

PROJECT REPORT
(Vol. II)

INFORMATION SYSTEM DEVELOPMENT
FOR ASEA LTD; BANGALORE

SUBMITTED IN PARTIAL FULFILMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
Master of Computer Applications
(M. C. A.)

Harsher Singh Grewal
DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING
THAPAR INSTITUTE OF ENGG. AND TECHNOLOGY, PATIALA
(DEEMED TO BE A UNIVERSITY)
1988

```

** OTMAIN.PRG
** Main menu for Overtime Calculation System (OCS)

clear
set talk off

** initialising variables

CHOICE = ' '
MTODAY = space(6)

** Checking if today's date has been set

do PRGDATE
if MTODAY <= '880325'
    @23,0
    wait "Today's date has not been set on booting the system.;
        Please reboot! Press <CR>.." to PRESS
    quit
endif

do while .T.

** Displaying the list of items in the Main Menu

clear
set color to /w
@ 4,38 say 'OCS'
@ 5,35 say 'MAIN MENU'
@ 5,54 say 'VERSION 0'
set color to w
@ 7,25 say '1. Add Personnel Information'
@ 8,25 say '2. Edit Personnel Data'
@ 9,25 say '3. Input OT Hours'
@ 10,25 say '4. Edit OT Hours Details'
@ 11,25 say '5. Enter Manual Adjustments'
@ 12,25 say '6. Check List Menu'
@ 13,25 say '7. Month end Calculations'
@ 14,25 say '8. Query Menu'
@ 15,25 say '9. Print Menu'
@ 16,25 say 'Q. Exit'
CHOICE = ' '
@ 19,25 say 'Enter your choice : ' get CHOICE picture '!'
read

** Case Statement

do case
    case CHOICE = '1'
        do PERSINFO

    case CHOICE = '2'
        do EDITPERS

    case CHOICE = '3'
        do OTGENIP

    case CHOICE = '4'
        do EDOTREG

    case CHOICE = '5'

```

```
do MANADJT

case CHOICE = '6'
do CKLSMNU

case CHOICE = '7'
do OTCALC

case CHOICE = '8'
do QMENU

case CHOICE = '9'
do PRIMENU

case CHOICE = 'Q'
clear all
return

endcase

enddo
```

```

** PRIMENU.PRG
** This is output report menu for getting the output for OT program

clear
set talk off

** Initialising the variables
FRDATE = space(4)
TODATE = space(4)
ANS     = ' '
DIV     = ' '
CATEGORY = ' '

do while .T.

  ** Displaying list of output reports available

  clear
  set color to w+
  @ 4,31 say 'OCS'
  @ 5,28 say 'Print Menu'
  set color to w
  @ 8,22 say '1. Overtime Summary Report-General staff'
  @ 9,22 say '2. Overtime Summary Report-Workers'
  @ 10,22 say '3. Overtime Summary Report-Temporary employees'
  @ 11,22 say '4. Worksheet'
  @ 12,22 say '0. Exit to previous Menu'
  CHOICE = 0
  @ 15,22 say 'Enter your Choice :' get CHOICE picture '9'
  read

  ** Opening the databases

  select A
    use OTREG index OTREG
  select B
    use PERS index PERS
  select C
    use TEMPWORK

  if CHOICE = 0
    clear
    close database
    return
  endif

  clear
  if CHOICE = 1 .or. CHOICE = 2 .or. CHOICE = 3 .or. CHOICE = 4
    do GETPD with FRDATE, TODATE, ANS
    if ANS = 'N'
      close database
      loop
    endif
  endif

  if CHOICE = 1
    CATEGORY = 'G'
  endif
  if CHOICE = 2
    CATEGORY = 'W'
  endif

```

```

if CHOICE = 3
    CATEGORY = 'WT'
endif
if CHOICE = 1 .or. CHOICE = 2 .or. CHOICE = 3
    @ 23,0 clear
    DIV = ' '
    @ 23,0 say 'Enter the Division as R or Y for summary report';
    get DIV picture '!'
    read
    do PRISUM with FRDATE, TODATE, CATEGORY, DIV
endif

if CHOICE = 4
    do PRIWKSH with FRDATE, TODATE
endif

enddo
close database
return

```

```

** CKLSMNU.PRG
** This is the Check list menu to print the inputs as entered

clear
set talk off

** Opening the databases

select A
  use OTREG index OTREG
select B
  use PERS index PERS
select C
  use MAN_ADJ

do while .T.
  clear
  set color to W+
  @ 4,33 say 'OCS'
  @ 5,27 say 'Check List Menu'
  set color to w
  @ 8,20 say '1. For personnel information'
  @ 10,20 say '2. For overtime hours'
  @ 12,20 say '3. For manual adjustments'
  @ 14,20 say '0. Exit to previous Menu'
  CHOICE = 0
  @ 17,23 say 'Enter your Choice :' get CHOICE picture '9'
  read

** Case statement

do case
  case CHOICE = 1
    @ 23,0 clear
    @ 23,0 say 'Set printer on ... and press any key'
    wait '' to KEY
    @ 24,0 say 'Printing is going on...'
    select A
    report form PERCKLST to print

  case CHOICE = 2
    do OTCKLST

  case CHOICE = 3
    @ 23,0 clear
    @ 23,0 say 'Set printer on ... and press any key'
    wait '' to KEY
    @ 24,0 say 'Printing is going on...'
    select C
    report form MANCKLST for PERDFR = '880217' noeject to print

  case CHOICE = 0
    clear
    close database
    return
endcase
enddo
close database
return

```

```

** QMENU.PRG
** This is the Query Menu for asking queries on terminal

clear
set talk off

do while .T.

  ** Displaying the queries available

  clear
  set color to W+
  @ 4,34 say 'OCS'
  @ 5,30 say 'Query Menu'
  set color to w
  @ 9,15 say '1. For a person give the overtime details'
  @ 10,22 say 'for the given period'
  @ 12,15 say '2. For a particular date give'
  @ 13,22 say 'the overtime details of employees'
  @ 15,15 say '0. Exit to previous Menu'
  CHOICE = 0
  @ 18,22 say 'Enter your Choice :' get CHOICE picture '9'
  read

  ** Opening the databases

  select A
    use OTREG index OTREG

  select B
    use PERS index PERS

  ** Case statement

  do case
    case CHOICE = 1
      do QOTEMP

    case CHOICE = 2
      do QEMPDT

    case CHOICE = 0
      clear
      exit
  endcase

enddo

```

```

** PERSINFO.PRG
** add screen
** Paints the screen for getting information about all the employees

clear
set talk off

** Painting headings
@ 1,20 say 'Personnel Information - Add screen'
@ 2,51 say 'Cate'
@ 2,57 say 'Card'
@ 3,2 say 'Emp.no'
@ 3,19 say 'Name'
@ 3,42 say 'Basic'
@ 3,51 say 'gory'
@ 3,58 say 'no'
@ 3,63 say 'Div.'
@ 3,68 say 'Dept'

** open PERSINFO database **
use PERS index PERS

ROW = 5

** Initialising the variables

XEMP_NO = ' '
XENAME = space(30)
XBASIC = 0.00
XCATEGORY = ' '
XCARD_NO = 0
XDIV = ' '
XDEPT = ' '
STATUS = ' '

do while .T.
@ ROW,3 get XEMP_NO picture '!999'
@ ROW,9 get XENAME picture '!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!'
@ ROW,41 get XBASIC picture '9999.99'
@ ROW,52 get XCATEGORY picture '!!'
@ ROW,57 get XCARD_NO picture '999'
@ ROW,64 get XDIV picture '!'
@ ROW,68 get XDEPT picture '!!!'
read
@ 20,0 clear

** if no name is given, check for going out **

if len(trim(XENAME)) = 0
ANS = 'N'
@ 23,0 say 'Do you want to continue ? (Y/N):' get ANS picture '!'
read
if ANS <> 'Y'
clear
close database
return
else
@23,0 clear
loop
endif
endif

```

```
** validations  program **
```

```
do VALIDPERS with XEMP_NO, XCATEGORY, XDIV, XDEPT
  if STATUS = 'L'
    loop
  endif
locate for EMP_NO = XEMP_NO
if .not. eof()
  @ 20,0 say 'Employee no. entered already exists, enter a new no.'
  loop
endif
```

```
** add record to database **
```

```
append blank
replace EMP_NO with XEMP_NO, ENAME with XENAME, BASIC with XBASIC
replace CATEGORY with XCATEGORY, CARD_NO with XCARD_NO,
replace DIV with XDIV, DEPT with XDEPT
```

```
** Initialising the variables
```

```
XEMP_NO = ' '
XENAME = space(30)
XBASIC = 0.00
XCATEGORY = ' '
XCARD_NO = 0
XDIV = ' '
XDEPT = ' '
PROW=ROW
ROW = ROW + 2
if ROW > 20
  ROW = 5
  @ 5,0 clear
  @ PROW,3 say EMP_NO
  @ PROW,9 say ENAME
  @ PROW,41 say BASIC
  @ PROW,52 say CATEGORY
  @ PROW,57 say CARD_NO
  @ PROW,64 say DIV
  @ PROW,68 say DEPT
endif
if ROW>10
  @ 19,0 clear
endif
enddo
close database
return
```

```

** EDITPERS.PRG
** Screen for editing the Personnel Information in PERS File

clear
set talk off

** Initialising the variables

XEMP_NO = ' '
XENAME = space(30)
XBASIC = 0.00
XCATEGORY = ' '
XCARD_NO = 0
XDIV = ' '
XDEPT = ' '
COMMAND = ' '
STATUS = ' '

** Open PERS database

use PERS index PERS

do while .T.

** Printing the headings

@ 3,20 say 'Personnel Information Editing Screen'
set color to /w
@ 5,5 say 'N-Next      P-Previous      E-Edit      S-Seek      X-Exit'
set color to w
@ 8,5 say 'Emp. no: '+EMP_NO
@ 8,26 say 'Name____: '+ENAME
@ 10,5 say 'Basic___: '+str(BASIC,7,2)
@ 10,26 say 'Category: '+CATEGORY
@ 12,5 say 'Card no: '+str(CARD_NO,3,0)
@ 12,26 say 'Division: '+DIV
@ 12,46 say 'Dept : '+DEPT

do while .T.

** Using the edit options

COMMAND = ' '
@ 1,0 say 'Give Command:' get COMMAND picture '!'
read
@20,0 clear

do case

case COMMAND = 'N'
if .not. eof()
skip
endif
if eof()
@ 23,0 say 'End of file, no more records after this !'
else
@ 8,14 say EMP_NO
@ 8,36 say ENAME
@ 10,14 say BASIC
@ 10,36 say CATEGORY

```

```

        @ 12,14 say CARD_NO
        @ 12,36 say DIV
        @ 12,53 say DEPT
    endif

case COMMAND = 'P'
    if .not. bof()
        skip -1
    endif
    if bof()
        @ 23,0 say 'You are already at the beginning of file !'
    else
        @ 8,14 say EMP_NO
        @ 8,36 say ENAME
        @ 10,14 say BASIC
        @ 10,36 say CATEGORY
        @ 12,14 say CARD_NO
        @ 12,36 say DIV
        @ 12,53 say DEPT
    endif

case COMMAND = 'S'
    do while .T.
        @ 1,20 say 'Give Emp. no:' get XEMP_NO picture '!999'
        read
        seek XEMP_NO
        if eof()
            @ 23,0 say '* Given Emp. no. not found *'
            ANS = 'N'
            @ 24,0 say 'Do you want to give new Emp. no. ? (Y/N):';
                get ANS picture '!'
            read
            if ANS <> 'Y'
                clear
                close database
                return
            else
                @ 23,0 clear
                loop
            endif
        endif
        @ 8,14 say EMP_NO
        @ 8,36 say ENAME
        @ 10,14 say BASIC
        @ 10,36 say CATEGORY
        @ 12,14 say CARD_NO
        @ 12,36 say DIV
        @ 12,53 say DEPT
        exit
    enddo

case COMMAND = 'E'
    XEMP_NO =EMP_NO
    XENAME =ENAME
    XBASIC =BASIC
    XCATEGORY=CATEGORY
    XCARD_NO =CARD_NO
    XDIV =DIV
    XDEPT =DEPT
    do while .T.
        @ 8,14 get XEMP_NO picture '!999'

```

```

@ 8,36 get XENAME picture '!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!'
@ 10,14 get XBASIC picture '9999.99'
@ 10,36 get XCATEGORY picture '!!'
@ 12,14 get XCARD_NO picture '999'
@ 12,36 get XDIV picture '!'
@ 12,53 get XDEPT picture '!!!'
read
@ 20,0 clear
do VALIDPERS with XEMP_NO, XCATEGORY, XDIV, XDEPT
  if STATUS = 'L'
    loop
  else
    exit
  endif
enddo
replace EMP_NO with XEMP_NO, ENAME with XENAME, BASIC with XBASIC
replace CATEGORY with XCATEGORY, CARD_NO with XCARD_NO
replace DIV with XDIV, DEPT with XDEPT
@23,0 say 'Record updated !'
case COMMAND = 'X'
  exit
endcase
enddo

ANS = 'N'
@ 24,0 say 'Do you want to edit information of another employee?(Y/N):';
get ANS picture '!'
read
if ANS <> 'Y'
  clear
  close database
  return
endif
@ 6,0 clear
enddo

```

```

** OTGENIP.PRQ
** Paints screen for getting OT details for an employee **

clear
set talk off

** initialising the variables

FRDATE=space(4)
TODATE=space(4)
XEMP_NO=' '
XCARD_NO=0
XHRFR=00.00
XHRTO=00.00
OK = 0
DAYS = ' '
STATUS = ' '

do while .T.

** painting the screen and get period, employee no, card no

set color to w+
@ 1,25 say "OT Hours add screen"
set color to w
@ 2,17 say "Period:"
@ 2,25 get FRDATE picture '9999'
set color to /w
@ 2,29 say '17'
set color to w
@ 2,33 say "to"
@ 2,37 get TODATE picture '9999'
set color to /w
@ 2,41 say '16'
set color to w
read
if FRDATE>=TODATE
    set color to w*
    @ 23,0 say 'Error in given period, correct the field'
    set color to w
    loop
endif
FRDATE=FRDATE+'17'
TODATE=TODATE+'16'
do CHKDATE with FRDATE, OK
if OK = 0
    set color to w*
    @ 23,0 say 'Error in given period, correct the field'
    set color to w
    loop
endif
do CHKDATE with TODATE, OK
if OK = 0
    set color to w*
    @ 23,0 say 'Error in given period, correct the field'
    set color to w
    loop
endif
exit
enddo
@ 23,0 clear

```

```

select A
use PERS index PERS
select B
use OTREG index OTREG

do while .T.

    XEMP_NO = '
    @ 2,49 say "Emp. no.:"
    @ 2,59 get XEMP_NO picture '!999'
    read

    select A
    seek XEMP_NO
    if .not. eof()
        set color to w+
        @ 3,17 say "Card no.:"
        @ 3,28 say CARD_NO
        @ 3,41 say "Name:"
        @ 3,48 say ENAME
        set color to w
    else
        @23,0 say '** Given Emp. no. not found **'
        ANS = '
        @24,0 say 'Do you want to give new Emp. no.?(Y/N):' get ANS picture '!'
        read
        if ANS <> 'Y'
            clear
            close database
            return
        else
            @23,0 clear
            loop
        endif
    endif

    select B
    go top
    locate for EMP_NO=XEMP_NO .and. PERDFR=FRDATE .and. PERDTO=TODATE

    if XEMP_NO = EMP_NO
        @ 23,0 say 'Given Emp. no. already present, enter new no.'
        loop
    endif
    @ 23,0 clear
    append blank
    replace EMP_NO with XEMP_NO, PERDFR with FRDATE, PERDTO with TODATE
    XDATE = FRDATE
    M = substr ( XDATE, 3, 2 )
    Y = substr ( XDATE, 1, 2 )
    do GETDAYS with Y, M , DAYS

    ROW=6
    COL=0
    @ 4,COL+9 say "OT Hours"
    @ 5,COL+1 say "Date From To"

    do while .T.

        @ ROW,COL+0 say XDATE

```

```

@ ROW,COL+7  get  XHRFR picture '99.99'
@ ROW,COL+13 get  XHRTO picture '99.99'
read
do OTHRCHK with XHRFR, XHRTO, STATUS
  if STATUS = 'E'
    exit
  endif
  if STATUS = 'L'
    loop
  endif

D = substr (XDATE, 5, 2)
FR = 'HRFR' + D
TO = 'HRTO' + D
replace &FR with XHRFR, &TO with XHRTO

if XDATE=TODATE
  exit
endif

ROW = ROW + 2
if ROW >21
  ROW=6
  COL = COL + 20
  @ 4,COL+9 say "OT Hours"
  @ 5,COL+1 say "Date From To"
endif
do GETNXTDT with XDATE, DAYS
enddo

@ 23,0 clear
ANS=' '
@24,0 say 'Do you want to enter OT details for another person ?(Y/N):';
get ANS picture '!'
read
if ANS<>'Y'
  clear
  close database
  return
endif
@3,0 clear
enddo

