contains several input fields and radio button options. The fields are arranged in two columns. The first column contains "Name:", "Surname:", "Location:", and "Period:". The second column contains "Id_no:", "Work_hours:", "Sex:", and "Status:". The "Sex:" and "Status:" fields have radio button options. At the bottom of the window is a toolbar with 12 icons: a left arrow, a right arrow, a magnifying glass, a left arrow, a right arrow, a document icon, a folder icon, a document with a red X, a floppy disk, a circular arrow, a printer, and a folder with a lightning bolt.

Name:	<input type="text" value="ramiz"/>	Id_no:	<input type="text" value="1021"/>
Surname:	<input type="text" value="tatary"/>	Work_hours:	<input type="text" value="08"/>
Location:	<input type="text" value="cach"/>	Sex:	<input checked="" type="radio"/> male <input type="radio"/> female
Period:	<input type="text" value="morning"/>	Status:	<input checked="" type="radio"/> single <input type="radio"/> married

Navigation icons: [Left Arrow] [Right Arrow] [Magnifying Glass] [Left Arrow] [Right Arrow] [Document] [Folder] [Document with X] [Floppy Disk] [Circular Arrow] [Printer] [Folder with Lightning Bolt]


```

* *****
*
* * 05/21/00      EMPLOYEE.SPR      23:06:47
*
* *****
*
* * Author's Name
*
* * Copyright (c) 2000 Company Name
* * Address
* * City, Zip
*
* * Description:
* * This program was automatically generated by GENSCRN.
*
* *****
* *****
*
* *      EMPLOYEE/Windows Setup Code - SECTION 1
*
* *****

```

#REGION 1

PRIVATE wzfields,wztalk

IF SET("TALK") = "ON"

SET TALK OFF

m.wztalk = "ON"

ELSE

m.wztalk = "OFF"

ENDIF

m.wzfields=SET('FIELDS')

SET FIELDS OFF

IF m.wztalk = "ON"

SET TALK ON

ENDIF

#REGION 0

REGIONAL m.currarea, m.talkstat, m.compstat

IF SET("TALK") = "ON"

SET TALK OFF

m.talkstat = "ON"

ELSE

m.talkstat = "OFF"

ENDIF

m.compstat = SET("COMPATIBLE")

SET COMPATIBLE FOXPLUS

```
m.rborder = SET("READBORDER")
SET READBORDER ON
```

```
m.currarea = SELECT()
```

```
* *****
*
* *   S3208083/Windows Databases, Indexes, Relations
* *
* *****
*
```

```
IF USED("employee")
    SELECT employee
    SET ORDER TO TAG "_03y1d1wqw"
ELSE
    SELECT 0
    USE (LOCFILE("\ramiz\employee.dbf","DBF","Where is employee?"));
    AGAIN ALIAS employee ;
    ORDER TAG "_03y1d1wqw"
ENDIF
```

```
* *****
*
* *   Windows Window definitions
* *
* *****
*
```

```
IF NOT WEXIST("_03y1djfro")
    DEFINE WINDOW _03y1djfro ;
        AT 0.000, 0.000 ;
        SIZE 14.550, 71.429 ;
        TITLE "Employee" ;
        FONT "Times New Roman", 12 ;
        STYLE "B" ;
        FLOAT ;
        CLOSE ;
        MINIMIZE ;
        COLOR RGB(,,128,0,0)
    MOVE WINDOW _03y1djfro CENTER
ENDIF
```

```

*      *****
*      *
*      *      EMPLOYEE/Windows Setup Code - SECTION 2
*      *
*      *****
*
#REGION 1

#DEFINE C_DBFEMPTY      'Database is empty, add a record?'
#DEFINE C_EDITS          'Please finish your edits.'
#DEFINE C_TOPFILE        'Top of file.'
#DEFINE C_ENDFILE        'End of file.'
#DEFINE C_BRTITLE        'Locate Record'
#DEFINE C_NOLOCK         'Sorry, could not lock record -- try again later.'
#DEFINE C_ECANCEL        'Edits Canceled.'
#DEFINE C_DELREC         'Delete selected record?'
#DEFINE C_NOFEAT         'Feature not available yet.'
#DEFINE C_NOWIZ          'Wizard application is not available.'
#DEFINE C_MAKEREPO       'Creating report with Report Wizard.'
#DEFINE C_NOREPO         'Could not create report.'
#DEFINE C_DELNOTE        'Deleting records...'
#DEFINE C_READONLY       'Table is read-only. No editing allowed.'
#DEFINE C_NOTABLE        'No table selected. Open table or run query.'
#DEFINE C_BADEXP         'Invalid expression.'
#DEFINE C_LOCWIZ         'Locate WIZARD.APP:'
#DEFINE C_MULTITABLE     'You have multiple related tables. Adding records in not
allowed.'

MOVE WINDOW '_03y1djfro' CENTER
PRIVATE isediting,isadding,wztblarr
PRIVATE wzolddelete,wzolderror,wzoldesc
PRIVATE wzalias, tempcurs,wzlastrec
PRIVATE isreadonly,find_drop,is2table

IF EMPTY(ALIAS())
    WAIT WINDOW C_NOTABLE
    RETURN
ENDIF

m.wztblarr= "
m.wzalias=SELECT()
m.isediting=.F.
m.isadding=.F.
m.is2table = .F.
m.wzolddelete=SET('DELETE')

```

```

SET DELETED ON
m.tempcurs=SYS(2015) &&used if General field
m.wzlastrec = 1
m.wzolderror=ON('error')
ON ERROR DO wizerrorhandler
wzoldesc=ON('KEY','ESCAPE')
ON KEY LABEL ESCAPE
m.find_drop = IIF(_DOS,0,2)

m.isreadonly=IIF(ISREAD(),.T.,.F.)
IF m.isreadonly
    WAIT WINDOW C_READONLY TIMEOUT 1
ENDIF

IF RECCOUNT()=0 AND !m.isreadonly AND fox_alert(C_DBFEMPTY)
    APPEND BLANK
ENDIF

GOTO TOP
SCATTER MEMVAR MEMO

*      *****
*      *
*      *      EMPLOYEE/Windows Screen Layout
*      *
*      *****
*

#REGION 1
IF WVISIBLE("_03y1djfro")
    ACTIVATE WINDOW _03y1djfro SAME
ELSE
    ACTIVATE WINDOW _03y1djfro NOSHOW
ENDIF
@ 1.750,0.000 TO 1.750,71.429 ;
    PEN 2, 8 ;
    STYLE "1"
@ 3.050,3.714 SAY "Name:" ;
    SIZE 0.938,12.000 ;
    FONT "Times New Roman", 10 ;
    STYLE "BT" ;
    PICTURE "@J" ;
    COLOR RGB(255,255,255,255,255,255)
@ 3.050,16.286 GET m.name ;
    SIZE 0.938,17.500 ;