```

```

** EDOTREG.PRG
** Screen for editing the Overtime Hours entered in the OTREG file

clear
set talk off

** Initialising the variables

FRDATE =space(4)
TODATE =space(4)
XHRFR =0.00
XHRTO =0.00
DAYS = ' '
STATUS = ' '

select A
use OTREG index OTREG

set color to /w
@ 1,25 say 'OT Hours Editing Screen'
@ 2,0 say 'N - Next      E - Edit      X - Exit'
set color to w

do while .T.

XEMP_NO = ' '
COMMAND = ' '
** Creates editing screen

@ 3,13 say 'Emp. no:'
@ 3,23 get XEMP_NO picture '!999'
@ 3,31 say 'Period:'
@ 3,39 get FRDATE picture '9999'
set color to /w
@ 3,43 say '17'
set color to w
@ 3,47 say 'to'
@ 3,51 get TODATE picture '9999'
set color to /w
@ 3,55 say '16'
set color to w
read

FRDATE = substr(FRDATE,1,4) + '17'
TODATE = substr(TODATE,1,4) + '16'
select A
go top
locate for EMP_NO = XEMP_NO .and. PERDFR = FRDATE .and. PERDTO = TODATE

if eof()
@ 23,0 say '* Given Emp. no. for the said period not found *'
ANS = 'N'
@ 24,0 say 'Do you want to give new Emp. no ?(Y/N):' get ANS picture '!'
read
if ANS <> 'Y'
clear
close database
return
else
@ 23,0 clear
loop

```

```

endif
endif

XDATE = FRDATE
M = substr(XDATE, 3, 2)
Y = substr(XDATE, 1, 2)
do GETDAYS with Y, M, DAYS

ROW = 6
COL = 0
@ 4, COL+9 say 'OT Hours'
@ 5, COL+1 say 'Date From To'
D = substr(XDATE, 5, 2)
FR = 'HRFR' + D
TO = 'HRTO' + D
@ ROW, COL say XDATE
@ ROW, COL+7 say &FR
@ ROW, COL+13 say &TO

do while .T.
  COMMAND=' '
  do while COMMAND<>'N' .and. COMMAND<>'E' .and. COMMAND<>'X'
    @ 0,0 say 'Give Command:' get COMMAND picture '!'
    read
  enddo

  @ 22,0 clear
  do case
    case COMMAND = 'N'
      do GETNXTDT with XDATE, DAYS
      D = substr(XDATE, 5, 2)
      FR = 'HRFR' + D
      TO = 'HRTO' + D
      ROW = ROW+2
      if ROW > 22
        ROW = 6
        COL = COL+20
        @ 4, COL+9 say 'OT Hours'
        @ 5, COL+1 say 'Date From To'
      else
        if ROW > 21
          ROW = 7
          COL = COL+20
          @ 4, COL+9 say 'OT Hours'
          @ 5, COL+1 say 'Date From To'
        endif
      endif
    endif
    if XDATE > TODATE
      exit
    endif
    @ ROW, COL+0 say XDATE
    @ ROW, COL+7 say &FR
    @ ROW, COL+13 say &TO

  case COMMAND = 'E'
    do while .T.
      XHRFR = &FR
      XHRTO = &TO
      @ ROW, COL+0 say XDATE
      @ ROW, COL+7 get XHRFR picture '99.99'
      @ ROW, COL+13 get XHRTO picture '99.99'
    enddo
  endcase
enddo

```

```

        read

        do OTHRCHK with XHRFR, XHRTD, STATUS
        if STATUS = 'L'
            loop
        else
            exit
        endif
        enddo
        D = substr(XDATE, 5, 2)
        FR = 'HRFR' + D
        TO = 'HRTD' + D
        replace &FR with XHRFR, &TO with XHRTD
        @ 22,0 say 'Record updated !'

        case COMMAND = 'X'
            exit
        endcase
    enddo

    ANS = 'N'
    @ 24,0 say 'Do you want to edit information of another employee ?(Y/N):';
    get ANS picture '!'
    read
    if ANS <> 'Y'
        clear
        close database
        return
    endif
    @ 6,0 clear
enddo

```

```

** OTCALC.PRG
** Program to do month end overtime calculations
clear
set talk off

** Initialising the variables

FRDATE= space(4)
TODATE= space(4)
TOTHR= 0.00
TAMT = 0.00
DPAY = 0.00
DASL = 0.00
DASH = 0.00
DAW = 0.00
ANS = ' '
XDIV = ' '

** Opening the databases

select A
use OTREG index OTREG

select B
use PERS index PERS

select C
use TEMPWORK
zap

select D
use MAN_ADJ

** Getting the period for which calculations are to be done

do GETPD with FRDATE, TODATE, ANS
if ANS = 'N'
clear
close database
return
endif

select A
go top
seek FRDATE+TODATE
if eof()
@ 23,0 clear
@ 23,0 say 'No OT details found for the given period, press any key'
wait '' to ANYKEY
return
endif

do WKSHEET with FRDATE, TODATE

@ 23,0 clear
@ 23,10 say 'Enter the D.A. for all for the above period'
@ 24,0 say 'DA Gen-Basic <101:' get DASL picture '999.99'
@ 24,31 say 'DA Gen-Others:' get DASH picture '999.99'
@ 24,58 say 'DA Workers:' get DAW picture '999.99'
read
@ 23,0 clear

```

@ 24,0 say 'Calculating OT amount...'

```
select A
go top
seek FRDATE+TODATE
```

```
do while .T.
```

```
    DPAY = 0.00
```

```
    TAMT = 0.00
```

```
    SHRS = 0.00
```

```
    ADJT = 0.00
```

```
    select B
```

```
    seek A->EMP_NO
```

```
    XCATGY = substr(CATEGORY, 1, 1)
```

```
    if XCATGY = 'G' .and. BASIC < 101
```

```
        DPAY = (BASIC+DASL)/208.00
```

```
    else
```

```
        if XCATGY = 'G'
```

```
            DPAY = (BASIC+DASH)/208.00
```

```
        else
```

```
            if (XCATGY = 'W' .and. substr(CATEGORY, 2, 1) = 'T');
```

```
                .or. (XCATGY = 'N' .and. substr(CATEGORY, 2, 1) = 'G')
```

```
                DPAY = BASIC/208.00
```

```
            else
```

```
                if XCATGY = 'W'
```

```
                    DPAY = (BASIC+DAW)/208.00
```

```
                endif
```

```
            endif
```

```
        endif
```

```
    endif
```

```
    select D
```

```
    go top
```

```
    locate for PERDFR = FRDATE .and. PERDTO = TODATE .and. EMP_NO= A->EMP_NO
```

```
    store HRS_SGL to SHRS
```

```
    store AMT_ADJ to ADJT
```

```
    select A
```

```
    TOTHR = (TOT_OT_HRS*2) - SHRS
```

```
    TAMT = DPAY*TOTHR + ADJT
```

```
    replace OT_AMT with TAMT
```

```
    skip
```

```
    if (PERDFR<>FRDATE .or. PERDTO<>TODATE) .or. eof()
```

```
        exit
```

```
    endif
```

```
enddo
```

```
@ 21,0 clear
```

```
@ 23,0 say 'Month end calculations complete, press any key'
```

```
wait '' to KEY
```

```
close database
```

```
return
```

```
** WKSHEET.PRG
** Outputs a Worksheet showing daily overtime hours and monthly totals
```

```
clear
set talk off
parameters FRDATE, TODATE
```

```
** Initialising the variables
```

```
DAYS = ' '
XHRFR = SPACE(6)
XHRTO = SPACE(6)
FR=0.00
TO=0.00
DIFF = 0.00
```

```
@23,0 clear
@ 23,0 say 'Wait, .... doing the calculations'
NOE=1
```

```
** Total OT hours calculation
```

```
do while .T.
```

```
  XDATE = FRDATE
  M = substr(XDATE, 3, 2)
  Y = substr(XDATE, 1, 2)
  do GETDAYS with Y, M, DAYS

  @23,40 say 'No of emp :'+str(NOE,3)
```

```
  select B
  seek A->EMP_NO
```

```
  select C
  append blank
  replace EMP_NO with A->EMP_NO
  replace ENAME with B->ENAME
  replace PERDFR with FRDATE, PERDTO with TODATE
  TOTHR = 0.00
```

```
  do while .T.
    DIFF = 0.00
    select A
    DT = substr(XDATE, 5, 2)
    XHRFR='HRFR'+DT
    XHRTO='HRTO'+DT
    FR=&XHRFR
    TO=&XHRTO
    select C
    HD = 'D' + DT
```

```
    if (FR>0.00 .and. TO>0.00)
      do HRSDIFF with FR, TO, DIFF
      if (FR < 12.00 .and. TO > 12.30)
        DIFF = DIFF - 0.50
      endif
      if TO > 21.45
        DIFF = DIFF - 0.50
      endif
```

```

        replace &HD with DIFF
        TOTHR5 = TOTHR5 + DIFF
    else
        replace &HD with 0.00
    endif

    if (DT > '09' .and. DT < '28')
        NEWD = str(val(DT)+1, 2)
        XDATE = substr(XDATE, 1, 4)+NEWD
    else
        do GETNXTDT with XDATE, DAYS
    endif
    if XDATE > TODATE
        exit
    endif
enddo
replace TOT_OT_HRS with TOTHR5, CALC_DATE with MTDAY

select A
replace TOT_OT_HRS with TOTHR5
skip
if (PERDFR<>FRDATE .or. PERDTO<>TODATE) .or. eof()
    exit
endif
NOE=NOE+1
enddo

@24,0
@24,0 say 'Worksheet calculation complete...Press any key'
wait '' to KEYPRS
return

```

```

** QEMPDT.PRG
** Query program for displaying the overtime details for the employees
** for a given date

clear
set talk off

** Initialising the variables
XDATE= space(6)
PFR   = space(4)
PTO   = space(4)
OK     = 0
CNT    = ' '
DAYS   = ' '
DT     = ' '
YM     = ' '

** Painting the screen

do while .T.
XDATE = space(6)
@ 2,2 say 'Give date for which data required:' get XDATE picture '999999'
read
do CHKDATE with XDATE, OK
    if OK = 0
        @ 23,0 say 'Error in given date '
        ANS = 'N'
        @ 24,0 say 'Do you want to give another date ?(Y/N):';
            get ANS picture '!'
        read
        if ANS <> 'Y'
            clear
            return
        else
            @ 23,0 clear
            loop
        endif
    endif
endif

select A
go top
DT = substr(XDATE, 5, 2)
YM = substr(XDATE, 1, 4)

if val(DT) >= 17
    locate for substr(PERDFR,1, 4) = YM
    if eof()
        @ 24,0 say 'Given date not found, give new one'
        loop
    endif
else
    locate for substr(PERDTD,1, 4) = YM
    if eof()
        @ 24,0 say 'Given date not found, give new one'
        loop
    endif
endif

@ 1,0 clear
set color to w+
@ 1,25 say 'Overtime details'

```

```

set color to w
@ 2,1 say 'Date: '+XDATE
@ 2,41 say 'OT Hours'
@ 3,1 say 'Emp. no'+space(8)+'Name'+space(20)+'From To'
@ 3,58 say 'Dept.'

FR = 'HRFR' + DT
TO = 'HRTD' + DT
ROW = 4
do while YM = substr(PERDFR,1, 4) .or. YM = substr(PERDTD, 1, 4)
  if (&FR >0.00 .and. &TO >0.00)
    @ ROW,2 say EMP_NO
    select B
    seek A->EMP_NO
    @ ROW,10 say ENAME
    @ ROW,59 say DEPT

    select A
    @ ROW,39 say &FR
    @ ROW,46 say &TO
    ROW = ROW + 2
    if ROW > 21
      @ 24,0 say 'Press space bar to continue'
      wait '' to PAGE
      @ 4,0 clear
      ROW = 4
    endif
  endif
  skip
enddo

@ 23,0 say 'Press any key to exit'
wait '' to CNT
exit
enddo
clear
close database
return

```

```

** OTCKLST.PRG
** Check list to print the overtime hours entered in OTREG file

clear
set talk off

** Initialising the variables

FRDATE = space(4)
TODATE = space(4)
DAYS   = ' '
ANS    = ' '

do while .T.
do GETPD with FRDATE, TODATE, ANS
if ANS = 'N'
clear
return
endif

select A
go top
locate for PERDFR = FRDATE .and. PERDTO = TODATE
if eof()
@ 23,0 say 'No OT details found for the given period, press a key'
wait '' to KEY
clear
return
endif
exit
enddo

@ 23,0 clear
@ 23,0 say 'Set printer on ... and press any key'
wait '' to PRT
set print on
@ 23,0 clear
@ 24,0 say 'Printing the checklist...'
set console off
? space(22)+'OT Hours - Check list'
? space(22)+'Period: '+FRDATE+' to '+TODATE

do while .T.
select B
seek A->EMP_NO
EMPCNT = 0
?
? 'Emp no: '+EMP_NO+space(4)+'Name: '+ENAME
? space(8)+'OT Hours'+space(12)+'OT Hours'+space(12)+'OT Hours'+
space(12)+'OT Hours'
? ' Date From To '+' Date From To '+' Date From To '
?? ' Date From To '
XDATE = FRDATE
M = substr(XDATE, 3, 2)
Y = substr(XDATE, 1, 2)
do GETDAYS with Y, M, DAYS

select A
do while .T.
?
?
```

CNT = 0

do while CNT < 4

 D = substr(XDATE, 5, 2)

 FR = 'HRFR' + D

 TO = 'HRTD' + D

 ?? XDATE+' '+str(&FR,5,2)+' '+str(&TO,5,2)+' '

 CNT = CNT + 1

 do GETNXTDT with XDATE, DAYS

 if XDATE > TODATE

 exit

 endif

enddo

if XDATE > TODATE

 exit

endif

enddo

EMPCNT = EMPCNT + 1

if EMPCNT = 3

 ?

 eject

 exit

endif

skip

?

?

if PERDFR <> FRDATE .or. PERDTR <> TODATE .or. eof()

 exit

endif

enddo

set print off

set console on

@ 23,0 clear

@ 24,0 say ' Printing is over, press a key '

wait '' to KEY

return

```

** GETNXTDT.PRG
** Program for changing the date according to the calender

parameters XDATE, DAYS

D = substr(XDATE, 5, 2)
M = substr(XDATE, 3, 2)
Y = substr(XDATE, 1, 2)
VAL_D = val(D)+1
V_DAYS = val(DAYS)
if VAL_D > V_DAYS
    VAL_D = 1
    VAL_M = val(M) + 1
    if VAL_M = 13
        VAL_M = 1
        VAL_Y = val(Y) + 1
        Y = str(VAL_Y, 2)
    endif

    if VAL_M < 10
        M = '0'+str(VAL_M,1)
    else
        M = str( VAL_M, 2 )
    endif
endif

if VAL_D < 10
    D = '0'+str(VAL_D,1)
else
    D = str( VAL_D, 2 )
endif

XDATE = Y+M+D
return

```

```

** QOTEMP.PRG
** Query program for displaying the OT details for a person for a given period

clear
set talk off

** Initialising the variables
FRDATE = space(4)
TODATE = space(4)
XEMP_NO= ' '
DAYS    = ' '
ANS     = ' '

** getting the period for which OT details wanted

do while .T.

    @ 1,10 say 'Enter the period for which you want the information'
    @ 2,10 say 'Period:' get FRDATE picture '9999'
    set color to /w
    @ 2,22 say '17'
    set color to w
    @ 2,25 say 'to' get TODATE picture '9999'
    set color to /w
    @ 2,32 say '16'
    set color to w
    read

    FRDATE = substr(FRDATE, 1, 4) + '17'
    TODATE = substr(TODATE, 1, 4) + '16'

    @ 2,48 say 'Emp. no:' get XEMP_NO picture '!999'
    read

    select A
    go top
    locate for PERDFR = FRDATE .and. PERDTO = TODATE .and. EMP_NO = XEMP_NO
    if eof()
        @ 23,0 say '* Given Emp. no. for the said period not found *'
        ANS = 'N'
        @ 24,0 say 'Do you want to give new Emp. no. ?(Y/N):' get ANS picture '!'
        read
        if ANS <> 'Y'
            clear
            return
        else
            @ 23,0 clear
            loop
        endif
    endif

** Displaying the OT details

clear
set color to w+
@ 1,23 say 'OT details for an employee'
set color to w
@ 2,1 say 'Period: '+FRDATE+ ' to '+TODATE+space(23)+'Emp. no: '+XEMP_NO

select B
seek A->EMP_NO

```

```

@ 3,1 say 'Name : '+ENAME+space(9)+'Div : '+DIV+space(5)+'Dept: '+DEPT
ROW = 6
COL = 0
@ 4,COL+9 say 'OT Hours'
@ 5,COL+1 say 'Date From To'

select A
XDATE = FRDATE
M = substr(XDATE, 3, 2)
Y = substr(XDATE, 1, 2)
do GETDAYS with Y, M, DAYS

do while .T.
  D = substr(XDATE,5,2)
  FR = 'HRFR'+D
  TO = 'HRTD'+D
  @ ROW,COL+0 say XDATE
  @ ROW,COL+7 say &FR
  @ ROW,COL+13 say &TO
  ROW = ROW + 2
  if ROW > 21
    ROW = 6
    COL = COL + 20
    @ 4,COL+9 say 'OT Hours'
    @ 5,COL+1 say 'Date From To'
  endif
  if XDATE = TODATE
    exit
  endif
  do GETNXTDT with XDATE, DAYS
enddo

@ 22,0 say 'Total OT Hours:'
@ 22,16 say TOT_OT_HRS
@ 22,34 say 'Total OT Amt: Rs.'
@ 22,51 say OT_AMT

ANS = 'N'
@ 24,0 say 'Do you want to see OT Details for another employee';
get ANS picture '!'
read
if ANS <> 'Y'
  clear
  return
else
  @ 24,0 clear
  loop
endif
enddo