```



```

DEFAULT " " ;
FONT "Times New Roman", 10 ;
STYLE "B" ;
PICTURE "@K XXXXXXXXXXXXXXXXXXXX" ;
WHEN isediting ;
COLOR ,RGB(0,0,0,255,255,255)
@ 4.900,3.714 SAY "Surname:" ;
SIZE 0.938,12.000 ;
FONT "Times New Roman", 10 ;
STYLE "BT" ;
PICTURE "@J" ;
COLOR RGB(255,255,255,255,255,255)
@ 4.900,16.286 GET m.surname ;
SIZE 0.938,17.500 ;
DEFAULT " " ;
FONT "Times New Roman", 10 ;
STYLE "B" ;
PICTURE "@K XXXXXXXXXXXXXXXXXXXX" ;
WHEN isediting ;
COLOR ,RGB(0,0,0,255,255,255)
@ 3.250,40.857 SAY "Id_no:" ;
SIZE 0.938,12.000 ;
FONT "Times New Roman", 10 ;
STYLE "BT" ;
PICTURE "@J" ;
COLOR RGB(255,255,255,255,255,255)
@ 3.250,53.429 GET m.id_no ;
SIZE 0.938,13.333 ;
DEFAULT " " ;
FONT "Times New Roman", 10 ;
STYLE "B" ;
PICTURE "@K XXXXXXXXXXXX" ;
WHEN isediting ;
COLOR ,RGB(0,0,0,255,255,255)
@ 6.950,3.714 SAY "Location:" ;
SIZE 0.938,12.000 ;
FONT "Times New Roman", 10 ;
STYLE "BT" ;
PICTURE "@J" ;
COLOR RGB(255,255,255,255,255,255)
@ 6.950,16.286 GET m.location ;
SIZE 0.938,17.500 ;
DEFAULT " " ;
FONT "Times New Roman", 10 ;
STYLE "B" ;
PICTURE "@K XXXXXXXXXXXXXXXXXXXX" ;

```

WHEN isediting ;
 COLOR ,RGB(0,0,0,255,255,255)
 @ 4.850,41.286 SAY "Work_hours:" ;
 SIZE 0.938,12.000 ;
 FONT "Times New Roman", 10 ;
 STYLE "BT" ;
 PICTURE "@J" ;
 COLOR RGB(255,255,255,255,255,255)
 @ 4.850,53.857 GET m.work_hours ;
 SIZE 0.938,13.333 ;
 DEFAULT " " ;
 FONT "Times New Roman", 10 ;
 STYLE "B" ;
 PICTURE "@K XXXXXXXXXXXXX" ;
 WHEN isediting ;
 COLOR ,RGB(0,0,0,255,255,255)
 @ 9.100,3.714 SAY "Period:" ;
 SIZE 0.938,12.000 ;
 FONT "Times New Roman", 10 ;
 STYLE "BT" ;
 PICTURE "@J" ;
 COLOR RGB(255,255,255,255,255,255)
 @ 9.100,16.286 GET m.period ;
 SIZE 0.938,17.500 ;
 DEFAULT " " ;
 FONT "Times New Roman", 10 ;
 STYLE "B" ;
 PICTURE "@K XXXXXXXXXXXXXXXXXXXX" ;
 WHEN isediting ;
 COLOR ,RGB(0,0,0,255,255,255)
 @ 6.950,40.857 SAY "Sex:" ;
 SIZE 0.938,12.000 ;
 FONT "Times New Roman", 10 ;
 STYLE "BT" ;
 PICTURE "@J" ;
 COLOR RGB(255,255,255,255,255,255)
 @ 9.000,40.857 SAY "Status:" ;
 SIZE 0.938,12.000 ;
 FONT "Times New Roman", 10 ;
 STYLE "BT" ;
 PICTURE "@J" ;
 COLOR RGB(255,255,255,255,255,255)
 @ 12.050,0.000 TO 12.050,71.429 ;
 PEN 2, 8 ;
 STYLE "1"
 @ 12.700,5.429 GET m.top_btn ;

```

PICTURE "@*BHN " + ;

(LOCFILE("\fpw26\wizards\wizbmps\wztop.bmp", "BMP|ICO|PCT|ICN", "Where
is wztop?")) ;
SIZE 1.350,3.857,0.571 ;
DEFAULT 1 ;
FONT "Times New Roman", 12 ;
STYLE "B" ;
VALID btn_val("TOP") ;
MESSAGE 'Go to first record.'
@ 12.700,10.571 GET m.prev_btn ;
PICTURE "@*BHN " + ;

(LOCFILE("\fpw26\wizards\wizbmps\wzback.bmp", "BMP|ICO|PCT|ICN", "Where
is wzback?")) ;
SIZE 1.350,3.857,0.571 ;
DEFAULT 1 ;
FONT "Times New Roman", 12 ;
STYLE "B" ;
VALID btn_val("PREV") ;
MESSAGE 'Go to previous record.'
@ 12.700,15.714 GET m.loc_btn ;
PICTURE "@*BHN " + ;

(LOCFILE("\fpw26\wizards\wizbmps\wzlocate.bmp", "BMP|ICO|PCT|ICN", "Where
is wzlocate?")) ;
SIZE 1.350,3.857,0.571 ;
DEFAULT 1 ;
FONT "Times New Roman", 12 ;
STYLE "B" ;
VALID btn_val("LOCATE") ;
MESSAGE 'Locate a record.'
@ 12.700,20.857 GET m.next_btn ;
PICTURE "@*BHN " + ;

(LOCFILE("\fpw26\wizards\wizbmps\wznnext.bmp", "BMP|ICO|PCT|ICN", "Where
is wznext?")) ;
SIZE 1.350,3.857,0.571 ;
DEFAULT 1 ;
FONT "Times New Roman", 12 ;
STYLE "B" ;
VALID btn_val("NEXT") ;
MESSAGE 'Go to next record.'
@ 12.700,26.000 GET m.end_btn ;
PICTURE "@*BHN " + ;

```

(LOCFILE("\\fpw26\\wizards\\wizbmps\\wzend.bmp", "BMP|ICO|PCT|ICN", "Where is wzend?"));
SIZE 1.350,3.857,0.571 ;
DEFAULT 1 ;
FONT "Times New Roman", 12 ;
STYLE "B" ;
VALID btn_val('END') ;
MESSAGE 'Go to last record.'
@ 12.700,31.143 GET m.add_btn ;
PICTURE "@*BHN " + ;

(LOCFILE("\\fpw26\\wizards\\wizbmps\\wznew.bmp", "BMP|ICO|PCT|ICN", "Where is wznew?"));
SIZE 1.350,3.857,0.571 ;
DEFAULT 1 ;
FONT "Times New Roman", 12 ;
STYLE "B" ;
VALID btn_val('ADD') ;
MESSAGE 'Add a new record.'
@ 12.700,36.286 GET m.edit_btn ;
PICTURE "@*BHN " + ;