```

```

** MANADJT.PRG
** add screen
** Paints the screen for getting OT adjustment information about employees

clear
set talk off

** Painting headings

    set color to w+
    @ 1,25 say 'Manual Adjustment Screen'
    set color to w
    @ 2,8 say 'Period'+ ' '+'Period'+ ' '+'Single'
    @ 2,31 say 'Adjust'+space(33)+'Entered'
    @ 3,1 say 'Emp.no'+ ' '+'From'+ ' '+'To'+ ' '+'Hours'
    @ 3,31 say 'Amount'+space(8)+'Reason'+space(19)+'By-name'

** open Manual Adjustment database **

    use MAN_ADJ

    ROW = 5

** Initialising the variables

    XEMP_NO = ' '
    XPERDFR = space(6)
    XPERDTO = space(6)
    XHRS_SGL= 0.00
    XAMT_ADJ= 0.00
    XREASON = space(30)
    XENTD_BY= space(10)

do while .T.

    @ ROW,2 get XEMP_NO picture '!999'
    @ ROW,8 get XPERDFR picture '999999'
    @ ROW,16 get XPERDTO picture '999999'
    @ ROW,24 get XHRS_SGL picture '99.99'
    @ ROW,31 get XAMT_ADJ picture '9999.99'
    @ ROW,39 get XREASON picture '!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!'
    @ ROW,70 get XENTD_BY picture '!!!!!!!!!!!!'
    read
    @ 20,0 clear

** if no period is given, check for going out **

    if XPERDFR = ' ' .and. XPERDTO = ' '
        ANS = 'N'
        @ 23,0 say 'Do you want to continue ? (Y/N):' get ANS picture '!'
        read
        if ANS <> 'Y'
            clear
            close database
            return
        else
            @23,0 clear
            loop
        endif
    endif

```

```
** validations  program **
```

```
locate for EMP_NO = XEMP_NO .and. PERDFR = XPERDFR .and. PERDTO = XPERDTO
if .not. eof()
    @ 20,0 say 'Employee no. entered already exists, enter a new no.'
    loop
endif
```

```
** add record to database **
```

```
append blank
replace EMP_NO with XEMP_NO, PERDFR with XPERDFR, PERDTO with XPERDTO
replace HRS_SGL with XHRS_SGL, AMT_ADJ with XAMT_ADJ, REASON;
with XREASON, ENTD_BY with XENTD_BY
```

```
** Initialising the variables
```

```
XEMP_NO = ' '
XPERDFR = space(6)
XPERDTO = space(6)
XHRS_SGL= 0.00
XAMT_ADJ= 0.00
XREASON = space(30)
XENTD_BY= space(10)
```

```
PROW=ROW
ROW = ROW + 2
if ROW > 20
    ROW = 5
    @ 5,0 clear
    @ ROW,3 say XEMP_NO
    @ ROW,9 say XPERDFR
    @ ROW,17say XPERDTO
    @ ROW,25say XHRS_SGL
    @ ROW,31say XAMT_ADJ
    @ ROW,38 say XREASON
    @ ROW,69 say XENTD_BY
endif
if ROW>10
    @ 19,0 clear
endif
enddo
```

```
close database
return
```

```

** PRISUM.PRG
** To print output reports for the overtime hours and amount

clear
set talk off
parameters FRDATE, TODATE, CATEG, XDIV

** initialising the variables

SNO = 0
ANS = ' '
HEADING = 0
KEY = ' '

select A
go top
locate for PERDFR = FRDATE .and. PERDTO = TODATE
if eof()
    @ 23,0 say 'No OT details found for the given period'
    return
endif

** printing the summary

@ 23,0 clear
@ 23,0 say 'Set printer on ... and press any key' get ANS
read
set print on
eject
SNO = 1
NOOTDET = 1

** printing the headings

do while .T.
    select B
    seek A->EMP_NO
    XCATGY = CATEGORY
    if (XCATGY = CATEG .and. XCATGY <> 'WT') .or.;
        (CATEG = 'G' .and. XCATGY = 'NG');
        .or. (CATEG = 'WT' .and. XCATGY = 'WT')

    select A
    if substr(EMP_NO, 1, 1) = XDIV
        if HEADING = 0
            @ 24,0 say 'Printing .....'
            ? space(48)+'Date: '+MTODAY
            ? chr(27)+chr(69)+space(22)+'Overtime Summary Report'
            if CATEG = 'G' .or. XCATGY = 'NG'
                ? space(26)+'(General Staff)'
            endif
            if CATEG = 'W' .and. XCATGY<>'WT'
                ? space(28)+'(Workers)'
            endif
            if CATEG = 'WT'
                ? space(24)+'(Temporary Employees)'
            endif
            ? chr(27)+chr(70)

            YRS = substr(FRDATE,1,2)
            NYR = substr(TODATE,1,2)
        endif
    endif
endwhile

```

```

YR = '19' + YRS
? space(7)+'Year' : '+YR
if NYR > YRS
?? '-' + NYR
endif

??space(29)+'Division: '+XDIV
? space(7)+'Period: '+FRDATE+' to '+TODATE
?
? space(7)+'SNo'+space(4)+'Emp. no'+space(5)+'Total OT Hours'
??space(7)+'Total OT Amount'
? space(7)+'-----'
?? '-----'
?
HEADING = 1
endif
NOOTDET = 0

** printing the data

? str(SNO)+space(6)+EMP_NO+space(10)+str(TOT_OT_HRS,6,2)
??space(14)+str(OT_AMT,7,2)
?
SNO = SNO + 1
endif
endif
select A
skip
if (PERDFR<>FRDATE .or. PERDTO<>TODATE) .or. eof()
exit
endif
enddo
if NOOTDET = 1
@ 23,0 say ' No OT details found for this Category '
@ 24,0 say ' Press any key... to continue ' get KEY
read
return
endif

clear
@ 24,0 say ' Printing is over, press any key '
set print off
wait '' to ANYKEY
return

```

```

** PRIWKSH.PRG
** Outputs a Worksheet showing daily overtime hours and monthly totals

clear
set talk off
parameters FRDATE, TODATE

** Initialising the variables
DAYS = ' '
ANS = ' '

select C
if .not. (PERDFR = FRDATE .and. PERDTO = TODATE)
    @ 23,0 clear
    @ 23,0 say 'No data available for the given period, press a key'
    wait '' to ANYKEY
    return
else
** Set printer on
    @ 23,0 say 'Set printer on and press any key...' get ANS picture '!'
    read
    set print on
    @ 23,0
    @ 23,0 say 'Printing is going on...'
    set console off

    ? chr(27)+chr(15)
    ?? 'Report Date: '+MTODAY
    ?? space(62)+'WORKSHEET'+ for '+'Period: '+FRDATE+' to '+TODATE
    ?
    ?
    ? space(84)+'OT in Hours for dates in first row'
    ?
    ? 'Emp. no'+ ' '

    XDATE = FRDATE
    M = substr(XDATE, 3, 2)
    Y = substr(XDATE, 1, 2)
    do GETDAYS with Y, M, DAYS
    if DAYS = '28'
        ?? ' 17 '+' 18 '+' 19 '+' 20 '+' 21 '+' 22 '+' 23 '
        ?? ' 24 '+' 25 '+' 26 '+' 27 '+' 28 '
    endif
    if DAYS = '29'
        ?? ' 17 '+' 18 '+' 19 '+' 20 '+' 21 '+' 22 '+' 23 '
        ?? ' 24 '+' 25 '+' 26 '+' 27 '+' 28 '+' 29 '
    endif
    if DAYS = '30'
        ?? ' 17 '+' 18 '+' 19 '+' 20 '+' 21 '+' 22 '+' 23 '
        ?? ' 24 '+' 25 '+' 26 '+' 27 '+' 28 '+' 29 '+' 30 '
    endif
    if DAYS = '31'
        ?? ' 17 '+' 18 '+' 19 '+' 20 '+' 21 '+' 22 '+' 23 '
        ?? ' 24 '+' 25 '+' 26 '+' 27 '+' 28 '+' 29 '+' 30 ;
        '+' 31 '
    endif

    ?? ' 1 '+' 2 '+' 3 '+' 4 '+' 5 '+' 6 '+' 7 '
    ?? ' 8 '+' 9 '+' 10 '+' 11 '+' 12 '+' 13 '+' 14 '
    ?? ' 15 '+' 16 '
    ?? 'Total OT Hrs'+space(3)+'Calc Date'

```

```

do while .not. eof()
  ? space(1)+EMP_NO+space(4)
  XDATE = FRDATE
  do while XDATE <= TODATE
    D = substr(XDATE, 5, 2)
    HD = 'D' + D
    if &HD = 0.00
      ?? space(6)
    else
      ?? str(&HD,5,2)+' '
    endif

    do GETNXTDT with XDATE, DAYS
  enddo
  ??space(4)+str(TDT_OT_HRS,6,2)+space(6)+CALC_DATE
  skip
enddo
endif

?chr(18)
set print off
set console on
@ 24,0 clear
@ 24,0 say 'Printing is over'
return

```

```
** OTHRCHK.PRG
** Program for validating the OT Hours entered
```

```
parameters XHRFR, XHRTD, STATUS
STATUS = ' '
```

```
FRS = ' '
FRM = ' '
TOS = ' '
TOM = ' '
```

```
NOERR = .T.
FRS = str(XHRFR, 5, 2)
FRM = substr(FRS, 4, 2)
TOS = str(XHRTD, 5, 2)
TOM = substr(TOS, 4, 2)
```

```
NOERR= (FRM='15' .or. FRM='30' .or. FRM='45' .or. FRM='00');
       .and. (TOM='15' .or. TOM='30' .or. TOM='45' .or. TOM='00')
```

```
if .not. (XHRFR=0.00 .and. XHRTD=0.00)
  if XHRFR > 24.0 .or. XHRTD > 24.0 .or. XHRFR>XHRTD .or. .not. NOERR
    ANS = ' '
    @ 23,0 say 'Error in time. Do you want to continue ? (Y/N):';
    get ANS picture '!';
    read
    if ANS <> 'Y'
      STATUS = 'E'
    else
      @23,0 clear
      STATUS = 'L'
    endif
  endif
endif
return
```

```

** VALIDPERS.PRG
** Program to validate various parameters in persinfo file

parameters XEMP_NO, XCATEGORY, XDIV, XDEPT

STATUS = ' '

** EMP_NO starts with 'Y', 'R' and greater than 300

if (substr(XEMP_NO, 1, 1) <> 'R' .and. substr(XEMP_NO, 1, 1) <> 'Y');
  .or. substr(XEMP_NO, 2, 3) < '300' .or. XEMP_NO = '
    @ 21,0 say 'Error in Emp. no. ,give >= R300 or Y300 '
    STATUS = 'L'
  endif

** CATEGORY's first character is 'G', 'W', 'T', 'M' or 'N'

XCATGY = substr(XCATEGORY, 1, 1)
if (XCATGY <> 'G' .and. XCATGY <> 'M' .and. XCATGY <> 'T' .and. XCATGY <>
  'W' .and. XCATGY <> 'N')
  @ 22,0 say "Error in Category type, give first figure only as;
    'G','W','T','N' or 'M'"
  STATUS = 'L'
endif

** DIV is 'R', 'Y' and DIV= 1st character in EMP_NO

if (XDIV <> 'R' .and. XDIV <> 'Y') .or. XDIV = ' ';
  .or. XDIV <> substr(XEMP_NO, 1, 1)
  @ 23,0 say "Error in Division type, give only 'R' or 'Y';
    and same as in Emp. no."
  STATUS = 'L'
endif

** DEPT starts with 'R' or 'Y' only
if substr(XDEPT,1, 1) <> substr(XEMP_NO, 1, 1) .or. XDEPT = ' '
  @ 24,0 say "Error in Dept. type, give first letter same as in Emp. no."
  STATUS = 'L'
endif
return

```

```

** GETPD.PRG
** Gets the period for which computations are to be done

clear
set talk off
parameters FRDATE, TODATE, ANS

** Initialising the variables

FRDATE = space(4)
TODATE = space(4)

do while .T.

    FRDATE = space(4)
    TODATE = space(4)
    @ 21,10 say 'Enter the period for which you want the information'
    @ 22,10 say 'Period:' get FRDATE picture '9999'
    set color to /w
    @ 22,22 say '17'
    set color to w
    @ 22,25 say 'to' get TODATE picture '9999'
    set color to /w
    @ 22,32 say '16'
    set color to w
    read

    if FRDATE = ' ' .or. TODATE = ' '
        ANS = 'N'
        @ 23,0 say 'Error in period. Do you want to continue?(Y/N):';
        get ANS picture '!'
        read
        if ANS <> 'Y'
            ANS = 'N'
            exit
        else
            @ 23,0 clear
            loop
        endif
    endif

    FRDATE = FRDATE + '17'
    TODATE = TODATE + '16'
    exit
enddo
return

```

```

** GETDAYS.PRG
** Program to get no. of days in the given month **

parameters YEAR, MON, DAYS

** initialising variable **

LY = 0.00
Y = 0.0
LYINT = 0

** checking for months with 31 days **

if MON = '01' .or. MON = '03' .or. MON = '05' .or. MON = '07';
    .or. MON = '08' .or. MON = '10' .or. MON = '12'
    DAYS = '31'
else

** checking for months with 30 days **

    if MON = '04' .or. MON = '06' .or. MON = '09' .or. MON = '11'
        DAYS = '30'
    endif
endif

** checking for the month of february - leap year or not **

if MON = '02'
    store val(YEAR) to Y
    LY = Y / 4.0
    store int(LY) to LYINT
    if LYINT <> LY
        DAYS = '28'
    else
        DAYS = '29'
    endif
endif
return

```

```

** CHKDATE.PRG
** Program for checking the correctness of given date

parameter DATE, OK
OK = 1

** checking the year **

if substr( DATE, 1, 2 ) < '88'
    OK = 0
endif

** checking the month **

if substr( DATE, 3, 2 ) < '01' .OR. substr( DATE, 3, 2 ) > '12'
    OK = 0
endif

** checking the date **

DAYS = ' '
do GETDAYS with substr(DATE, 1, 2), substr(DATE, 3, 2), DAYS
if substr(DATE, 5, 2) < '01' .or. substr(DATE, 5, 2) > DAYS
    OK = 0
endif

```

```

** HRSDIFF.PRG
** Program to calculate the OT Hours for an employee

    parameters XHRFR, XHRT0, DIFF

** Initialising the variables

    FMIN= 0.00
    TMIN= 0.00
    MHRF= 0.00
    MHRT= 0.00
    XHRF= 0.00
    XHRT= 0.00

** Calculating the overtime hours

    FRS = str(XHRFR, 5, 2)
    TOS = str(XHRT0, 5, 2)
    FMIN =val( substr(FRS, 4, 2 ))
    TMIN =val( substr(TOS, 4, 2 ))
    MHRF = FMIN / 60.0
    MHRT = TMIN / 60.0
    XHRF = MHRF + val(substr(FRS, 1, 2))
    XHRT = MHRT + val(substr(TOS, 1, 2))
    DIFF = XHRT - XHRF

return

```

```
** PRGDATE.PRG  
** Extracts ASEA's standard date format from system date
```

```
Y=' '  
M=' '  
D=' '  
TEMP=' '  
MTODAY=' '  
TEMP=dtoc(Date())  
store substr(TEMP,7) to Y  
store substr(TEMP,1,2) to M  
store substr(TEMP,4,2) to D  
MTODAY=Y+M+D
```

ORDER HANDLING SYSTEM

```

** OHMENU.PRG
** Main menu for Order Handling System

clear
set talk off

** initialising variables

CHOICE = ' '
MTODAY = space(6)

** checking if today's date has been set

do PRGDATE
if MTODAY <= '880301'
    @ 23,0
    wait "Today's date has not been set on booting the system.;
        Please reboot ! Press <CR>..." to PRESS
    quit
endif

do while .T.

** Displaying the list of items in the Main Menu

clear
set color to /w
@ 4,33 say 'OHS'
@ 5,30 say 'MAIN MENU'
set color to w
@ 7,25 say '1. Adding Menu'
@ 9,25 say '2. Edit Order Information'
@ 11,25 say '3. Edit Activity Information'
@ 13,25 say '4. Print Menu'
@ 15,25 say '5. Checklist Menu'
@ 17,25 say 'Q. Exit'
CHOICE = ' '
@ 19,25 say 'Enter your choice : ' get CHOICE picture '!'
read

** Case Statement

do case
case CHOICE = '1'
do ADDMENU

case CHOICE = '2'
do EDORDER

case CHOICE = '3'
do EDACTY

case CHOICE = '4'
do PRTMENU

case CHOICE = '5'
do CHKMENU

case CHOICE = 'Q'
clear all
return

```

endcase

enddo

```

** ADDMENU.PRG
** Menu for Adding Order, Product and Activity Information as well as
**      Planned, Revised plan and Actual weeks

      clear
      set talk off

** Initialising the variables

      CHOICE = 0

do while .T.

** Opening weeks, order, activity and product information databases

      select A
      use WEEKS index WEEKS

      select B
      use ORDINFO index ORDINFO

      select C
      use ACTDAT index ACTDAT

      select D
      use PROD_ACT index PROD_ACT

** Displaying the options available for adding information

      clear
      set color to w+
      @ 2,32 say 'OHS'
      @ 3,30 say 'Add Menu'
      set color to w
      @ 5,20 say '1. Add Order Information and Planned Weeks'
      @ 7,20 say '2. Add Revised Planned Weeks'
      @ 9,20 say '3. Add Actual Weeks'
      @ 11,20 say '4. Add Activity Information'
      @ 13,20 say '5. Add Products and Time Lags'
      @ 15,20 say '0. Exit to previous Menu'
      CHOICE = 0
      @ 18,25 say 'Enter your Choice :' get CHOICE picture '9'
      read

do case

      case CHOICE = 1
      do ADDORD

      case CHOICE = 2
      do ADREVPL

      case CHOICE = 3
      do ADACTLWK

      case CHOICE = 4
      do ADDACT

      case CHOICE = 5
      do ADDPROD

```

```
case CHOICE = 0  
    exit  
endcase
```

```
enddo  
clear  
close database  
return
```

```

** CHKMENU.PRG
** Menu to output checklists for Order, Product and Activity Files

clear
set talk off

** Initialising the variables

CHOICE = 0
XDEPT = ' '

do while .T.

** Opening weeks, order, activity and product information databases

select A
  use WEEKS index WEEKS

select B
  use ORDINFO index ORDINFO

select C
  use ACTDAT index ACTDAT

select D
  use PROD_ACT index PROD_ACT

** Displaying the options available for adding information

clear
set color to w+
@ 4,36 say 'OHS'
@ 5,30 say 'Checklist Menu'
set color to w
@ 7,20 say '1. Order Information'
@ 9,20 say '2. Activity Information'
@ 11,20 say '3. Product Information'
@ 13,20 say '0. Exit to Previous Menu'
CHOICE = 0
@ 16,25 say 'Enter your Choice :' get CHOICE picture '9'
read

  ** get department for which to print checklist

  do while .T.
    @ 23,0 clear
    @ 23,0 say 'Enter Department Name :' get XDEPT picture '!!!'
    read
    select B
    locate for DEPT = XDEPT
    if eof()
      @ 24,0 say 'Given Department name not found, enter name again'
      loop
    endif
    @ 23,0 clear
    exit
  enddo

** set printer on

@ 23,0

```

```
wait 'Set printer on ... and press any key' to KEY
set print on
@ 23,0 clear
@ 23,0 say 'Printing is going on.... '
```

```
do case
```

```
  case CHOICE = 1
```

```
    report form ORDCKLST for DEPT = XDEPT to print
```

```
  case CHOICE = 2
```

```
    report form PRDCKLST for substr(ACTY_NO, 1, 2) = XDEPT to print
```

```
  case CHOICE = 3
```

```
    report form ACTCKLST for substr(ACTY_NO, 1, 2) = XDEPT to print
```

```
  case CHOICE = 0
```

```
    exit
```

```
  set print off
```

```
endcase
```

```
enddo
```

```
clear
```

```
close database
```

```
return
```

```
** PRTMENU.PRG
** Menu to choose an option available for printing reports
```

```
clear
set talk off
```

```
** initialising variables
```

```
CHOICE = 0
XDEPT = ' '
```

```
do while .T.
```

```
** opening weeks, order and activity information databases
```

```
select A
use WEEKS index WEEKS
```

```
select B
use ORDINFO index ORDINFO
```

```
select C
use ACTDAT index ACTDAT
```

```
** Displaying the list of items in the Print Menu
```

```
clear
set color to w+
@ 5,30 say 'Print Menu'
set color to w
@ 7,25 say '1. Completed Activities Report'
@ 9,25 say '2. Future Activities Report'
@ 11,25 say '3. Reminder for Pending Activities'
@ 13,25 say '4. Performance of Delivery Times'
@ 15,25 say '5. Report of Delay in Activities'
@ 17,25 say '6. Report of Loss Suffered'
@ 19,25 say '0. Exit'
CHOICE = 0
@ 21,25 say 'Enter your choice : ' get CHOICE picture '9'
read
```

```
** Case Statement
```

```
do case
  case CHOICE = 1
    do PRICMPWK

  case CHOICE = 2
    do PRIFTRWK

  case CHOICE = 3
    do PRIREMDR

  case CHOICE = 4
    do PRIDVTN

  case CHOICE = 5
    do PRIDELAY

  case CHOICE = 6
```

```

do while .T.
  @ 23,0 clear
  @ 23,0 say 'Enter Department Name :' get XDEPT picture '!!!'
  read
  select B
  locate for DEPT = XDEPT
  if eof()
    @ 24,0 say 'Given Department name not found, enter name again'
    loop
  endif
  @ 23,0 clear
  @ 23,0
  wait 'Set printer on ... and press any key' to KEY
  set print on
  @ 23,0 clear
  @ 23,0 say 'Printing is going on...'
  report form PRILOSS for DEPT = XDEPT .and. D_LOSS > '0'
plain noobject to print
  set print off
  exit
enddo

case CHOICE = 0
  clear
  close database
  return

endcase

enddo

```

```

** ADDORD.PRG
** add screen
** Displays a screen for getting information about all the orders

clear
set talk off

** Initialising the variables

XOI_NO = space(11)
XOI_DT = space(6)
XCNAME = space(25)
XPNAME = space(10)
XOR_TY = space(4)
XVAL   = 0.00
XQTY   = 0
STATUS = ' '
VALID  = ' '
RCPTWK = ' '
DLRYWK = ' '
XW_G_TY = ' '
XDEPT  = ' '
XPLAN_WK = ' '
DATECHK = 1

** Painting headings

set color to w+
@ 1,20 say 'Order Information - Add screen'
set color to w
@ 2,3 say 'O.I.' + space(6) + 'O.I.' + 'Customer' + space(15) + 'Product' +
'Order'
@ 2,65 say 'Quan' + 'Wk-gen'
@ 3,3 say 'No.' + space(7) + 'Date' + space(7) + 'Name' + space(19) + 'Name' +
'Type'
@ 3,60 say 'Dept' + 'tity' + 'Value' + 'Type'
ROW = 5
select B

do while .T.

@ ROW,0 get XOI_NO picture '!!!!!!'
@ ROW,12 get XOI_DT picture '999999'
@ ROW,19 get XCNAME picture '!!!!!!!!!!!!!!!!!!!!!!'
@ ROW,45 get XPNAME picture '!!!!!!'
@ ROW,56 get XOR_TY picture '!!!!'
@ ROW,61 get XDEPT picture '!!!'
@ ROW,65 get XQTY picture '99'
@ ROW,69 get XVAL picture '9999.99'
@ ROW,77 get XW_G_TY picture '!'
read
@ 21,0 clear

** if no customer name is given, check for going out **

if len(trim(XCNAME)) = 0
ANS = 'N'
@ 24,0 say 'Do you want to continue ? (Y/N):' get ANS picture '!'
read
if ANS <> 'Y'
close database

```

```

        return
    else
        @ 23,0 clear
        loop
    endif
endif

* check if O.I. number already exists in the order database

locate for OI_NO = XOI_NO
if .not. eof()
    @ 21,0 say 'O.I. no entered already exists, enter a new no.'
    loop
endif

do CHKDATE with XOI_DT, DATECHK
    if DATECHK = 0
        @ 24,0 say 'Error in date, enter new date in yymmdd format'
        loop
    endif

* validate week generation type should be 'A' or 'M' only

if .not. (XW_G_TY = 'A' .or. XW_G_TY = 'M')
    @ 24,0 clear
    @ 24,0 say 'Error in week gen. type, give only A or M'
    loop
endif

* to validate product name, order type and department

do VALIDORD with XPNAME, XOR_TY, XDEPT, VALID
    if VALID = 'L'
        loop
    else
        exit
    endif
endif

enddo

** add record to database **

select B
append blank
replace OI_NO with XOI_NO, OI_DATE with XOI_DT, CNAME with XCNAME
replace PNAME with XPNAME, ORD_TYPE with XOR_TY, QTY with XQTY
replace DEPT with XDEPT, VALUE with XVAL, WK_GEN_TYP with XW_G_TY

** see for week generation type
* if 'A' ie automatic, calculate planned weeks with time lag and display

if XW_G_TY = 'A'

    do while .T.
        @ 23,0 say 'Enter O.I. Receipt week no : ' get RCPTWK picture '999'
        @ 23,41 say 'Enter O.I. Delivery week no: ' get DLRYWK picture '999'
        read
        @ 24,0 clear
        if val(substr(RCPTWK, 2, 2)) > 53
            @ 24,0 say 'Error in Receipt week, give last 2 digits <= 53'
            loop
        endif
    enddo

```

```

endif
if val(substr(DLRYWK, 2, 2)) > 53
    @ 24,0 say 'Error in Delivery week, give last 2 digits <= 53'
    loop
endif
exit
enddo

@ 24,0 say 'Weeks are being calculated'
@ 6,1 say 'Acty no'+space(5)+'Acty - Name'+space(6)+'Plan-Wk'
@ 6,41 say 'Acty no'+space(5)+'Acty - Name'+space(6)+'Plan-Wk'
ROW = 7
COL = 2
select D

do while .T.

    PL_WK = ' '
    VALPL_WK = val(RCPTWK) + TIME_LAG
    PL_WK = str(VALPL_WK, 3)

    do NEXTYRWK with PL_WK

        @ ROW, COL say D->ACTY_NO
        select C
        seek D->ACTY_NO
        @ ROW, COL+7 say substr(ACT_NAME, 1, 20)
        @ ROW+1,COL+7 say substr(ACT_NAME, 21, 40)
        @ ROW, COL+30 say PL_WK

        if len(trim(ACT_NAME)) > 20
            ROW = ROW + 2
        else
            ROW = ROW + 1
        endif
        if ROW > 21
            ROW = 7
            COL = COL + 40
        endif

        select A
        append blank
        replace DI_NO with XDI_NO, ACTY_NO with D->ACTY_NO
        replace PLAN_WK with PL_WK, PLAN_DATE with MTDAY

        select D
        skip
        if trim(PNAME)<>trim(XPNAME) .or. eof()
            skip - 1
            if len(trim(C->ACT_NAME)) > 20
                @ ROW-2,COL+30 say DLRYWK
            else
                @ ROW-1,COL+30 say DLRYWK
            endif
            select A
            replace PLAN_WK with DLRYWK
            exit
        endif

    enddo
endif

```

* if week generation type is 'M' ie manual, get planned weeks from user

```
if XW_G_TY = 'M'
  ROW = 7
  COL = 2
  select D
  go top
  locate for trim(PNAME) = trim(XPNAME)
  @ 6,1 say 'Acty no'+space(5)+'Acty - Name'+space(6)+'Plan-Wk'
  @ 6,41 say 'Acty no'+space(5)+'Acty - Name'+space(6)+'Plan-Wk'

  do while .T.

    select C
    seek D->ACTY_NO
    @ ROW,COL say ACTY_NO
    @ ROW, COL+7 say substr(ACT_NAME, 1, 20)
    @ ROW+1,COL+7 say substr(ACT_NAME, 21, 40)
    select A

    do while .T.
      @ ROW,COL+30 get XPLAN_WK picture '999'
      read
      @ 24,0 clear
      do CHKWEEK with XPLAN_WK, STATUS
        if STATUS = 'E'
          @ 24,3 say 'Error in planned week, give last 2 digits <= 53'
          loop
        endif
        if len(trim(XPLAN_WK)) < 3
          @ 24,0 say 'Error in given week, key in all 3 digits'
          loop
        endif
      exit
    enddo

    if len(trim(C->ACT_NAME)) > 20
      ROW = ROW + 2
    else
      ROW = ROW + 1
    endif
    if ROW > 21
      ROW = 7
      COL = COL + 40
    endif

    select A
    append blank
    replace OI_NO with XOI_NO, ACTY_NO with D->ACTY_NO
    replace PLAN_WK with XPLAN_WK, PLAN_DATE with MTODAY
    select D
    skip
    if trim(PNAME) <> trim(XPNAME) .or. eof()
      exit
    endif
  enddo
endif
```

** ask if planned weeks need to be changed

* if yes, go back and allow user to enter them manually else return

```

ANS = ' '
@ 23,0 clear
do while .not. (ANS = 'Y' .or. ANS = 'N')
    @ 24,0 say 'Do you want the above planned weeks to be stored in;
        database?(Y/N):' get ANS picture '!'
    read
enddo
if ANS <> 'Y'
    ROW = 7
    COL = 2
    select A
    seek B->OI_NO

    do while .T.

        @ ROW,COL say ACTY_NO
        select C
        seek A->ACTY_NO
        @ ROW, COL+7 say substr(ACT_NAME, 1, 20)
        @ ROW+1,COL+7 say substr(ACT_NAME, 21, 40)

        select A
        XPLAN_WK = PLAN_WK

        do while .T.
            @ ROW,COL+30 get XPLAN_WK picture '999'
            read
            @ 24,0 clear
            do CHKWEEK with XPLAN_WK, STATUS
                if STATUS = 'E'
                    @ 24,0 say 'Error in planned week, give last 2 digits <= 53'
                    loop
                endif
                if len(trim(XPLAN_WK)) < 3
                    @ 24,0 say 'Error in given week, key in all 3 digits'
                    loop
                endif
            enddo
            exit

            if len(trim(C->ACT_NAME)) > 20
                ROW = ROW + 2
            else
                ROW = ROW + 1
            endif
            if ROW > 21
                ROW = 7
                COL = COL + 40
            endif

            replace PLAN_WK with XPLAN_WK
            skip
            if OI_NO <> B->OI_NO .or. eof()
                exit
            endif
        enddo
        @ 23,0 clear
        @ 24,0 say 'Final Planned Weeks added in database'

    endif
endif

```

```
clear  
close database  
return
```

```

** ADREVPL.PRG
** Program to add the revised planned weeks in weeks database

clear
set talk off

** initializing the variables

XOI_NO   = space(11)
XACTY_NO = space(5)
XPLAN_WK = ' '
XREV_WK   = ' '
STATUS    = ' '

set color to w+
@ 1,23 say 'Adding Screen - Revised Plan Weeks'
set color to w

** ask for OI no whose planned weeks are to be revised and seek from database
do while .T.

XOI_NO = space(11)
@ 2,1 say 'Enter O.I. no for whom to revise plan weeks:'
get XOI_NO picture '!!!!!!'
read

select B
go top
seek XOI_NO
if eof()
    @ 23,0 say 'O.I. no. entered not found, enter new no.'
    ANS = 'N'
    @ 24,0 say 'Do you want to give new O.I. no. ?(Y/N):' get ANS picture '!'
    read
    if ANS = 'Y'
        @ 23,0 clear
        loop
    else
        clear
        close database
        return
    endif
endif

@ 3,1 say 'Cust. name: '+CNAME
@ 3,37 say 'Product name: '+PNAME
@ 4,1 say 'Acty'+space(5)+'Activity'+space(11)+'Plan'+ ' '+ 'Revd.'
@ 5,2 say 'no.'+space(7)+'name'+space(13)+'wk'+space(3)+'wk'

do while .T.
    select A
    seek XOI_NO
    ROW = 6
    COL = 1

    do while .T.

        * getting the revised plan weeks
        @ ROW,COL say ACTY_NO
        select C

```

```

seek A->ACTY_NO
@ ROW,COL+7 say substr(ACT_NAME, 1, 20)
@ ROW+1,COL+7 say substr(ACT_NAME, 21, 40)

select A
@ ROW,COL+28 say PLAN_WK
@ ROW,COL+33 say REV_PL_WK
STATUS = ' '

if val(ACT_CMP_WK) = 0
  XREV_WK = REV_PL_WK
  do while .T.
    @ ROW,COL+33 get XREV_WK picture '999'
    read
    @ 23,0 clear
    do CHKWEEK with XREV_WK, STATUS
    if STATUS = 'E'
      ANS = 'N'
      @ 23,0 say 'Error in revised week, give last 2 digits <= 53'
      @ 24,0 say 'Do you want to continue?(Y/N):' get ANS picture '!'
      read
      if ANS <> 'Y'
        STATUS = 'X'
        exit
      else
        @ 23,0 clear
        loop
      endif
    endif
    replace REV_PL_WK with XREV_WK, REVD_DATE with MTODAY
    exit
  enddo
endif

if STATUS = 'X'
  XREV_WK = ' '
  @ ROW,COL+33 say XREV_WK
  exit
endif

* incrementing the rows and print headings if row > 21
if len(trim(C->ACT_NAME)) > 20
  ROW = ROW + 2
else
  ROW = ROW + 1
endif
if ROW > 21
  ROW = 6
  COL = COL + 40
  @ ROW-2,COL say 'Acty'+space(5)+'Activity'+space(11);
  @ ROW-2,COL+33 say 'Plan'+space(11)+'Revd.'
  @ ROW-1,COL+1 say 'no.'+space(7)+'name'+space(13)+'wk'+space(3)+'wk'
endif

skip
if OI_NO <> XOI_NO .or. eof()
  exit
endif
enddo

* ask if present weeks can be stored weeks database

```

```

@ 23,0 clear
ANS = ' '
do while .not. (ANS = 'Y' .or. ANS = 'N')
    @ 23,0 say 'Shall the above listed revised weeks be updated to the
        weeks database ?(Y/N):';
    get ANS picture '!';
    read
enddo
if ANS = 'Y'
    @ 23,0 clear
    exit
endif
if ANS = 'N'
    @ 6,0 clear
    loop
endif
enddo

* ask for revising plans for another order
ANS = ' '
@ 23,0 clear
do while .not. (ANS = 'Y' .or. ANS = 'N')
    @ 23,0 say 'Revised weeks updated to the database'
    @ 24,0 say 'Do you want to revise planned weeks for another order
?(Y/N):';
    get ANS picture '!';
    read
enddo
if ANS = 'Y'
    @ 3,0 clear
    loop
endif
if ANS = 'N'
    close database
    return
endif
enddo

```

```

** ADDPROD.PRG
** add screen
** Displays the screen for getting the products and time lags

clear
set talk off

** Initialising the variables

DEPT = ' '
XPNAME = space(10)

** Painting headings and getting dept. and product name

set color to w+
@ 1,15 say 'Product and Activity time-lag - adding screen'
set color to w

do while .T.
  @ 2,1 say 'Name of Dept:' get DEPT picture '!!'
  read
  select C
  go top
  locate for substr(ACTY_NO, 1, 2) = DEPT
  if eof()
    @ 24,0 say 'No activities found for this dept. in activity file, enter;
               new one'
    loop
  endif
enddo

do while .T.
  @ 2,19 say 'Product Name:' get XPNAME picture '!!!!!!!!!!!!'
  read
  select D
  go top
  locate for PNAME = XPNAME
  if .not. eof()
    @ 24,0 say 'Product name entered already exists, enter new name'
    loop
  endif
enddo

XPRDOLD = XPNAME
@ 3,1 say 'Acty'+ ' '+'Activity'+space(10)+'Reqd'+ ' '+'Time'
@ 4,2 say 'no.'+ ' '+'name'+space(13)+'Y/N'+ ' '+'Lag'

RDW = 5
COL = 1
select C

do while .T.