(LOCFILE("\\fpw26\\wizards\\wizbmps\\wzedit.bmp", "BMP|ICO|PCT|ICN", "Where is wzedit?"));
SIZE 1.350,3.857,0.571 ;
DEFAULT 1 ;
FONT "Times New Roman", 12 ;
STYLE "B" ;
VALID btn_val('EDIT') ;
MESSAGE 'Edit current record.'
@ 12.700,41.429 GET m.del_btn ;
PICTURE "@*BHN " + ;

(LOCFILE("\\fpw26\\wizards\\wizbmps\\wzdelete.bmp", "BMP|ICO|PCT|ICN", "Where is wzdelete?"));
SIZE 1.350,3.857,0.571 ;
DEFAULT 1 ;
FONT "Times New Roman", 12 ;
STYLE "B" ;
VALID btn_val('DELETE') ;
MESSAGE 'Delete current record.'
@ 12.700,46.571 GET m.save_btn ;
PICTURE "@*BHN " + ;


```

(LOCFILE("\fpw26\wizards\wizbmps\wzsave.bmp","BMP|ICO|PCT|ICN","Where is wzsave?"));
SIZE 1.350,3.857,0.571 ;
DEFAULT 1 ;
FONT "Times New Roman", 12 ;
STYLE "B" ;
VALID btn_val('SAVE') ;
MESSAGE 'Save edits.'
@ 12.700,51.714 GET m.can_btn ;
PICTURE "@*BHN " + ;

```

```

(LOCFILE("\fpw26\wizards\wizbmps\wzundo.bmp","BMP|ICO|PCT|ICN","Where is wzundo?"));
SIZE 1.350,3.857,0.571 ;
DEFAULT 1 ;
FONT "Times New Roman", 12 ;
STYLE "B" ;
VALID btn_val('CANCEL') ;
MESSAGE 'Cancel edits.'
@ 12.700,56.857 GET m.prnt_btn ;
PICTURE "@*BHN " + ;

```

```

(LOCFILE("\fpw26\wizards\wizbmps\wzprint.bmp","BMP|ICO|PCT|ICN","Where is wzprint?"));
SIZE 1.350,3.857,0.571 ;
DEFAULT 1 ;
FONT "Times New Roman", 12 ;
STYLE "B" ;
VALID btn_val('PRINT') ;
MESSAGE 'Print report.'
@ 12.700,62.000 GET m.exit_btn ;
PICTURE "@*BHN " + ;

```

```

(LOCFILE("\fpw26\wizards\wizbmps\wzclose.bmp","BMP|ICO|PCT|ICN","Where is wzclose?"));
SIZE 1.350,3.857,0.571 ;
DEFAULT 1 ;
FONT "Times New Roman", 12 ;
STYLE "B" ;
VALID btn_val('EXIT') ;
MESSAGE 'Close screen.'
@ 0.450,27.143 SAY "Employee File" ;
FONT "Times New Roman", 12 ;
STYLE "BT" ;
COLOR RGB(255,255,255,,,)

```

```

@ 6.650,56.571 GET m.sex ;
  PICTURE "@*RVN male;female" ;
  SIZE 1.308,11.500,0.308 ;
  DEFAULT 1 ;
  FONT "MS Sans Serif", 8 ;
  STYLE "BT" ;
  WHEN isediting ;
  COLOR ,,,,,,RGB(255,255,255,,,,),RGB(255,255,255,,,,)

@ 9.050,56.571 GET m.status ;
  PICTURE "@*RVN single;married" ;
  SIZE 1.308,11.333,0.308 ;
  DEFAULT 1 ;
  FONT "MS Sans Serif", 8 ;
  STYLE "BT" ;
  WHEN isediting ;
  COLOR ,,,,,,RGB(255,255,255,,,,),RGB(255,255,255,,,,)

```

```

IF NOT WVISIBLE("_03y1djfro")
  ACTIVATE WINDOW _03y1djfro
ENDIF

```

```

* *****
* *
* * WindowsREAD contains clauses from SCREEN s3208083
* *
* *****
*

```

```

READ CYCLE ;
  ACTIVATE READACT() ;
  DEACTIVATE READDEAC() ;
  NOLOCK

```

```

RELEASE WINDOW _03y1djfro

```

```

* *****
* *
* * Windows Closing Databases
* *
* *****
*

```

```

IF USED("employee")
  SELECT employee
  USE

```

ENDIF

SELECT (m.currarea)

#REGION 0

SET READBORDER &rborder

IF m.talkstat = "ON"

SET TALK ON

ENDIF

IF m.compstat = "ON"

SET COMPATIBLE ON

ENDIF

```
* *****
*
*      *      EMPLOYEE/Windows Cleanup Code
*      *
*      *****
*
```

#REGION 1

SET DELETED &wzolddelete

SET FIELDS &wzfields

ON ERROR &wzolderror

ON KEY LABEL ESCAPE &wzoldesc

DO CASE

CASE _DOS AND SET('DISPLAY')='VGA25'

@24,0 CLEAR TO 24,79

CASE _DOS AND SET('DISPLAY')='VGA50'

@49,0 CLEAR TO 49,79

CASE _DOS

@24,0 CLEAR TO 24,79

ENDCASE

****Procedures****

```
* *****
*
*      *      EMPLOYEE/Windows Supporting Procedures and Functions
*      *
*      *****
```

*

#REGION 1

PROCEDURE readdeac

IF isediting

 ACTIVATE WINDOW '_03y1djfro'

 WAIT WINDOW C_EDITS NOWAIT

ENDIF

IF !WVISIBLE(WOUTPUT())

 CLEAR READ

 RETURN .T.

ENDIF

RETURN .F.

PROCEDURE readact

IF !isediting

 SELECT (m.wzalias)

 SHOW GETS

ENDIF

DO REFRESH

RETURN

PROCEDURE wizerrorhandler

 * This very simple error handler is primarily intended

 * to trap for General field OLE errors which may occur

 * during editing from the MODIFY GENERAL window.

 WAIT WINDOW message()

RETURN

PROCEDURE printrec

 PRIVATE sOldError,wizfname,saverec,savearea,tmpcurs,tmpstr

 PRIVATE prnt_btn,p_recs,p_output,pr_out,pr_record

 STORE 1 TO p_recs,p_output

 STORE 0 TO prnt_btn

 STORE RECNO() TO saverec

 m.sOldError=ON('error')

 DO pdialog

 IF m.prnt_btn = 2

 RETURN

 ENDIF

 IF !FILE(ALIAS()+'.FRX')

 m.wizfname=SYS(2004)+'WIZARDS\'+ 'WIZARD.APP'

 IF !FILE(m.wizfname)

 ON ERROR *

 m.wizfname=LOCFILE('WIZARD.APP','APP',C_LOCWIZ)


```

        ON ERROR &sOldError
        IF !'WIZARD.APP'$UPPER(m.wizfname)
        WAIT WINDOW C_NOWIZ
        RETURN
        ENDIF
    ENDIF
WAIT WINDOW C_MAKEREPO NOWAIT
    m.savearea=SELECT()
    m.tmpcurs='_'+LEFT(SYS(3),7)
    CREATE CURSOR (m.tmpcurs) (comment m)
    m.tmpstr = '* LAYOUT = COLUMNAR'+CHR(13)+CHR(10)
    INSERT INTO (m.tmpcurs) VALUES(m.tmpstr)
    SELECT (m.savearea)
    DO (m.wizfname) WITH
    ", 'WZ_QREPO', 'NOSCRN/CREATE', ALIAS(), m.tmpcurs
    USE IN (m.tmpcurs)
WAIT CLEAR
    IF !FILE(ALIAS()+'FRX') &&wizard could not create report
    WAIT WINDOW C_NOREPO
    RETURN
    ENDIF
ENDIF

m.pr_out=IIF(m.p_output=1, 'TO PRINT NOCONSOLE', 'PREVIEW')
m.pr_record=IIF(m.p_recs=1, 'NEXT 1', 'ALL')
REPORT FORM (ALIAS()) &pr_out &pr_record
GO m.saverec
RETURN
PROCEDURE BTN_VAL
    PARAMETER m.btnname
    DO CASE
    CASE m.btnname='TOP'
        GO TOP
        WAIT WINDOW C_TOPFILE NOWAIT
    CASE m.btnname='PREV'
        IF !BOF()
            SKIP -1
        ENDIF
        IF BOF()
            WAIT WINDOW C_TOPFILE NOWAIT
            GO TOP
        ENDIF
    CASE m.btnname='NEXT'
        IF !EOF()
            SKIP 1
        ENDIF

```

```

        IF EOF()
            WAIT WINDOW C_ENDFILE NOWAIT
            GO BOTTOM
        ENDIF
CASE m.btnname='END'
    GO BOTTOM
    WAIT WINDOW C_ENDFILE NOWAIT
CASE m.btnname='LOCATE'
    DO loc_dlog
CASE m.btnname='ADD' AND !isediting &&add record
    isediting=.T.
    isadding=.T.
    =edithand('ADD')
    _curobj=1
    DO refresh
    SHOW GETS
    RETURN
CASE m.btnname='EDIT' AND !isediting &&edit record
    IF EOF() OR BOF()
        WAIT WINDOW C_ENDFILE NOWAIT
        RETURN
    ENDIF
    IF RLOCK()
        isediting=.T.
        _curobj=1
        DO refresh
        RETURN
    ELSE
        WAIT WINDOW C_NOLOCK
        RETURN
    ENDIF
CASE m.btnname='SAVE' AND isediting &&save record
    IF isadding
        =edithand('SAVE')
    ELSE
        GATHER MEMVAR MEMO
    ENDIF
    UNLOCK
    isediting=.F.
    isadding=.F.
    DO refresh
CASE m.btnname='CANCEL' AND isediting &&cancel record
    IF isadding
        =edithand('CANCEL')
    ENDIF
    isediting=.F.

```

```

        isadding=.F.
        UNLOCK
        WAIT WINDOW C_ECANCEL NOWAIT
        DO refresh
CASE m.btnname='DELETE'
    IF EOF() OR BOF()
        WAIT WINDOW C_ENDFILE NOWAIT
        RETURN
    ENDIF
    IF fox_alert(C_DELREC)
        DELETE
        IF !EOF() AND DELETED()
            SKIP 1
        ENDIF
        IF EOF()
            WAIT WINDOW C_ENDFILE NOWAIT
            GO BOTTOM
        ENDIF
    ELSE
        RETURN
    ENDIF
CASE m.btnname='PRINT'
    DO printrec
    RETURN
CASE m.btnname='EXIT'
    m.bailout=.T. &&this is needed if used with FoxApp
    CLEAR READ
    RETURN
OTHERWISE
    RETURN
ENDCASE
SCATTER MEMVAR MEMO
SHOW GETS
RETURN
PROCEDURE REFRESH
DO CASE
CASE m.isreadonly AND RECCOUNT()=0
    SHOW GETS DISABLE
    SHOW GET exit_btn ENABLE
CASE m.isreadonly
    SHOW GET add_btn DISABLE
    SHOW GET edit_btn DISABLE
    SHOW GET del_btn DISABLE
    SHOW GET save_btn DISABLE
    SHOW GET can_btn DISABLE
CASE (RECCOUNT()=0 OR EOF()) AND !m.isediting

```

```

    SHOW GETS DISABLE
    SHOW GET add_btn ENABLE
    SHOW GET exit_btn ENABLE
CASE m.isediting
    SHOW GET find_drop DISABLE
    SHOW GET top_btn DISABLE
    SHOW GET prev_btn DISABLE
    SHOW GET loc_btn DISABLE
    SHOW GET next_btn DISABLE
    SHOW GET end_btn DISABLE
    SHOW GET add_btn DISABLE
    SHOW GET edit_btn,1 DISABLE
    SHOW GET del_btn,1 DISABLE
    SHOW GET prnt_btn DISABLE
    SHOW GET exit_btn DISABLE
    SHOW GET save_btn ENABLE
    SHOW GET can_btn ENABLE
    ON KEY LABEL ESCAPE DO BTN_VAL WITH 'CANCEL'
    RETURN
OTHERWISE
    SHOW GET find_drop ENABLE
    SHOW GET top_btn ENABLE
    SHOW GET prev_btn ENABLE
    SHOW GET loc_btn ENABLE
    SHOW GET next_btn ENABLE
    SHOW GET end_btn ENABLE
    SHOW GET add_btn ENABLE
    SHOW GET edit_btn,1 ENABLE
    SHOW GET del_btn,1 ENABLE
    SHOW GET prnt_btn ENABLE
    SHOW GET exit_btn ENABLE
    SHOW GET save_btn DISABLE
    SHOW GET can_btn DISABLE
ENDCASE
IF m.is2table
    SHOW GET add_btn DISABLE
ENDIF
ON KEY LABEL ESCAPE
RETURN

PROCEDURE edithand
    PARAMETER m.paction
    DO CASE
    CASE m.paction = 'ADD'
        SCATTER MEMVAR MEMO BLANK
    CASE m.paction = 'SAVE'