```

```

** ADDPROD.PRG
** add screen
** Displays the screen for getting the products and time lags

clear
set talk off

** Initialising the variables

DEPT = ' '
XPNAME = space(10)

** Painting headings and getting dept. and product name

set color to w+
@ 1,15 say 'Product and Activity time-lag - adding screen'
set color to w

do while .T.
    @ 2,1 say 'Name of Dept:' get DEPT picture '!!'
    read
    select C
    go top
    locate for substr(ACTY_NO, 1, 2) = DEPT
    if eof()
        @ 24,0 say 'No activities found for this dept. in activity file, enter
                    new one'
        loop
    endif
enddo

do while .T.
    @ 2,19 say 'Product Name:' get XPNAME picture '!!!!!!!!!!!!'
    read
    select D
    go top
    locate for PNAME = XPNAME
    if .not. eof()
        @ 24,0 say 'Product name entered already exists, enter new name'
        loop
    endif
enddo

XPRDOLD = XPNAME
@ 3,1 say 'Acty'+ ' '+'Activity'+space(10)+'Reqd'+ ' '+'Time'
@ 4,2 say 'no.'+ ' '+'name'+space(13)+'Y/N'+ ' '+'Lag'

ROW = 5
COL = 1
select C

do while .T.

** ask if activity required, get time lag if affirmative

REQD = ' '
XTIME_LAG = 0
@ ROW,COL say ACTY_NO
@ ROW,COL+5 say substr(ACT_NAME,1,20)
@ ROW+1,COL+5 say substr(ACT_NAME,21,40)
@ ROW,COL+27 get REQD picture '!'

```

```

read

if REQD = 'Y'
  @ ROW,COL+33 get XTIME_LAG picture '99'
  read
  select D
  append blank
  replace PNAME with XPNAME, ACTY_NO with C->ACTY_NO
  replace TIME_LAG with XTIME_LAG
endif
if len(trim(C->ACT_NAME)) > 20
  ROW = ROW + 2
else
  ROW = ROW + 1
endif
if ROW > 21
  ROW = 5
  COL = COL + 41
  @ ROW-2,COL say 'Acty'+      '+'Activity'+space(10)+'Reqd'+      '+'Time'
  @ ROW-1,COL say 'no.'+      '+'name'+space(13)+'Y/N'+      '+'Lag'
endif

select C
skip
if substr(ACTY_NO, 1, 2) <> DEPT .or. eof()
  exit
endif
enddo

* ask if time lags can be stored
ANS = ' '
@ 23,0 clear
do while .not. (ANS = 'Y' .or. ANS = 'N')
  @ 24,0 say 'Do you want the above time lags to be stored ? (Y/N):';
  get ANS picture '!'
  read
enddo
if ANS = 'Y'
  clear
  close database
  return
else

** if No then go back and get time lags again

do while .T.
  @ 2,0 clear
  select D
  locate for PNAME = XPRDOLD
  XPNAME = PNAME
  @ 2,19 say 'Product Name:' get XPNAME picture '!!!!!!!!!!!!'
  read
  if XPRDOLD <> XPNAME
    locate for PNAME = XPNAME
    if .not. eof()
      @ 24,0 say 'Product name entered already exists, enter new one'
      loop
    endif
  endif
  @ 3,1 say 'Acty'+      '+'Activity'+space(10)+'Reqd'+      '+'Time'
  @ 4,2 say 'no.'+      '+'name'+space(13)+'Y/N'+      '+'Lag'

```

```
ROW = 5
COL = 1
```

```
do while .T.
```

```
  @ ROW,COL say ACTY_NO
  @ ROW,COL+5 say substr(C->ACT_NAME,1,20)
  @ ROW+1,COL+5 say substr(C->ACT_NAME,21,40)
  XTIME_LAG = TIME_LAG
  @ ROW,COL+33 get XTIME_LAG picture '99'
  read
  replace PNAME with XPNAME, TIME_LAG with XTIME_LAG
  if len(trim(C->ACT_NAME)) > 20
    ROW = ROW + 2
  else
    ROW = ROW + 1
  endif
  if ROW > 21
    ROW = 5
    COL = COL + 41
    @ ROW-2,COL say 'Acty'+ ' '+ 'Activity'+space(10)+'Reqd'+ ' ' ;
    @ ROW-2,COL say 'Time'+ ' '+ 'Time'
    @ ROW-1,COL say ' no.'+ ' '+ 'name'+space(13)+'Y/N'+ ' ' '+ 'Lag'
  endif
```

```
  skip
  if PNAME <> XPNAME
    exit
  endif
```

```
enddo
@ 23,0 say 'Product Details Added'
exit
```

```
enddo
endif
```

```
clear
close database
return
```

```

** ADDACT.PRG
** add screen
** Displays the screen for entering information about the activities

clear
set talk off

** Initialising the variables

XACTY_NO      = ' '
XACT_NAME     = space(40)
XDEPT_RESP1  = space(4)
XDEPT_RESP2  = space(4)
XDEPT_RESP3  = space(4)

** Painting headings

set color to w+
@ 1,15 say 'Activity and Responsibility - adding screen'
set color to w
@ 2,1 say 'Activity'
@ 2,18 say 'Activity'
@ 2,45 say 'Department Responsible'
@ 3,1 say 'Number'
@ 3,18 say 'Name'
@ 3,45 say '1st      2nd      3rd'

ROW = 5
select C
go top

do while .T.

@ ROW,2 get XACTY_NO picture '!!!999'
@ ROW,10 get XACT_NAME picture '!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!'
@ ROW,52 get XDEPT_RESP1 picture '!!!!'
@ ROW,61 get XDEPT_RESP2 picture '!!!!'
@ ROW,71 get XDEPT_RESP3 picture '!!!!'
read
@ 23,0 clear

** if no activity name is given, check for going out **

if len(trim(XACT_NAME)) = 0
  ANS = 'N'
  @ 23,0 say 'Do you want to continue ? (Y/N):' get ANS picture '!'
  read
  if ANS <> 'Y'
    clear
    close database
    return
  else
    @ 23,0 clear
    loop
  endif
endif

locate for ACTY_NO = XACTY_NO
if .not. eof()
  @ 20,0 say 'Activity number entered already exists, enter a new no.'
  loop

```

```

endif

** add record to database **

append blank
replace ACTY_NO with XACTY_NO, ACT_NAME with XACT_NAME
replace DEPT_RESP1 with XDEPT_RESP1, DEPT_RESP2 with XDEPT_RESP2,,
      DEPT_RESP3 with XDEPT_RESP3

** Initialising the variables

XACTY_NO      = ' '
XACT_NAME     = space(40)
XDEPT_RESP1   = space(4)
XDEPT_RESP2   = space(4)
XDEPT_RESP3   = space(4)

PROW=ROW
ROW = ROW + 2
if ROW > 20
  ROW = 5
  @ 5,0 clear
  @ PROW,2 say ACTY_NO
  @ PROW,10 say ACT_NAME
  @ PROW,52 say DEPT_RESP1
  @ PROW,61 say DEPT_RESP2
  @ PROW,71 say DEPT_RESP3
endif
if ROW > 10
  @ 19,0 clear
endif
enddo

close database
return

```

```

** ADACTLWK.PRG
** Program to enter the actual completion weeks in weeks database

clear
set talk off

** initializing the variables

XOI_NO    = space(11)
XACTY_NO  = space(5)
XPLAN_WK  = ' '
XREV_WK   = ' '
XACT_WK   = ' '
STATUS    = ' '

** ask for OI no whose actual weeks are to be entered

do while .T.

set color to w+
@ 1,25 say 'Adding Screen - Actual Weeks'
set color to w
XOI_NO = space(11)
@ 2,0 say 'Enter O.I. no for whom to enter actual weeks: ';
get XOI_NO picture '!!!!!!!!!!!!'
read

select B
go top
seek XOI_NO
if eof()
@ 23,0 say 'O.I. no. entered not found in order information file'
ANS = 'N'
@ 24,0 say 'Do you want to give new O.I. no. ?(Y/N):' get ANS picture '!'
read
if ANS = 'Y'
@ 23,0 clear
loop
else
clear
close database
return
endif
endif

@ 3,0 say 'Cust. name: '+CNAME
@ 3,38 say 'Product name: '+PNAME

@ 4,0 say 'Acty'+space(4)+'Activity'+space(9)+'Plan'+ ' '+Rev'd'+ ' '+Act1
@ 5,2 say 'no.'+space(6)+'name'+space(11)+'wk'+ ' '+wk'+ ' '+wk

do while .T.

```

```

** ADACTLWK.PRG
** Program to enter the actual completion weeks in weeks database

clear
set talk off

** initializing the variables

XOI_NO    = space(11)
XACTY_NO  = space(5)
XPLAN_WK  = ' '
XREV_WK   = ' '
XACT_WK   = ' '
STATUS    = ' '

** ask for OI no whose actual weeks are to be entered

do while .T.

set color to w+
@ 1,25 say 'Adding Screen - Actual Weeks'
set color to w
XOI_NO = space(11)
@ 2,0 say 'Enter O.I. no for whom to enter actual weeks:'
      get XOI_NO picture '!!!!!!!!!!'
read

select B
go top
seek XOI_NO
if eof()
    @ 23,0 say 'O.I. no. entered not found in order information file'
    ANS = 'N'
    @ 24,0 say 'Do you want to give new O.I. no. ?(Y/N):' get ANS picture '!'
    read
    if ANS = 'Y'
        @ 23,0 clear
        loop
    else
        clear
        close database
        return
    endif
endif

@ 3,0 say 'Cust. name: '+CNAME
@ 3,38 say 'Product name: '+PNAME

@ 4,0 say 'Acty'+space(4)+'Activity'+space(9)+'Plan'+ ' '+ 'Rev'd'+ ' '+ 'Act1'
@ 5,2 say 'no.'+space(6)+'name'+space(11)+'wk'+ ' '+ 'wk'+ ' '+ 'wk'

do while .T.
    select A
    seek XOI_NO
    ROW = 6
    COL = 0

    do while .T.

        * getting the actual completion weeks for present OI no.
        @ ROW,COL say ACTY_NO

```

```

select C
seek A->ACTY_NO
@ ROW,COL+6 say substr(ACT_NAME, 1, 20)
@ ROW+1,COL+6 say substr(ACT_NAME, 21, 40)

select A
@ ROW,COL+26 say PLAN_WK
@ ROW,COL+31 say REV_PL_WK
@ ROW,COL+36 say ACT_CMP_WK
STATUS = ''
if val(ACT_CMP_WK) = 0
  XACT_WK = ACT_CMP_WK
  do while .T.
    @ ROW,COL+36 get XACT_WK picture '999'
    read
    @ 23,0 clear
    if val(XACT_WK) = 0
      @ 23,0 say 'Actual week is blank, enter all 3 digits'
      loop
    endif

    do CHKWEEK with XACT_WK, STATUS
      if STATUS = 'E'
        ANS = 'N'
        @ 23,0 say 'Error in actual week, give last 2 digits <= 53'
        @ 24,0 say 'Do you want to continue?(Y/N):' get ANS picture
        read
        if ANS <> 'Y'
          STATUS = 'X'
          exit
        else
          @ 23,0 clear
          loop
        endif
      endif
      replace ACT_CMP_WK with XACT_WK, ACTL_DATE with MTODAY
      exit
    enddo
  endif

  if STATUS = 'X'
    XACT_WK = ''
    @ ROW,COL+36 say XACT_WK
    exit
  endif

  if len(trim(C->ACT_NAME)) > 20
    ROW = ROW + 2
  else
    ROW = ROW + 1
  endif
  if ROW > 21
    ROW = 6
    COL = COL + 41
    @ ROW-2,COL say 'Acty'+space(4)+'Activity'+space(9)+'Plan'+';
    '+ 'Rev'+ ' '+ 'Actl'
    @ ROW-1,COL+1 say 'no.'+space(6)+'name'+space(12)+'wk'+ ' '+ 'wk'+ ' ';
    '+ 'wk'
  endif

  skip

```

```

    if OI_NO <> XOI_NO .or. eof()
        exit
    endif
enddo

* ask if actual weeks can be stored in weeks database
@ 23,0 clear
ANS = ' '
do while .not. (ANS = 'Y' .or. ANS = 'N')
    @ 23,0 say 'Shall the above listed actual weeks be updated to the weeks
    database ?(Y/N):';
    get ANS picture '! '
    read
enddo
if ANS = 'Y'
    exit
endif
if ANS = 'N'
    @ 6,0 clear
    loop
endif
enddo

* ask if actual weeks for another order are to be entered
ANS = ' '
@ 23,0 clear
@ 23,0 say 'Actual weeks updated to the database'

do while .not. (ANS = 'Y' .or. ANS = 'N')
    @ 24,0 say 'Do you want to enter actual weeks for another order
    ?(Y/N):';
    get ANS picture '! '
    read
enddo
if ANS = 'Y'
    @ 3,0 clear
    loop
endif
if ANS = 'N'
    close database
    return
endif
enddo

```

```

** EDORDER.PRG
** Screen for editing the Order Information in ORDINFO File

clear
set talk off

** Initialising the variables

XOI_NO = space(11)
XOI_DT = space(6)
XCNAME = space(25)
XPNAME = space(10)
XOR_TY = space(4)
XVAL   = 0.00
XQTY   = 0
XDEPT  = ' '
VALID  = ' '
REM1   = space(70)
REM2   = space(70)
REM3   = space(70)
REM4   = space(70)
REM5   = space(70)

** Opening order information database

select A
  use ORDINFO index ORDINFO
select D
  use PROD_ACT index PROD_ACT

** Painting headings

do while .T.

set color to w+
@ 1,20 say 'Order Information - Editing screen'
@ 2,1  say 'N-Next      P-Previous      E-Edit      S - Seek      X-E'
set color to w
do while .T.
  @ 3,1 say 'Enter Dept. Name:' get XDEPT picture '!!'
  read
  select A
  go top
  locate for DEPT = XDEPT
  if eof()
    @ 24,0 say 'No order information found for this dept., enter new one'
    loop
  endif
  @ 23,0 clear
  exit
enddo

@ 5,5  say 'O.I. no   : '+OI_NO
@ 5,35 say 'O.I. Date: '+OI_DATE
@ 5,58 say 'Department: '+DEPT
@ 7,5  say 'Cust. Name: '+CNAME
@ 7,58 say 'Prod. Name: '+PNAME
@ 9,5  say 'Order Type: '+ORD_TYPE
@ 9,35 say 'Quantity : '+str(QTY, 2)
@ 9,58 say 'Value      : '+str(VALUE, 7, 2)

```

```
do while .T.
```

```
** Using the edit options
```

```
COMMAND = ' '
```

```
@ 1,1 say 'Give Command:' get COMMAND picture '!'
```

```
read
```

```
@20,0 clear
```

```
do case
```

```
case COMMAND = 'N'
```

```
if (DEPT = XDEPT .or. .not. eof())
```

```
skip
```

```
endif
```

```
if (DEPT <> XDEPT .or. eof())
```

```
@ 23,0 say 'End of file, no more records after this !'
```

```
else
```

```
@ 5,17 say OI_NO
```

```
@ 5,46 say OI_DATE
```

```
@ 5,70 say DEPT
```

```
@ 7,17 say CNAME
```

```
@ 7,70 say PNAME
```

```
@ 9,17 say ORD_TYPE
```

```
@ 9,46 say str(QTY, 2)
```

```
@ 9,70 say str(VALUE, 7, 2)
```

```
@ 11,0 clear
```

```
endif
```

```
case COMMAND = 'P'
```

```
if (DEPT = XDEPT .or. .not. bof())
```

```
skip -1
```

```
endif
```

```
if (DEPT <> XDEPT .or. bof())
```

```
@ 23,0 say 'You are already at the beginning of file !'
```

```
else
```

```
@ 5,17 say OI_NO
```

```
@ 5,46 say OI_DATE
```

```
@ 5,70 say DEPT
```

```
@ 7,17 say CNAME
```

```
@ 7,70 say PNAME
```

```
@ 9,17 say ORD_TYPE
```

```
@ 9,46 say str(QTY, 2)
```

```
@ 9,70 say str(VALUE, 7, 2)
```

```
@ 11,0 clear
```

```
endif
```

```
case COMMAND = 'S'
```

```
do while .T.
```

```
@ 1,20 say 'Give O.I. no:' get XOI_NO picture '!!!!!!!!!!!!'
```

```
read
```

```
seek XOI_NO
```

```
if eof()
```

```
@ 23,0 say '* Given O.I. no. not found *'
```

```
ANS = 'N'
```

```
@ 24,0 say 'Do you want to give new O.I. no. ? (Y/N):';
```

```
get ANS picture '!'
```

```
read
```

```
if ANS <> 'Y'
```

```
clear
```

```
close database
```

```

        return
    else
        @ 23,0 clear
    loop
    endif
endif

@ 5,17 say OI_NO
@ 5,46 say OI_DATE
@ 5,70 say DEPT
@ 7,17 say CNAME
@ 7,70 say PNAME
@ 9,17 say ORD_TYPE
@ 9,46 say str(QTY, 2)
@ 9,70 say str(VALUE, 7, 2)
@ 11,0 clear
exit
enddo

case COMMAND = 'E'
    XOI_NO = OI_NO
    XOI_DT = OI_DATE
    XCNAME = CNAME
    XPNAME = PNAME
    XOR_TY = ORD_TYPE
    XVAL = VALUE
    XQTY = QTY
    XDEPT = DEPT
    OINOLD = OI_NO

    do while .T.

        @ 5,17 get XOI_NO picture '!!!!!!!!!!!!'
        @ 5,46 get XOI_DT picture '999999'
        @ 5,70 get XDEPT picture '!!'
        @ 7,17 get XCNAME picture '!!!!!!!!!!!!!!!!!!!!!!!!!!!!'
        @ 7,70 get XPNAME picture '!!!!!!!!!!!!'
        @ 9,17 get XOR_TY picture '!!!!'
        @ 9,46 get XQTY picture '99'
        @ 9,70 get XVAL picture '9999.99'
        @ 11,5 say 'Actual Completion Remarks:'
        @ 13,5 get REM1
        @ 14,5 get REM2
        @ 15,5 get REM3
        @ 16,5 get REM4
        @ 17,5 get REM5
        read
        @ 21,0 clear

        if OINOLD <> XOI_NO
            locate for OI_NO = XOI_NO
            if .not. eof()
                @ 24,0 say 'O.I. no entered already exists, enter new one'
            loop
            endif
        endif

        do VALIDORD with XPNAME, XOR_TY, XDEPT, VALID
            if VALID = 'L'
                loop
            else
                exit
            endif
        enddo
    loop
enddo

```

```

endif

enddo

XREASON = trim(REM1)+' '+trim(REM2)+' '+trim(REM3)+' '+trim(REM4)+' '+trim

** update record to database **

select A
replace DI_NO with XOI_NO, DI_DATE with XOI_DT, CNAME with XCNAME
replace PNAME with XPNAME, ORD_TYPE with XOR_TY, QTY with XQTY
replace VALUE with XVAL, DEPT with XDEPT, D_REASON with XREASON

@23,0 say 'Record updated !'
case COMMAND = 'X'
    exit
endcase
enddo

ANS = 'N'
@ 24,0 clear
@ 24,0 say 'Do you want to edit information of another order ?(Y/N):'get ANS pict
read
if ANS <> 'Y'
    clear
    close database
    return
endif
@ 4,0 clear
enddo

```

```

** EDACTY.PRG
** Displays the screen for editing the information about the activities

clear
set talk off

** Initialising the variables

XACTY_NO      = ' '
XACT_NAME     = space(40)
XDEPT_RESP1  = space(4)
XDEPT_RESP2  = space(4)
XDEPT_RESP3  = space(4)
DEPT         = ' '

** open Activity database **

use ACTDAT index ACTDAT

** Painting headings

set color to w+
@ 1,20 say 'Activity and Responsibility - Editing screen'
set color to /w
@ 2,29 say 'N - Next      P - Previous      E - Edit      X - Exit'
set color to w
do while .T.
    @ 2,1 say 'Enter Dept. Name:' get DEPT picture '!!'
    read
    go top
    locate for substr(ACTY_NO, 1, 2) = DEPT
    if eof()
        @ 24,0 say 'No activities found for this dept. in activity;
            file, enter new one'
        loop
    endif
    exit
enddo

@ 3,1 say 'Activity'
@ 3,18 say 'Activity'
@ 3,52 say 'Department Responsible'
@ 4,1 say 'Number'
@ 4,18 say 'Name'
@ 4,52 say '1st      2nd      3rd'

ROW = 6

@ ROW,2 say ACTY_NO
@ ROW,10 say ACT_NAME
@ ROW,52 say DEPT_RESP1
@ ROW,61 say DEPT_RESP2
@ ROW,71 say DEPT_RESP3

do while .T.

** using the edit options

COMMAND = ' '
@ 1,1 say 'Give Command:' get COMMAND picture '!'

```

```

read

do case

  case COMMAND = 'N'

    if (substr(ACTY_NO, 1, 2) = DEPT .or. .not. eof())
      skip
      PROW = ROW
      ROW = ROW + 2
      if ROW > 21
        ROW = 6
        @ 6,0 clear
        @ PROW,2 say ACTY_NO
        @ PROW,10 say ACT_NAME
        @ PROW,52 say DEPT_RESP1
        @ PROW,61 say DEPT_RESP2
        @ PROW,71 say DEPT_RESP3
      else
        @ ROW,2 say ACTY_NO
        @ ROW,10 say ACT_NAME
        @ ROW,52 say DEPT_RESP1
        @ ROW,61 say DEPT_RESP2
        @ ROW,71 say DEPT_RESP3
      endif
      if ROW > 10
        @ 20,0 clear
      endif
    else
      @ 23,0 say 'End of file, no more records for this dept. after this'
    endif

  case COMMAND = 'P'

    if (substr(ACTY_NO, 1, 2) = DEPT .or. .not. bof())
      skip - 1
      if substr(ACTY_NO, 1, 2) = DEPT

        PROW = ROW
        ROW = ROW - 2
        if ROW < 6
          ROW = 20
          @ 6,0 clear
          @ PROW,2 say ACTY_NO
          @ PROW,10 say ACT_NAME
          @ PROW,52 say DEPT_RESP1
          @ PROW,61 say DEPT_RESP2
          @ PROW,71 say DEPT_RESP3
        else
          @ ROW,2 say ACTY_NO
          @ ROW,10 say ACT_NAME
          @ ROW,52 say DEPT_RESP1
          @ ROW,61 say DEPT_RESP2
          @ ROW,71 say DEPT_RESP3
        endif
      else
        @ 23,0 say 'You are at beginning of records of this dept.'
      endif
    endif

  case COMMAND = 'E'

```

```
XACTY_NO      = ACTY_NO
XACT_NAME     = ACT_NAME
XDEPT_RESP1   = DEPT_RESP1
XDEPT_RESP2   = DEPT_RESP2
XDEPT_RESP3   = DEPT_RESP3
```

```
@ ROW,2  get XACTY_NO picture '!!999'
@ ROW,10 get XACT_NAME picture '!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!'
@ ROW,52 get XDEPT_RESP1 picture '!!!!'
@ ROW,61 get XDEPT_RESP2 picture '!!!!'
@ ROW,71 get XDEPT_RESP3 picture '!!!!'
read
@ 23,0 clear
```

```
** add record to database **
```

```
replace ACTY_NO with XACTY_NO, ACT_NAME with XACT_NAME
replace DEPT_RESP1 with XDEPT_RESP1, DEPT_RESP2 with XDEPT_RESP2,
        DEPT_RESP3 with XDEPT_RESP3
@ 23,0 say 'Record Updated !'
```

```
case COMMAND = 'X'
    exit
endcase
enddo
clear
close database
return
```

```

** PRIREMDR.PRG
** Program to print a reminder to various departments for activities to be done

clear
set talk off

** initialising the variables

SNO = 0
CNT = 0
ANS = ' '
KEY = ' '
XDIV = ' '
LESS = ' '
XDEPT = ' '
WKNO = ' '
ACTWK = ' '
HEADING= 1

** opening weeks, order and activity information databases

select A
  use weeks index weeks
select B
  use ordinfo index ordinfo
select C
  use actdat index actdat

do while .T.
  @ 21,0 say 'Enter Division Name : ' get XDIV picture '!'
  @ 22,0 say 'Enter Department Name: ' get XDEPT picture '!!!!'
  @ 23,0 say 'Enter week no. from which to print pending activities:'
    get WKNO picture '999'
  read
  if WKNO = ' ' .or. val(substr(WKNO, 2, 2)) > 53
    ANS = 'N'
    @ 24,0 say 'Error in week no. Do you want to continue ?(Y/N):'
    get ANS picture '!'
    read
    if ANS <> 'Y'
      clear
      close database
      return
    else
      @ 24,0 clear
      loop
    endif
  endif

  select B
  locate for substr(DEPT, 1, 1) = XDIV
  if eof()
    @ 24,0 clear
    @ 24,0 say 'No order information found for this division, enter again'
    loop
  endif

  select C
  locate for DEPT_RESP1 = XDEPT .or. DEPT_RESP2 = XDEPT .or.
  DEPT_RESP3 = XDEPT
  if eof()

```

```

    @ 24,0 clear
    @ 24,0 say 'No activities found related to this Department, enter again'
    loop
endif

```

** setting the printer on

```

@ 21,0 clear
ANS = ' '
@ 23,0 say 'Set printer on ... and press any key' get ANS
read
@ 23,0 clear
@ 23,0 say 'Printing is on ...'
set print on
set console off
eject

```

```

SND = 0
WKLESS = 0
HEADING = 1

```

```

do while .T.
    select A
    seek B->OI_NO
    NEWNO = 1

```

```

do while .T.
    LESS = ' '
    select C
    seek A->ACTY_NO

```

```

    if DEPT_RESP1 = XDEPT .or. DEPT_RESP2 = XDEPT .or. DEPT_RESP3 = XDEPT

```

```

        select A
        VLREVD = val(REV_PL_WK)
        if VLREVD = 0
            PLANWK = PLAN_WK
            do DIFFWK with PLANWK, WKNO, LESS
        else
            REVDWK = REV_PL_WK
            do DIFFWK with REVDWK, WKNO, LESS
        endif

```

```

    if LESS = 'N'
        if HEADING = 1

```

** printing the headings

```

        ? space(42)+'PENDING ACTIVITIES FOR '+XDIV+' DIVISION'+
        space(25)+'DATE: '
        ?? MTODAY
        ?
        ? space(42)+'REMINDER TO '+XDEPT+
        ' DEPARTMENT AS ON WEEK NO: '+WKNO
        ?
        ? space(42)+'Acty'+space(43)+"Dept's Resp."+' '+Plan'+
        ' '+Revd'
        ? 'Sno'+space(3)+'O.I. no.'+space(3)+"Customer's name"+
        space(11)+'no.'
        ?? space(7)+'Activity name '+space(22)+'1st 2nd 3rd'+
        space(3)+'wk'+ ' '+wk'

```

```

        LINE = '-----'
        WIDTH = 112
        do LINEPRT with LINE, WIDTH
        ? LINE
        CNT = CNT + 7
        HEADING = 0
    endif
    if NEWNO = 1
        SNO = SNO + 1
        ?
        ? str(SNO, 2)+ ' '+OI_NO+ ' '+B->CNAME
        CNT = CNT + 2
        NEWNO = 0
    endif

    WKLESS = 1
    select C
    seek A->ACTY_NO
    select A
    ?? ACTY_NO+ ' '+C->ACT_NAME+C->DEPT_RESP1+ ' '+C->DEPT_RESP2+ ' '
    ?? C->DEPT_RESP3+ ' '+PLAN_WK+ ' '+REV_PL_WK
    endif
endif
select A
skip
if (WKLESS = 1 .and. LESS = 'N')
    ? space(42)
    CNT = CNT + 1
endif
if OI_NO <> B->OI_NO .or. eof()
    exit
endif
enddo

select B
skip
if CNT > 50
    CNT = 0
    HEADING = 1
    ? LINE
    ?
    eject
endif

if substr(DEPT, 1, 1 ) <> XDIV .or. eof()
    ? LINE
    ?
    exit
endif
enddo
if WKLESS = 0
    @ 21,0 clear
    @ 23,0 say 'No activities found pending for this department'
    @ 24,0 say 'Press any key to continue' get KEY
    read
    close database
    return
endif

set print off
set console on

```

```

do while .not. (ANS = 'Y' .or. ANS = 'N')
  ANS = ' '
  @ 23,0 clear
  @ 23,0 say 'Printing is over.'
  @ 24,0 say 'Do you want to print reminder for another department ?(Y/N):'
  get ANS picture '!'
  read
enddo
if ANS = 'Y'
  @ 23,0 clear
  loop
else
  if ANS = 'N'
    clear
    close database
    return
  endif
endif
enddo
return

```

```

** PRIFTRWK.PRG
** To output the completion weeks for activities to be done in future

clear
set talk off

** initialising the variables

SNO = 0
CNT = 0
ANS = ' '
KEY = ' '
LESS = ' '
XDEPT = ' '
WKNO = ' '
ACTWK = ' '
HEADING= 1

** opening weeks, order and activity information databases

select A
  use weeks index weeks
select B
  use ordinfo index ordinfo
select C
  use actdat index actdat

do while .T.
  @ 22,0 say 'Enter department name : ' get XDEPT picture '!!!'
  @ 23,0 say 'Enter week no. for which you want future weeks report : '
    get WKNO picture '999'
  read
  if WKNO = ' ' .or. val(substr(WKNO, 2, 2)) > 53
    ANS = 'N'
    @ 24,0 say 'Error in week no. Do you want to continue ?(Y/N): '
    get ANS picture '!'
    read
    if ANS <> 'Y'
      clear
      close database
      return
    else
      @ 24,0 clear
      loop
    endif
  endif

  select B
  locate for DEPT = XDEPT
  if eof()
    @ 24,0 clear
    @ 24,0 say 'No order information found for this department, enter again'
    loop
  endif
  exit
enddo

** setting the printer on.

@ 22,0 clear
ANS = ' '

```

```
@ 23,0 say 'Set printer on ... and press any key' get ANS
read
set print on
```

```
SNO = 0
WKLESS = 0
```

```
do while .T.
  select A
  seek B->DI_NO
  NEWNO = 1
```

```
do while .T.
  LESS = ' '
  VLREVD = val(REV_PL_WK)
  if VLREVD = 0
    PLANWK = PLAN_WK
    do DIFFWK with PLANWK, WKNO, LESS
  else
    REVDWK = REV_PL_WK
    do DIFFWK with REVDWK, WKNO, LESS
  endif
```

```
if LESS = 'N'
  if HEADING = 1
```

```
    ** printing the headings
```

```
    ? space(35)+'FUTURE ACTIVITIES REPORT AS OF WEEK No: '+WKNO+
    ? ' for '+XDEPT+' DEPARTMENT'
```

```
    ?
```

```
    ? space(42)+'Acty'+space(43)+"Dept's Resp."+' '+Plan'+
    ? ' '+Revd'
```

```
    ? 'Sno'+space(3)+'O.I. no.'+space(3)+"Customer's name"+
    ? space(11)+'no.'
```

```
    ?? space(7)+'Activity name '+space(22)+'1st 2nd 3rd'+
    ? space(3)+'wk'+ ' '+wk'
```

```
    LINE = '-----'
```

```
    WIDTH = 114
```

```
    do LINEPRT with LINE, WIDTH
```

```
    ? LINE
```

```
    CNT = CNT + 5
```

```
    HEADING = 0
```

```
  endif
```

```
  if NEWNO = 1
```

```
    SNO = SNO + 1
```

```
    ?
```

```
    ? str(SNO, 2)+' '+DI_NO+' '+B->CNAME
```

```
    NEWNO = 0
```

```
    CNT = CNT + 2
```

```
  endif
```

```
  WKLESS = 1
```

```
  select C
```

```
  seek A->ACTY_NO
```

```
  select A
```

```
  ?? ACTY_NO+' '+C->ACT_NAME+C->DEPT_RESP1+' '+C->DEPT_RESP2+' '
```

```
  ?? C->DEPT_RESP3+' '+PLAN_WK+' '+REV_PL_WK
```

```
endif
```

```
skip
```

```
if (WKLESS = 1 .and. LESS = 'N')
```

```

        ? space(42)
        CNT = CNT + 1
    endif
    if OI_NO <> B->OI_NO .or. eof()
        exit
    endif
enddo

select B
skip
if CNT > 50
    CNT = 0
    HEADING = 1
    ? LINE
    ?
    eject
endif

if DEPT <> XDEPT .or. eof()
    exit
endif
enddo
if LESS = 'N'
    ? LINE
    ?
endif
if WKLESS = 0
    @ 22,0 clear
    @ 23,0 say 'No future activities found for this department'
    @ 24,0 say 'Press any key to return' get KEY
    read
    close database
    return
endif

clear
@ 24,0 say ' Printing is over, press any key to return '
set print off
wait '' to KEY
close database
return

```

```

** PRIDVTN.PRG
** To print the performance of delivery times and the deviations from actuals

clear
set talk off

** initialising the variables

SNO = 0
ANS = ' '
KEY = ' '
DATA = 0
XDEPT = ' '
HEADING= 1

** opening weeks, order and activity information databases

select A
  use weeks index weeks
select B
  use ordinfo index ordinfo
select C
  use actdat index actdat

do while .T.
  @ 23,0 say 'Enter department name : ' get XDEPT picture '!!'
  read

  select B
  locate for DEPT = XDEPT
  if eof()
    @ 23,0 clear
    @ 23,0 say 'No weeks information found for this department'
    ANS = 'N'
    @ 24,0 say 'Do you want to continue ?(Y/N): ' get ANS picture '!'
    read
    if ANS <> 'Y'
      clear
      close database
      return
    else
      @ 23,0 clear
      loop
    endif
  endif
  exit
enddo

** setting the printer on

@ 22,0 clear
ANS = ' '
@ 23,0 say 'Set printer on ... and press any key' get ANS
read
set print on

SNO = 1
DATA = 0

** seeking an O.I. no. and calculating total planned and actual weeks