```



```

        INSERT INTO (ALIAS()) FROM MEMVAR
CASE m.paction = 'CANCEL'
    * nothing here
ENDCASE
RETURN
PROCEDURE fox_alert
    PARAMETER wzalrtmess
    PRIVATE alrtbtn
    m.alrtbtn=2
        DEFINE WINDOW _qec1ij2t7 AT 0,0 SIZE 8,50 ;
        FONT "MS Sans Serif",10 STYLE 'B' ;
        FLOAT NOCLOSE NOMINIMIZE DOUBLE TITLE WTITLE()
        MOVE WINDOW _qec1ij2t7 CENTER
        ACTIVATE WINDOW _qec1ij2t7 NOSHOW
        @ 2,(50-txtwidth(wzalrtmess))/2 SAY wzalrtmess;
        FONT "MS Sans Serif", 10 STYLE "B"
        @ 6,18 GET m.alrtbtn ;
        PICTURE "@*HT \<OK;\?!\<Cancel" ;
        SIZE 1.769,8.667,1.333 ;
        FONT "MS Sans Serif", 8 STYLE "B"
        ACTIVATE WINDOW _qec1ij2t7
        READ CYCLE MODAL
        RELEASE WINDOW _qec1ij2t7
    RETURN m.alrtbtn=1
PROCEDURE pdialog
    DEFINE WINDOW _qjn12zbvh ;
        AT 0.000, 0.000 ;
        SIZE 13.231,54.800 ;
        TITLE "Microsoft FoxPro" ;
        FONT "MS Sans Serif", 8 ;
        FLOAT NOCLOSE MINIMIZE SYSTEM
    MOVE WINDOW _qjn12zbvh CENTER
    ACTIVATE WINDOW _qjn12zbvh NOSHOW
    @ 2.846,33.600 SAY "Output:" ;
        FONT "MS Sans Serif", 8 ;
        STYLE "BT"
    @ 2.846,4.800 SAY "Print:" ;
        FONT "MS Sans Serif", 8 ;
        STYLE "BT"
    @ 4.692,7.200 GET m.p_recs ;
        PICTURE "@*RVN \<Current Record;\<All Records" ;
        SIZE 1.308,18.500,0.308 ;
        DEFAULT 1 ;
        FONT "MS Sans Serif", 8 ;
        STYLE "BT"
    @ 4.692,36.000 GET m.p_output ;

```

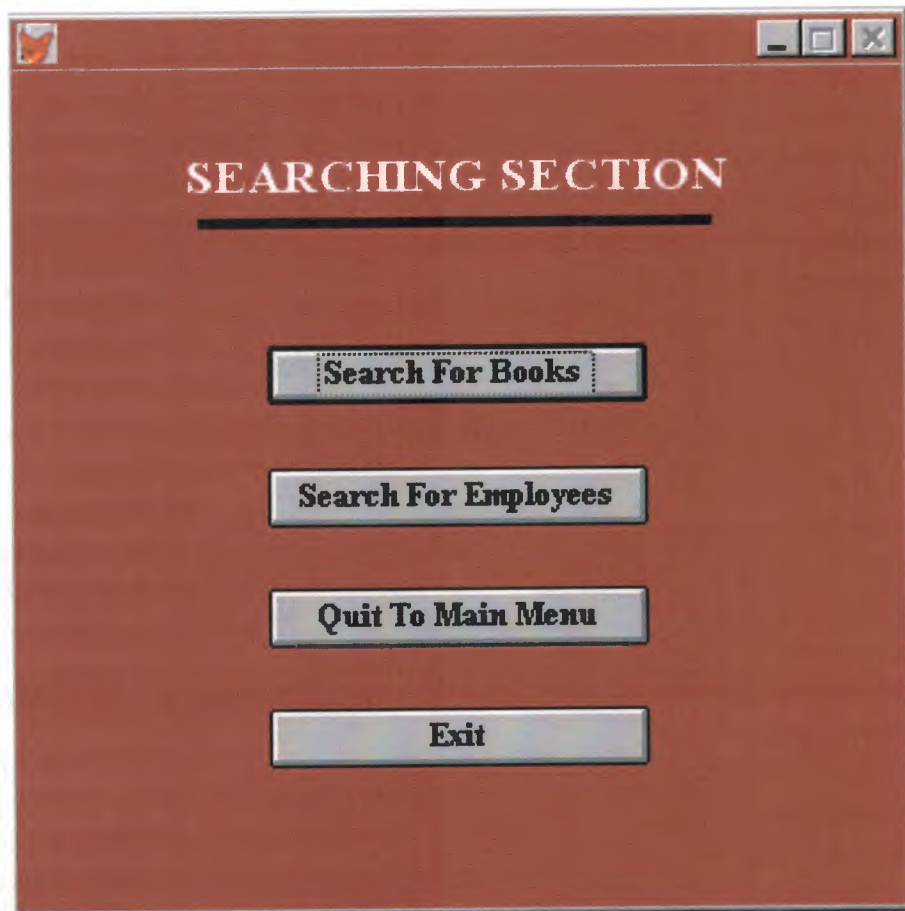
```

        PICTURE "@*RVN \<Printer;Pre\<view" ;
        SIZE 1.308,12.000,0.308 ;
        DEFAULT 1 ;
        FONT "MS Sans Serif", 8 ;
        STYLE "BT"
    @ 10.154,16.600 GET m.prnt_btn ;
        PICTURE "@*HT P\<rint;Ca\<ncel" ;
        SIZE 1.769,8.667,0.667 ;
        DEFAULT 1 ;
        FONT "MS Sans Serif", 8 ;
        STYLE "B"
    ACTIVATE WINDOW _qjn12zbvh
    READ CYCLE MODAL
    RELEASE WINDOW _qjn12zbvh
RETURN
PROCEDURE loc_dlog
    PRIVATE gfields,i
    DEFINE WINDOW wzlocate FROM 1,1 TO 20,40;
        SYSTEM GROW CLOSE ZOOM FLOAT FONT "MS Sans Serif",8
    MOVE WINDOW wzlocate CENTER
    m.gfields=SET('FIELDS',2)
    IF !EMPTY(RELATION(1))
        SET FIELDS ON
        IF m.gfields # 'GLOBAL'
            SET FIELDS GLOBAL
        ENDIF
        IF EMPTY(FLDLIST())
            m.i=1
            DO WHILE !EMPTY(OBJVAR(m.i))
                IF ATC('M.',OBJVAR(m.i))=0
                    SET FIELDS TO (OBJVAR(m.i))
                ENDIF
                m.i = m.i + 1
            ENDDO
        ENDIF
    ENDIF
    BROWSE WINDOW wzlocate NOEDIT NODELETE ;
        NOMENU TITLE C_BRTITLE
    SET FIELDS &gfields
    SET FIELDS OFF
    RELEASE WINDOW wzlocate
RETURN

```

SEARCHING SECTION

Here we used anew methods for searching menu which includes tow sections (book title and employee) we gave the user easier way to get his/her information directly by searching menu.



SET TALK OFF

```
IF NOT WEXIST("_03z1g50aa")
    DEFINE WINDOW _03z1g50aa ;
        AT 0.000, 0.000 ;
        SIZE 20.438,67.800 ;
        FONT "Times New Roman", 10 ;
        FLOAT ;
        CLOSE ;
        MINIMIZE ;
        SYSTEM ;
        COLOR RGB(,,128,0,0)
        move window _03z1g50aa center
ENDIF
```

```
* *****
*
*      SEARCH/Windows Screen Layout
*
* *****
*
```

```
#REGION 1
IF WVISIBLE("_03z1g50aa")
    ACTIVATE WINDOW _03z1g50aa SAME
ELSE
    ACTIVATE WINDOW _03z1g50aa NOSHOW
ENDIF
@ 6.750,19.400 GET a ;
    PICTURE "@*HN Search For Books " ;
    SIZE 1.438,24.167,0.667 ;
    DEFAULT 1 ;
    FONT "Times New Roman", 10 ;
    STYLE "B";
    valid bt('pro1')
@ 9.750,19.400 GET a ;
    PICTURE "@*HN Search For Employees" ;
    SIZE 1.438,24.167,0.667 ;
    DEFAULT 1 ;
    FONT "Times New Roman", 10 ;
    STYLE "B";
    valid bt('pro2')
@ 12.688,19.400 GET a ;
    PICTURE "@*HN Quit To Main Menu" ;
    SIZE 1.438,24.167,0.667 ;
```



```

DEFAULT 1 ;
FONT "Times New Roman", 10 ;
STYLE "B";
valid bt('pro3')
@ 15.625,19.400 GET M.BT_EXIT ;
PICTURE "@*HN Exit" ;
SIZE 1.438,24.167,0.667 ;
DEFAULT 1 ;
FONT "Times New Roman", 10 ;
STYLE "B" ;
VALID bt('EXIT')
@ 1.938,13.200 SAY "SEARCHING SECTION" ;
FONT "Times New Roman", 14 ;
STYLE "BT" ;
COLOR RGB(255,255,255,,,)
@ 3.625,14.200 TO 3.625,53.400 ;
PEN 4, 8 ;
STYLE "1"

```

```

IF NOT WVISIBLE("_03z1g50aa")
    ACTIVATE WINDOW _03z1g50aa
ENDIF
READ CYCLE
PROCEDURE BT
    PARAMETER m.btnname
    DO CASE
CASE m.btnname='pro1'
    Do c:\ramiz\serbook.spr
CASE m.btnname='pro2'
    DO c:\ramiz\seremp.spr
CASE m.btnname='pro3'
    DO c:\ramiz\menu.spr
CASE m.btnname='EXIT'
set sysm to defa
DEACTIVATE WINDOW _03z1g50aa
RELEASE WINDOW _03z1g50aa
    ENDCASE
RETURN

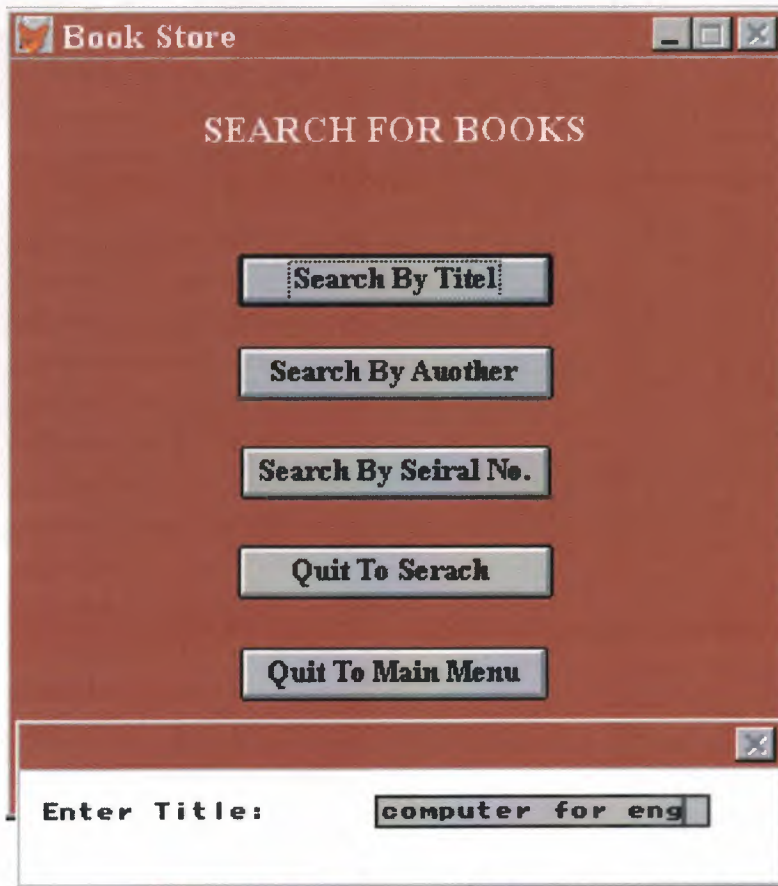
```

SEARCHING FOR BOOKS

Here we used some methods for searching for the books by title, author and serial number. This screen uses to find the specific books which he /she wants all information about the book, for example we want to find the book by title, steps are

1. Click the title button.
2. Enter the title name.
3. Press the enter key to find the book.

The screens of the searching a book by title are shown as the following screen menu:



The screenshot shows a window titled "Book Store" with a red background. The main heading is "SEARCH FOR BOOKS". Below it are five buttons: "Search By Titel", "Search By Auother", "Search By Seiral No.", "Quit To Serach", and "Quit To Main Menu". At the bottom, there is a text input field labeled "Enter Title:" containing the text "computer for eng".


```

* *****
*
* *   SERBOOK/Windows Databases, Indexes, Relations
* *
* *****
*
clear all

IF USED("books")
    SELECT books
    SET ORDER TO TAG "_03y1drnzc"
ELSE
    SELECT 0
    USE (LOCFILE("books.dbf","DBF","Where is books?"));
    AGAIN ALIAS books ;
    ORDER TAG "_03y1drnzc"
ENDIF

* *****
*
* *   Windows Window definitions
* *
* *****
*

IF NOT WEXIST("_03y1f7ah2")
    DEFINE WINDOW _03y1f7ah2 ;
        AT 0.000, 0.000 ;
        SIZE 20.250,54.167 ;
        TITLE "Book Store";
        FONT "Times New Roman", 10 ;
        STYLE "B" ;
        FLOAT ;
        CLOSE ;
        MINIMIZE ;
        SYSTEM ;
        COLOR RGB(,,128,0,0)
        move window _03y1f7ah2 center
ENDIF

* *****
*
* *   SERBOOK/Windows Screen Layout
* *
* *****
*
#REGION 1