```

```

do while .T.
  select A
  seek B->OI_NO
  PLRCPT = PLAN_WK
  ACTRCPT = ACT_CMP_WK

do while .T.
  TOTPLWK = 0
  TOTACTWK = 0
  skip
  if OI_NO <> B->OI_NO .or. eof()
    skip - 1
    PLDLRY = PLAN_WK
    if val(ACT_CMP_WK) = 0
      exit
    else
      ACTDLRY = ACT_CMP_WK

      ** do subprogram to calculate total weeks taken to complete
      ** an order

      do DVTNWK with PLRCPT, ACTRCPT, PLDLRY, ACTDLRY, TOTPLWK,;
      TOTACTWK
      DATA = 1
      exit
    endif
  endif
enddo

if DATA = 1
  if HEADING = 1

    ** printing the headings

    ? space(40)+'PERFORMANCE OF DELIVERY TIMES SCHEDULE for '+XDEPT+
    ' DEPARTMENT'
    ?
    ? space(59)+'O.I.'+' '+'Act1'+' '+'Devia-'
    ? space(44)+'Product'+space(8)+'Comp'+' '+'Comp'+' '+'tion'
    ? 'Sno'+' '+'O.I. no.'+space(3)+"Customer's name"+space(12)+
    'name'+space(7)+'Qty'
    ?? ' '+'wks'+' '+'wks'+' '+'(+/-)'+space(15)+'Remarks'
    LINE = '-----'
    WIDTH = 125
    do LINEPRT with LINE, WIDTH
    ? LINE
    HEADING = 0
  endif
  select B

  ** printing the fields if variable DATA = 1

  ?
  ? str(SNO, 2)+' '+'OI_NO+' '+'B->CNAME+' '+'PNAME+' '+'str(QTY, 2)
  ?? ' '+'str(TOTPLWK, 2)+' '+'str(TOTACTWK, 2)+' '+'D_WEEK+'
  ?? D_REASON
endif
SNO = SNO + 1
select B
skip
if DEPT <> XDEPT .or. eof()

```

```
        ? LINE
        ?
        exit
    endif
enddo

if DATA = 0
    @ 22,0 clear
    @ 23,0 say 'No completed orders found for this department'
    @ 24,0 say 'Press any key to continue' get KEY
    read
    close database
    return
endif
```

```
clear
@ 24,0 say ' Printing is over, press any key to return '
set print off
wait '' to KEY
close database
return
```

```

** PRIDELAY.PRG
** Program to print the delays occurred in various activities and
** the departments responsible for them.

clear
set talk off

** initialising the variables

OK = 0
SND = 0
CNT = 0
ANS = ' '
KEY = ' '
LESS = ' '
XDEPT = ' '
ACTWK = ' '
XDELAY = 0
HEADING= 1
DATA = 0

** opening weeks, order and activity information databases

select A
  use weeks index weeks
select B
  use ordinfo index ordinfo
select C
  use actdat index actdat

do while .T.
  @ 23,0 say 'Enter department name :' get XDEPT picture '!!!'
  read

  select B
  locate for DEPT = XDEPT
  if eof()
    @ 23,0 clear
    @ 23,0 say 'No weeks information found for this department'
    ANS = 'N'
    @ 24,0 say 'Do you want to continue ?(Y/N):' get ANS picture '!'
    read
    if ANS <> 'Y'
      clear
      close database
      return
    else
      @ 23,0 clear
      loop
    endif
  endif
  exit
enddo

** setting the printer on

@ 22,0 clear
ANS = ' '
@ 23,0 say 'Set printer on ... and press any key' get ANS
read
set print on

```

```

SNO = 0
DATA = 0

** seek O.I. no. and calculate delay for each activity

do while .T.
  select A
  seek B->OI_NO
  NEWNO = 1

  do while .T.

    XDELAY = 0
    VLREVD = val(REV_PL_WK)
    ACTLWK = ACT_CMP_WK
    if VLREVD = 0
      PL_WK = PLAN_WK
      do DELAYWK with PL_WK, ACTLWK, XDELAY
    else
      REVDWK = REV_PL_WK
      do DELAYWK with REVDWK, ACTLWK, XDELAY
    endif

    if XDELAY > 0
      if HEADING = 1

        ** printing the headings

        ? space(42)+'ACTIVITIES DELAYED REPORT for '+XDEPT+' DEPARTMENT'+
          space(10)+'DATE: '
        ?? MTODAY
        ?
        ? space(42)+'Acty'+space(42)+"Dept's Resp."+' '+Plan'+ ' '+
          'Revd'+ ' '+Actl'+ ' '+Delay'
        ? 'Sno'+space(3)+'O.I. no.'+space(3)+"Customer's name"+
          space(11)+'no.'
        ?? space(7)+'Activity name '+space(21)+'1st 2nd 3rd'+ ' '+
          'wk'+ ' '+wk'+ ' '+wk'+ ' '+wks'
        LINE = '-----'
        WIDTH = 125
        do LINEPRT with LINE, WIDTH
        ? LINE
        CNT = CNT + 5
        HEADING = 0
      endif
    endif
    if NEWNO = 1
      SNO = SNO + 1
      ?
      ? str(SNO, 2)+' '+OI_NO+' '+B->CNAME
      CNT = CNT + 2
      NEWNO = 0
    endif

    ** printing the data if DATA = 1

    DATA = 1
    select C
    seek A->ACTY_NO
    select A
    ?? ACTY_NO+' '+C->ACT_NAME+C->DEPT_RESP1+' '+C->DEPT_RESP2+' '+

```

```

        C->DEPT_RESP3
        ?? PLAN_WK+ ' '+REV_PL_WK+ ' '+ACT_CMP_WK+ ' '+str(XDELAY, 2)
    endif
    skip
    if (DATA = 1 .and. XDELAY > 0)
        ? space(42)
        CNT = CNT + 1
    endif
    if OI_NO <> B->OI_NO .or. eof()
        exit
    endif
enddo

select B
skip
if CNT > 50
    CNT = 0
    HEADING = 1
    ? LINE
    ?
    eject
endif

if DEPT <> XDEPT .or. eof()
    ? LINE
    ?
    exit
endif
enddo

if DATA = 0
    @ 22,0 clear
    @ 23,0 say 'No delayed activities found for this department'
    @ 24,0 say 'Press any key to continue' get KEY
    read
    close database
    return
endif

clear
@ 24,0 say 'Printing is over, press any key to return '
set print off
wait '' to KEY
close database
return

```

```

** PRICMPWK.PRG
** To output the completion weeks for activities completed

clear
set talk off

** initialising the variables

SNO = 0
CNT = 0
ANS = ' '
KEY = ' '
LESS = ' '
XDEPT = ' '
WKNO = ' '
ACTWK = ' '
HEADING= 1

** opening weeks, order and activity information databases

select A
  use weeks index weeks
select B
  use ordinfo index ordinfo
select C
  use actdat index actdat

do while .T.
  @ 22,0 say 'Enter department name : ' get XDEPT picture '!!'
  @ 23,0 say 'Enter week no. for which you want activity completion;
             report : ' get WKNO picture '999'
  read
  if WKNO = ' ' .or. val(substr(WKNO, 2, 2)) > 53
    ANS = 'N'
    @ 24,0 say 'Error in week no. Do you want to continue ?(Y/N): '
    get ANS picture '!'
    read
    if ANS <> 'Y'
      clear
      close database
      return
    else
      @ 24,0 clear
      loop
    endif
  endif

  select B
  locate for DEPT = XDEPT
  if eof()
    @ 24,0 clear
    @ 24,0 say 'No order information found for this department,;
              enter again'
    loop
  endif
  exit
enddo

** setting the printer on

@ 22,0 clear

```

ANS = ' '

@ 23,0 say 'Set printer on ... and press any key' get ANS

read

set print on

SNO = 0

WKLESS = 0

do while .T.

select A

seek B->OI_NO

NEWNO = 1

do while .T.

LESS = ' '

ACTWK = ACT_CMP_WK

do DIFFWK with ACTWK, WKNO, LESS

if LESS = 'Y'

if HEADING = 1

** printing the headings

? space(35)+'ACTIVITIES COMPLETED REPORT UPTO WEEK No: '+WKNO+
' for '+XDEPT+' DEPARTMENT'

?

? space(42)+'Acty'+space(43)+"Dept's Resp."+' '+Plan'+
' '+Revd'+ ' '+Act1'

? 'Sno'+space(3)+'O.I. no.'+space(3)+"Customer's name"+
space(11)+'no.'

?? space(7)+'Activity name '+space(22)+'1st 2nd 3rd'+
space(3)+'wk'+ ' '+wk'+ ' '+wk'

LINE = '-----'

WIDTH = 119

do LINEPRT with LINE, WIDTH

? LINE

CNT = CNT + 5

HEADING = 0

endif

if NEWNO = 1

SNO = SNO + 1

?

? str(SNO, 2)+' '+OI_NO+' '+B->CNAME

CNT = CNT + 2

NEWNO = 0

endif

WKLESS = 1

select C

seek A->ACTY_NO

select A

?? ACTY_NO+' '+C->ACT_NAME+C->DEPT_RESP1+' '+C->DEPT_RESP2+' '+
C->DEPT_RESP3

?? ' '+PLAN_WK+' '+REV_PL_WK+' '+ACT_CMP_WK

endif

skip

if (WKLESS = 1 .and. LESS = 'Y')

? space(42)

CNT = CNT + 1

endif

if OI_NO <> B->OI_NO .or. eof()

```

        exit
    endif
enddo

select B
skip
if CNT > 50
    CNT = 0
    HEADING = 1
    ? LINE
    ?
    eject
endif

if DEPT <> XDEPT .or. eof()
    ? LINE
    ?
    exit
endif
enddo

if WKLESS = 0
    @ 22,0 clear
    @ 23,0 say 'No actual weeks less than given week'
    @ 24,0 say 'Press any key to return' get KEY
    read
    close database
    return
endif

clear
@ 24,0 say ' Printing is over, press any key to return '
set print off
wait '' to KEY
close database
return

```

```
** NEXTYRWK.PRG
** Program to give the next year's week no. in case it is > 53
```

```
parameters WEEK
```

```
WKS = val(substr(WEEK, 2, 2))
if WKS > 53
    NYR = val(substr(WEEK, 1, 1)) + 1
    if NYR > 9
        NYR = 0
    endif

    NWKS = WKS - 53
    if NWKS <= 9
        WEEK = str(NYR, 1) + '0' + str(NWKS, 1)
    else
        WEEK = str(NYR, 1) + str(NWKS, 2)
    endif
endif
return
```

```
** LINEPRT.PRG
** Program to print a line of a given length

parameters LINE, SIZE

FLINE = LINE

do while .T.

    FLINE = FLINE + LINE

    if len(FLINE) >= SIZE
        LINE = FLINE
        exit
    endif
enddo
return
```

```

** VALIDORD.PRG
** Program to validate the data entered in fields of order information file

parameters PRODUCT, ORDTYPE, DEPRMT, VALID

VALID = ' '

* check for existence of product name in products file

select D
go top
locate for trim(FNAME) = trim(PRODUCT)
if eof()
    @ 21,0 say 'Product name not found in products file, enter new name'
    VALID = 'L'
endif

* validate that order type is external or IDTO only

if .not. (ORDTYPE = 'EXT' .or. ORDTYPE = 'IDTO')
    @ 22,0 say 'Error in order type, give only EXT or IDTO'
    VALID = 'L'
endif

* validate that department is of proper type

if .not. (DEPRMT='YT' .or. DEPRMT='YA' .or. DEPRMT='RK' .or. DEPRMT='RE')
    @ 23,0 say 'Error in department type, give YT, YA, RK or RE only'
    VALID = 'L'
endif

return

```

```
***** PRGDATE *****  
* Extracts ASEA's standard date format from system date  
*****
```

```
Y= ' '  
M= ' '  
D= ' '  
TEMP= ' '  
MTODAY= ' '  
TEMP=dtoc(Date())  
store substr(TEMP,7) to Y  
store substr(TEMP,1,2) to M  
store substr(TEMP,4,2) to D  
MTODAY=Y+M+D
```

```

** DVTNWK.PRG
** Program to calculate the deviations of actuals from planned weeks
   parameters PLRCT, ACTRCT, PLDRY, ACDRY, TOTPL, TOTACT
** initialising the variables

   TOTPL  = 0
   TOTACT = 0
   LOSS   = 0.00

** separating years and weeks

   VLPLRCT = val(substr(PLRCT, 1, 1))
   VLPLDRY = val(substr(PLDRY, 1, 1))

   VLACRCT = val(substr(ACTRCT, 1, 1))
   VLACDRY = val(substr(ACDRY, 1, 1))

   if VLPLRCT = VLPLDRY
       TOTPL = val(PLDRY) - val(PLRCT)
   else
       WKPLDRY = val(substr(PLDRY, 2, 2))
       WKPLRCT = val(substr(PLRCT, 2, 2))
       TOTPL   = WKPLDRY + (53 - WKPLRCT)
   endif
   if VLACRCT = VLACDRY
       TOTACT = val(ACDRY) - val(ACTRCT)
   else
       WKACDRY = val(substr(ACDRY, 2, 2))
       WKACRCT = val(substr(ACTRCT, 2, 2))
       TOTACT  = WKACDRY + (53 - WKACRCT)
   endif

   if TOTACT >= TOTPL
       select B
       XDVTN = TOTACT - TOTPL
       if XDVTN < 10
           POSDVTN = '+' + '0' + str(XDVTN, 1)
       else
           POSDVTN = '+' + str(XDVTN, 2)
       endif
       YRLOSS = VALUE * (18.0/100.00)
       WKLOSS = YRLOSS *(XDVTN/53)
       replace D_WEEK with POSDVTN
       replace D_LOSS with str(WKLOSS, 4)
   else
       XDVTN = TOTPL - TOTACT
       if XDVTN < 10
           NEGDVTN = '-' + '0' + str(XDVTN, 2)
       else
           NEGDVTN = '-' + str(XDVTN, 2)
       endif
       replace D_WEEK with NEGDVTN
   endif
   return

```

```

** DIFFWK.PRG
** Program to calculate the difference between the actual completion week
** and the week input

parameters STDWK, WEEKNO, LESS

** extracting the year and weeks from given and actual weeks

YRSTD = val(substr(STDWK, 1, 1))
YRWKNO = val(substr(WEEKNO, 1, 1))
WKSTD = val(substr(STDWK, 2, 2))
WKWEEK = val(substr(WEEKNO, 2, 2))

** put less = Y ie yes, for actual week less than given week

if (val(STDWK) <= val(WEEKNO) .and. val(STDWK) <> 0)
    LESS = 'Y'
else
    if YRSTD = 9 .and. YRWKNO = 0
        if WKSTD <= (WKWEEK + 53)
            LESS = 'Y'
        endif
    endif
endif

** put less = N ie no, for planned or reivsied week more than given week

if (val(STDWK) > val(WEEKNO) .and. val(STDWK) <> 0)
    LESS = 'N'
else
    if YRSTD = 0 .and. YRWKNO = 9
        if (WKSTD + 53) >= WKWEEK
            LESS = 'N'
        endif
    endif
endif

return

```

```

** DELAYWK.PRG
** Program to calculate the delay in an activity

parameters STDWK, ACTLWK, DELAY

VLSTDWK = val(substr(STDWK, 1, 1))
VLACTLWK = val(substr(ACTLWK, 1, 1))
WKSTD = val(substr(STDWK, 2, 2))
WKACTL = val(substr(ACTLWK, 2, 2))

if VLSTDWK = VLACTLWK
  if WKACTL > WKSTD
    DELAY = WKACTL - WKSTD
  endif
else
  if VLACTLWK > VLSTDWK .or. (VLACTLWK = 0 .and. VLSTDWK = 9)
    DELAY = WKACTL + (53 - WKSTD)
  endif
endif

return

```

```
** CHKWEEK.PRG
** Program to check for week no. to be <= 53
```

```
parameters WEEK, STATUS
```

```
STATUS = ' '
if val(substr(WEEK, 2, 2)) > 53
    STATUS = 'E'
endif
return
```

```
** CHKMONT.H.PR8
** Program  for checking the correctness of given year and month

parameter  DATE, OK
OK = 1

** checking the year **

if substr( DATE, 1, 2 ) < '88'
    OK = 0
endif

** checking the month **

if substr( DATE, 3, 2 ) < '01' .OR. substr( DATE, 3, 2 ) > '12'
    OK = 0
endif
return
```

TIME RECORDING SYSTEM

```

** TIMEMENU.PRG
** Main menu for Time Recording System

clear
set talk off

** initialising variables

CHOICE = ' '
MTODAY = space(6)

** checking if today's date has been set

do PRGDATE
if MTODAY <= '880301'
    @ 23,0
    wait "Today's date has not been set on booting the system.;
        Please reboot ! Press <CR>..." to PRESS
    quit
endif

do while .T.

** Displaying the list of items in the Main Menu

clear
set color to /w
@ 4,36 say 'TRS'
@ 5,33 say 'MAIN MENU'
set color to w
@ 7,25 say '1. Add/Edit Personal Information'
@ 9,25 say '2. Input Daily In/Out Times'
@ 11,25 say '3. Edit Times Entered'
@ 13,25 say '4. Month End Time Calculation'
@ 15,25 say '5. Print Menu'
@ 17,25 say '6. Checklist Menu'
@ 19,25 say 'Q. Exit'
CHOICE = ' '
@ 21,25 say 'Enter your choice :' get CHOICE picture '!'
read

** Case Statement

do case
    case CHOICE = '1'
        do PERSEDIT

    case CHOICE = '2'
        do TIMEIP

    case CHOICE = '3'
        do EDITIME

    case CHOICE = '4'
        do TIMECALC

    case CHOICE = '5'
        do PRNTMENU

    case CHOICE = '6'
        do CKLSMENU