```



```

IF WVISIBLE("_03y1f7ah2")
    ACTIVATE WINDOW _03y1f7ah2 SAME
ELSE
    ACTIVATE WINDOW _03y1f7ah2 NOSHOW
ENDIF
@ 5.250,16.000 GET a ;
    PICTURE "@*HN Search By Titel" ;
    SIZE 1.438,22.167,0.667 ;
    DEFAULT 1 ;
    FONT "Times New Roman", 10 ;
    STYLE "B";
    valid bt('pro1')
@ 7.750,16.000 GET a ;
    PICTURE "@*HN Search By Auother" ;
    SIZE 1.438,22.167,0.667 ;
    DEFAULT 1 ;
    FONT "Times New Roman", 10 ;
    STYLE "B";
    valid bt('pro2')
@ 10.438,16.167 GET a ;
    PICTURE "@*HN Search By Seiral No." ;
    SIZE 1.438,21.833,0.667 ;
    DEFAULT 1 ;
    FONT "Times New Roman", 10 ;
    STYLE "B";
    valid bt('pro3')
@ 13.125,16.000 GET a ;
    PICTURE "@*HN Quit To Serach " ;
    SIZE 1.438,22.167,0.667 ;
    DEFAULT 1 ;
    FONT "Times New Roman", 10 ;
    STYLE "B";
    valid bt('pro4')
@ 15.875,16.167 GET a ;
    PICTURE "@*HN Quit To Main Menu" ;
    SIZE 1.438,21.667,0.667 ;
    DEFAULT 1 ;
    FONT "Times New Roman", 10 ;
    STYLE "B";
    valid bt('pro5')
@ 1.313,13.500 SAY "SEARCH FOR BOOKS" ;
    FONT "Times New Roman", 12 ;
    STYLE "BT" ;
    COLOR RGB(255,255,255,,,)

IF NOT WVISIBLE("_03y1f7ah2")

```

```

        ACTIVATE WINDOW _03y1f7ah2
ENDIF

READ CYCLE
PROCEDURE BT
    PARAMETER m.btnname
    DO CASE
CASE m.btnname='pro1'
    Do c:\ramiz\title.prg
CASE m.btnname='pro2'
    DO c:\ramiz\auther.prg
CASE m.btnname='pro3'
    DO c:\ramiz\serial.prg
CASE m.btnname='pro4'
    DO c:\ramiz\search.spr
CASE m.btnname='pro5'
    DO c:\ramiz\menu.spr
    RELEASE WINDOW _03y1f7ah2
    ENDCASE
RETURN

close data base
do find
procedure find
define window a1 at 12,12 size 4,40 system
activate window a1
your_ch=space(17)
@1,1 say "Enter Title:"
@1,19 get your_ch
read
release window a1
use c:\ramiz\books.dbf
locate all for alltrim(title)=alltrim(your_ch)
if found() then
    WAIT WINDOW " Searching is OK " NOWAIT
    browse fields;
    title,auther,serial_no,shelf_no,price,copies,edition :H=' ';

else
    WAIT WINDOW " not found " NOWAIT
    WAIT WINDOW " press enter to exit " NOWAIT
endif
RETURN

close data base
do find

```

```

procedure find

define window a1 at 12,12 size 4,40 system
activate window a1
your_ch=space(8)
@1,1 say "Enter Aouther:"
@1,19 get your_ch
read
release window a1
use c:\ramiz\books.dbf
locate all for alltrim(aouther)=alltrim(your_ch)
if found() then
    WAIT WINDOW " Searching is OK " NOWAIT
    browse fields;
    aouther,serial_no,title,shelf_no,price,copies,edition :H=' ';

else
    WAIT WINDOW " not found " NOWAIT
    WAIT WINDOW " press enter to exit " NOWAIT
endif
return

close data base
do find

procedure find
define window a1 at 12,12 size 4,40 system
activate window a1
your_ch=space(8)
@1,1 say "Enter serial no:"
@1,19 get your_ch
read
release window a1
use c:\ramiz\books.dbf
locate all for alltrim(serial_no)=alltrim(your_ch)
if found() then
    WAIT WINDOW " Searching is OK " NOWAIT
    browse fields;
    serial_no,title,aouther,shelf_no,price,copies,edition :H=' ';

else
    WAIT WINDOW " not found " NOWAIT
    WAIT WINDOW " press enter to exit " NOWAIT
endif
return

close data base
do find

```

```

procedure find
define window a1 at 12,12 size 4,40 system
activate window a1
your_ch=space(17)
@1,1 say "Enter Title:"
@1,19 get your_ch
read
release window a1
use c:\ramiz\books.dbf
locate all for alltrim(title)=alltrim(your_ch)
if found() then
    WAIT WINDOW " Searching is OK " NOWAIT
    browse fields;
    title,aouther,serial_no,shelf_no,price,copies,edition :H=' ';
else
    WAIT WINDOW " not found " NOWAIT
    WAIT WINDOW " press enter to exit " NOWAIT
endif
RETURN

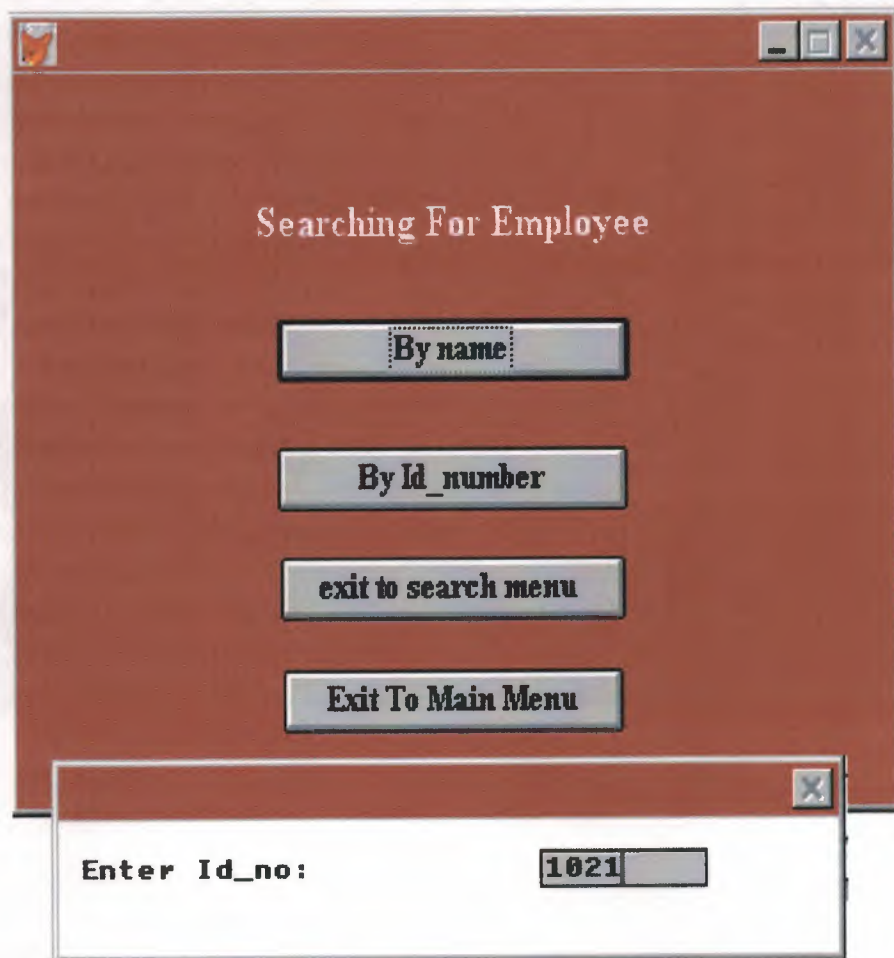
```


SEARCHING FOR EMPLOYEE MENU

Here we used some methods for searching for the books by employee, employee and serial number. This screen uses to find the specific books which he /she wants all information about the book, for example we want to find the book by employee, steps are:

4. By name.
5. By ID-No.
6. Press the enter key to find the employee.

The screens of the searching a book by title are shown as the following screen menu:



Searching For Employee

By name

By Id_number

exit to search menu

Exit To Main Menu

Enter Id_no: 1021

```

* *****
*
* * 05/25/00          SEREMP.SPR          22:23:02
*
* *****
*
* * Author's Name
*
* * Copyright (c) 2000 Company Name
* * Address
* * City, Zip
*
* * Description:
* * This program was automatically generated by GENSCRN.
*
* *****

```

clear all

#REGION 0

REGIONAL m.currarea, m.talkstat, m.compstat

IF SET("TALK") = "ON"

SET TALK OFF

m.talkstat = "ON"

ELSE

m.talkstat = "OFF"

ENDIF

m.compstat = SET("COMPATIBLE")

SET COMPATIBLE FOXPLUS

m.rborder = SET("READBORDER")

SET READBORDER ON

m.currarea = SELECT()

```

* *****
*
* * SEREMP/Windows Databases, Indexes, Relations
*
* *****
*

```

IF USED("books")

SELECT books

```

        SET ORDER TO TAG "_03y1drnzc"
ELSE
    SELECT 0
    USE (LOCFILE("books.dbf","DBF","Where is books?"));
    AGAIN ALIAS books ;
    ORDER TAG "_03y1drnzc"
ENDIF

```

```

* *****
* *
* *      Windows Window definitions
* *
* *****
*

```

```

IF NOT WEXIST("_0421bz696")
    DEFINE WINDOW _0421bz696 ;
        AT 0.000, 0.000 ;
        SIZE 13.350,51.571 ;
        FONT "Times New Roman", 12 ;
        STYLE "B" ;
        FLOAT ;
        CLOSE ;
        MINIMIZE ;
        SYSTEM ;
        COLOR RGB(,,128,0,0)
        move window _0421bz696 center
ENDIF

```

```

* *****
* *
* *      SEREMP/Windows Screen Layout
* *
* *****

```

```

#REGION 1
IF WVISIBLE("_0421bz696")
    ACTIVATE WINDOW _0421bz696 SAME
ELSE
    ACTIVATE WINDOW _0421bz696 NOSHOW
ENDIF
@ 4.500,15.429 GET a ;
    PICTURE "@*HN By name" ;
    SIZE 1.438,24.167,0.667 ;
    DEFAULT 1 ;

```

```

    FONT "Times New Roman", 10 ;
    STYLE "B";
    valid bt('pro1')
@ 6.850,15.429 GET a ;
    PICTURE "@*HN By Id_number" ;
    SIZE 1.438,24.000,0.667 ;
    DEFAULT 1 ;
    FONT "Times New Roman", 10 ;
    STYLE "B";
    valid bt('pro2')
@ 8.850,15.571 GET a ;
    PICTURE "@*HN exit to search menu " ;
    SIZE 1.438,23.667,0.667 ;
    DEFAULT 1 ;
    FONT "Times New Roman", 10 ;
    STYLE "B";
    valid bt('pro3')
@ 10.900,15.714 GET a ;
    PICTURE "@*HN Exit To Main Menu" ;
    SIZE 1.438,23.333,0.667 ;
    DEFAULT 1 ;
    FONT "Times New Roman", 10 ;
    STYLE "B";
    valid bt('pro4')
@ 2.250,14.143 SAY "Searching For Employee" ;
    FONT "Times New Roman", 12 ;
    STYLE "BT" ;
    COLOR RGB(255,255,255,,,)
IF NOT WVISIBLE("_0421bz696")
    ACTIVATE WINDOW _0421bz696
ENDIF
READ CYCLE
PROCEDURE BT
    PARAMETER m.btnname
    DO CASE
CASE m.btnname='pro2'
    Do c:\ramiz\id.prg
CASE m.btnname='pro1'
    DO c:\ramiz\employee.prg
CASE m.btnname='pro3'
    DO c:\ramiz\search.spr
CASE m.btnname='pro4'
    DO c:\ramiz\menu.spr
    close all
    ENDCASE
RETURN

```



```

close data base
do find
procedure find
define window a1 at 12,12 size 4,40 system
activate window a1
your_ch=space(8)
@1,1 say "Enter Id_no:"
@1,25 get your_ch
read
release window a1
use c:\ramiz\employee.dbf
locate all for alltrim(id_no)=alltrim(your_ch)
if found() then
    WAIT WINDOW " Searching is OK " NOWAIT
    browse fields;
    id_no,name,surname,location,work_hours,sex:H=' ';
else
    WAIT WINDOW " not found " NOWAIT
    WAIT WINDOW " press enter to exit " NOWAIT
endif
return

```

```

close data base
do find
procedure find
define window a1 at 12,12 size 4,40 system
activate window a1
your_ch=space(8)
@1,1 say "Enter employee name:"
@1,25 get your_ch
read
release window a1
use c:\ramiz\employee.dbf
locate all for alltrim(name)=alltrim(your_ch)
if found() then
    WAIT WINDOW " Searching is OK " NOWAIT
    browse fields;
    name,surname,id_no,location,work_hours,sex:H=' ';
else
    WAIT WINDOW " not found " NOWAIT
    WAIT WINDOW " press enter to exit " NOWAIT
endif
return

```

CONCULOISON

This program was designed according to the basic needs of a circulation desk of bookstore. It has the advantage of fast search ability according to different parameters.

The program is user friendly even for those people who are not so familiar to the computers.

For bookstore a small network could be set up, so simultaneously more than one staff respond to the members for selling and similar process. This program could be easily adopted for the network use.

More and more options could also be added to the main menu here. For instance options for periodicals, scientific journals, microfilm and CD's.

One of the problems that I was faced was not to be familiar to the bar code programming. This kind of program should actually support the bar code machines so full automation could be reached in the bookstore.

Generally I could say that I could reach my goals after doing this project as now I have a general view about the programs that create databases.