```

```
case CHOICE = 'Q'  
  clear all  
  return
```

```
endcase
```

```
enddo
```

```

** PRNTMENU.PRG
** Menu to choose an option available for printing reports

clear
set talk off

** initialising variables

CHOICE = 0
XDIV   = ' '
XDAYS  = 0
WDAYS  = 0
XMONTH = space(4)

do while .T.

** opening time and temporary department information databases

select A
  use TIMEREG index TIMEREG

select B
  use TEMPDEPT

** Displaying the list of items in the Print Menu

clear
set color to w+
@ 4,34 say 'TRS'
@ 5,30 say 'Print Menu'
set color to w
@ 7,25 say '1. Employee times summary report'
@ 9,25 say '2. Department summary report'
@ 12,25 say '0. Exit to previous menu'
CHOICE = 0
@ 21,25 say 'Enter your choice : ' get CHOICE picture '9'
read

clear
@ 23,0 say 'Enter division for which you want the report';
  get XDIV picture '!'
@ 24,0 say 'Enter month for which report is required ' ;
  get XMONTH picture '9999'
read

** Case Statement

do case
  case CHOICE = 1
    select A
    @ 23,0 clear
    @ 23,0 say 'Set printer on .... and press a key'
    wait '' to KEY
    set print on
    ? space(20)+'EMPLOYEE TIMES SUMMARY REPORT'
    ?
    ? space(8)+'Division: '+XDIV+space(26)+'Month: '+XMONTH
    ?
    report form PRIEMPTM for PERIOD = XMONTH plain noeject to print
    set print off

```

```

case CHOICE = 2
  select B
  @ 23,0 clear
  @ 23,0 say 'Enter number of days in this month: ' get X DAYS picture
  @ 24,0 say 'Enter number of working days: ' get W DAYS picture
  read
  @ 23,0 clear
  @ 23,0 say 'Set print on .... and press a key'
  wait '' to key
  set print on
  ? space(14)+'DEPARTMENTS SUMMARY REPORT'
  ?
  ? 'Division: '+XDIV+space(15)+'No. of days in this month: '+
    str(XDAYS, 2)
  ? 'Month : '+XMONTH+space(12)+'No. of working days : '+
    str(WDAYS, 2)
  ?
  report form PRIDEPTM for DIV = XDIV plain noeject to print
  set print off

case CHOICE = 0
  clear
  close database
  return

endcase

enddo

```

```

** CKLSMENU.PRG
** Check listing menu to print dump of data entered in pers and
** time register

clear
set talk off
** Initialising the variables
CHOICE = 0

do while .T.

** Opening personnel and time information databases

select A
  use PERS index PERS
select B
  use TIMEREG index TIMEREG

** Displaying the options available for check listing

clear
set color to w+
@ 4,36 say 'TRS'
@ 5,30 say 'Checklist Menu'
set color to w
@ 7,20 say '1. Personnel Information check list'
@ 9,20 say '2. Time register check list 1'
@ 11,20 say '3. Time register check list 2'
@ 13,20 say '0. Exit to previous Menu'
CHOICE = 0
@ 16,25 say 'Enter your Choice :' get CHOICE picture '9'
read

do case

  case CHOICE = 1
    select A
    report form PERSLIST for SHIFT = 'A' .or. SHIFT = 'B'
    plain noeject to print

  case CHOICE = 2
    select B
    report form TIMELST1 plain noeject to print

  case CHOICE = 3
    select B
    report form TIMELST2 plain noeject to print

  case CHOICE = 0
    exit

endcas

enddo
clear
close database
return

```

```

** PERSEDIT.PRG
** Screen for adding/editing the Personnel Information in PERS File

clear
set talk off

** Initialising the variables

XEMP_NO = ' '
XENAME = space(30)
XDIV = ' '
COMMAND = ' '
STATUS = ' '
XSHIFT = ' '
XDEPT = ' '
XCATEGORY = ' '

** Open PERS database

use PERS index PERS

do while .T.

** Printing the headings

@ 3,16 say 'Personnel Information - Adding/Editing Screen'
set color to /w
@ 5,5 say 'N-Next      P-Previous      E-Edit      A-Add      S-Seek      X-Exit'
set color to w
@ 8,5 say 'Emp. no: '+EMP_NO
@ 8,26 say 'Name____: '+ENAME
@ 10,5 say 'Category: '+CATEGORY
@ 10,26 say 'Shift____: '+SHIFT
@ 12,5 say 'Division: '+DIV
@ 12,26 say 'Dept____: '+DEPT

do while .T.

** Using the edit options

COMMAND = ' '
@ 1,0 say 'Give Command:' get COMMAND picture '!'
read
@20,0 clear

do case

case COMMAND = 'N'
if .not. eof()
skip
endif
if eof()
@ 23,0 say 'End of file, no more records after this !'
else
@ 8,15 say EMP_NO
@ 8,36 say ENAME
@ 10,15 say CATEGORY
@ 10,36 say SHIFT
@ 12,15 say DIV
@ 12,36 say DEPT
endif
endif

```

```

case COMMAND = 'P'
  if .not. bof()
    skip -1
  endif
  if bof()
    @ 23,0 say 'You are already at the beginning of file !'
  else
    @ 8,15 say EMP_NO
    @ 8,36 say ENAME
    @ 10,15 say CATEGORY
    @ 10,36 say SHIFT
    @ 12,15 say DIV
    @ 12,36 say DEPT
  endif

case COMMAND = 'S'
  do while .T.
    @ 1,20 say 'Give Emp. no:' get XEMP_NO picture '!999'
    read
    seek XEMP_NO
    if eof()
      @ 23,0 say '* Given Emp. no. not found *'
      ANS = 'N'
      @ 24,0 say 'Do you want to give new Emp. no. ? (Y/N):';
      get ANS picture '!'
      read
      if ANS <> 'Y'
        clear
        close database
        return
      else
        @ 23,0 clear
        loop
      endif
    endif
    @ 8,15 say EMP_NO
    @ 8,36 say ENAME
    @ 10,15 say CATEGORY
    @ 10,36 say SHIFT
    @ 12,15 say DIV
    @ 12,36 say DEPT
    exit
  enddo

case COMMAND = 'E'
  XEMP_NO = EMP_NO
  XENAME = ENAME
  XCATEGORY= CATEGORY
  XSHIFT = SHIFT
  XDIV = DIV
  XDEPT = DEPT
  OLDEMP = EMP_NO
  do while .T.
    @ 8,15 get XEMP_NO picture '!999'
    @ 8,36 get XENAME picture '!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!'
    @ 10,15 get XCATEGORY picture '!!!'
    @ 10,36 get XSHIFT picture '!'
    @ 12,15 get XDIV picture '!'
    @ 12,36 get XDEPT picture '!!!'
    read
  enddo

```

```

    if XEMP_NO <> OLDEMP
        locate for EMP_NO = XEMP_NO
        if .not. eof()
            @ 23,0 clear
            @ 23,0 say 'Given employee no. already present, enter new no.'
            loop
        endif
    endif
    @ 20,0 clear
    do VLDTMPERS with XEMP_NO, XCATEGORY, XDIV, XDEPT, XSHIFT
        if STATUS = 'L'
            loop
        else
            exit
        endif
    enddo

    replace EMP_NO with XEMP_NO, ENAME with XENAME
    replace CATEGORY with XCATEGORY, SHIFT with XSHIFT
    replace DIV with XDIV, DEPT with XDEPT
    @ 23,0 say 'Record updated !'

case COMMAND = 'A'
    XEMP_NO = ' '
    XENAME = space(30)
    XDIV = ' '
    XSHIFT = ' '
    XDEPT = ' '
    XCATEGORY = ' '

    @ 8,15 get XEMP_NO picture '!999'
    @ 8,36 get XENAME picture '!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!'
    @ 10,15 get XCATEGORY picture '!!'
    @ 10,36 get XSHIFT picture '!'
    @ 12,15 get XDIV picture '!'
    @ 12,36 get XDEPT picture '!!!!'
    read
    locate for EMP_NO = XEMP_NO
    if .not. eof()
        @ 23,0 clear
        @ 23,0 say 'Given employee no. already present, enter new no.'
        loop
    endif

    @ 20,0 clear
    do VLDTMPERS with XEMP_NO, XCATEGORY, XDIV, XDEPT, XSHIFT
        if STATUS = 'L'
            loop
        endif
    enddo
    append blank
    replace EMP_NO with XEMP_NO, ENAME with XENAME
    replace CATEGORY with XCATEGORY, SHIFT with XSHIFT
    replace DIV with XDIV, DEPT with XDEPT
    @ 23,0 say 'Record Added !'

case COMMAND = 'X'
    exit
endcase
enddo

ANS = 'N'

```

```
@ 24,0 say 'Do you want to add or edit information of another;  
employee ?(Y/N):'get ANS picture '!''  
read  
if ANS <> 'Y'  
    clear  
    close database  
    return  
endif  
@ 6,0 clear  
enddo
```

```

** TIMEIP.PRG
** Displays screen for getting the in and out times for an employee **

clear
set talk off

** initialising the variables

MONTH   = space(4)
XEMP_NO = ' '
XTMIN   = 0.00
XTMOUT  = 0.00
OK = 0
DAYS    = ' '
STATUS  = ' '
XDATE   = space(6)

do while .T.

** painting the screen and get month, employee no

set color to w+
@ 1,25 say "Time Record - Add Screen"
set color to w
@ 2,13 say "Period:"
set color to w
@ 2,21 get MONTH picture '9999'
read
do CHKMONT with MONTH, OK
if OK = 0
    set color to w*
    @ 23,0 say 'Error in given period, correct the field'
    set color to w
    loop
endif
exit
enddo
@ 23,0 clear

** opening personal and time information databases

select A
use PERS index PERS

select B
use TIMREG index TIMREG

do while .T.

XEMP_NO = ' '
set color to w+
@ 2,49 say "Emp. no.:"
set color to w
@ 2,59 get XEMP_NO picture '!999'
read

select A
seek XEMP_NO

if .not. eof()
    set color to w+

```

```

@ 3,13 say 'Name : '+ENAME
@ 3,49 say 'Shift : '+SHIFT
set color to w
else
@23,0 say '** Given Emp. no. not found **'
ANS = ''
@24,0 say 'Do you want to give new Emp. no.?(Y/N):' get ANS picture '!'
read
if ANS <> 'Y'
clear
close database
return
else
@23,0 clear
loop
endif
endif

select B
go top
locate for EMP_NO = XEMP_NO .and. PERIOD = MONTH

if .not. eof()
@ 23,0 say 'Given Emp. no. already present, enter new no.'
loop
endif
@ 23,0 clear

append blank
replace EMP_NO with XEMP_NO, PERIOD with MONTH
XDATE = MONTH + '01'
M = substr ( XDATE, 3, 2 )
Y = substr ( XDATE, 1, 2 )
do GETDAYS with Y, M , DAYS

ROW=6
COL=0
@ 4,COL+10 say "Time"
@ 5,COL+1 say "Date In Out"

do while .T.

@ ROW,COL+0 say XDATE
@ ROW,COL+7 get XTMIN picture '99.99'
@ ROW,COL+13 get XTMOUT picture '99.99'
read
do TIMECHK with XTMIN, XTMOUT, STATUS
if STATUS = 'E'
exit
endif
if STATUS = 'L'
loop
endif

D = substr (XDATE, 5, 2)
IN = 'IN' + D
OUT= 'OUT' + D
replace &IN with XTMIN, &OUT with XTMOUT

TODATE = MONTH + DAYS
if XDATE = TODATE

```

```

        exit
    endif

    ROW = ROW + 2
    if ROW > 21
        ROW = 6
        COL = COL + 20
        @ 4,COL+10 say "Time"
        @ 5,COL+1 say "Date   In       Out"
    endif

    do TMNEXTDT with XDATE, DAYS

enddo

@ 23,0 clear
ANS = ' '
@ 24,0 say 'Do you want to enter daily times for another person ?(Y/N):';
        get ANS picture '!'
read
if ANS<>'Y'
    clear
    close database
    return
endif

@ 3,0 clear
enddo

```

** TIMECHK.PRG

** Program for validating the time entered in time register

parameters XTIMEIN, XTIMEOUT, STATUS
STATUS = ' '

INSTR = ' '
INMIN = ' '
OUTSTR = ' '
OUTMIN = ' '

INSTR = str(XTIMEIN, 5, 2)
INMIN = substr(INSTR, 4, 2)
OUTSTR = str(XTIMEOUT, 5, 2)
OUTMIN = substr(OUTSTR, 4, 2)

if .not. (XTIMEIN = 0.00 .and. XTIMEOUT = 0.00)
 if XTIMEIN > 24.0 .or. XTIMEOUT > 24.0 .or. XTIMEIN >= XTIMEOUT .or.
 INSTR >= 60 .or. OUTSTR >= 60
 ANS = ' '
 @ 23,0 say 'Error in time. Do you want to continue ? (Y/N):';
 get ANS picture '!'
 read
 if ANS <> 'Y'
 STATUS = 'E'
 else
 @23,0 clear
 STATUS = 'L'
 endif
 endif
endif
return

```

** EDITIME.PRG
** Screen for editing the Daily Times entered in time register

clear
set talk off

** Initialising the variables

MONTH = space(4)
XDATE = space(6)
XTMIN = 0.00
XTMOUT = 0.00
DAYS = ' '
STATUS = ' '

select A
use TIMEREG index TIMEREG

set color to /w
@ 1,25 say 'Time Record Editing Screen'
@ 2,0 say 'N - Next      E - Edit      X - Exit'
set color to w

do while .T.

    XEMP_NO = ' '
    COMMAND = ' '

** Creates editing screen

    @ 3,16 say 'Emp. no:'
    @ 3,25 get XEMP_NO picture '!999'

    @ 3,39 say 'Period:'
    @ 3,47 get MONTH picture '9999'
    read

    select A
    go top
    locate for EMP_NO = XEMP_NO .and. PERIOD = MONTH

    if eof()
        @ 23,0 say '* Given Emp. no. for the said period not found *'
        ANS = 'N'
        @ 24,0 say 'Do you want to give new Emp. no ?(Y/N):' get ANS picture '!'
        read
        if ANS <> 'Y'
            clear
            close database
            return
        else
            @ 23,0 clear
            loop
        endif
    endif

    XDATE = MONTH + '01'
    M = substr(XDATE, 3, 2)
    Y = substr(XDATE, 1, 2)
    do GETDAYS with Y, M, DAYS

```

```

ROW = 6
COL = 0
@ 4, COL+10 say 'Time'
@ 5, COL+1 say 'Date In Out'

D = substr(XDATE, 5, 2)
IN = 'IN' + D
OUT = 'OUT' + D
@ ROW, COL say XDATE
@ ROW, COL+7 say &IN
@ ROW, COL+13 say &OUT

do while .T.
  COMMAND= ' '
  do while COMMAND<>'N' .and. COMMAND<>'E' .and. COMMAND<>'X'
    @ 1,0 say 'Give Command:' get COMMAND picture '!'
    read
  enddo

  @ 22,0 clear
  do case
    case COMMAND = 'N'

      do TMNEXTDT with XDATE, DAYS
        D = substr(XDATE, 5, 2)
        IN = 'IN' + D
        OUT = 'OUT' + D
        ROW = ROW + 2
        if ROW > 22
          ROW = 6
          COL = COL + 20
          @ 4, COL+10 say 'Time'
          @ 5, COL+1 say 'Date In Out'
        else
          if ROW > 21
            ROW = 7
            COL = COL+20
            @ 4, COL+10 say 'Time'
            @ 5, COL+1 say 'Date In Out'
          endif
        endif
      endif
      TODATE = MONTH + DAYS
      if XDATE > TODATE
        exit
      endif
      @ ROW, COL+0 say XDATE
      @ ROW, COL+7 say &IN
      @ ROW, COL+13 say &OUT

    case COMMAND = 'E'
      do while .T.
        XTMIN = &IN
        XTMOU = &OUT
        @ ROW, COL+0 say XDATE
        @ ROW, COL+7 get XTMIN picture '99.99'
        @ ROW, COL+13 get XTMOU picture '99.99'
        read

        do TIMECHK with XTMIN, XTMOU, STATUS
          if STATUS = 'L'
            loop
          endif
        enddo
      enddo
    endcase
  enddo
enddo

```

```

        else
            exit
        endif
    enddo

    D = substr(XDATE, 5, 2)
    IN  = 'IN'  + D
    OUT = 'OUT' + D
    replace &IN with XTMIN, &OUT with XTMOU
    @ 22,0 say 'Record updated !'

    case COMMAND = 'X'
        exit
    endcase

enddo

ANS = 'N'
@ 24,0 say 'Do you want to edit daily time of another employee ?(Y/N):';
        get ANS picture '!'
read
if ANS <> 'Y'
    clear
    close database
    return
endif
@ 6,0 clear
enddo

```

```

** TIMECALC.PRG
** Program to do month end early / late time record calculations

clear
set talk off

** Initialising the variables

MONTH = space(4)
ANS   = ' '
XDIV  = ' '

** Opening the databases

select A
use TIMEREG index TIMEREG

select B
use PERS index PERS

select C
use TEMPDEPT index TEMPDEPT
zap

** Getting the period for which calculations are to be done

do while .T.

    MONTH = space(4)
    @ 21,10 say 'Enter the period for which you want to do time calculations'
    @ 22,10 say 'Period:' get MONTH picture '9999'
    read

    if MONTH = ' '
        ANS = 'N'
        @ 23,0 say 'Error in period. Do you want to continue?(Y/N):' get ANS picture '9999'
        read
        if ANS <> 'Y'
            clear
            close database
            return
        else
            @ 23,0 clear
            loop
        endif
    endif
    exit
enddo

select A
go top
seek MONTH
if eof()
    @ 23,0 say 'No Time Details found for the given period'
    close database
    return
endif

** Calculate the extra/short time for the employees

DAYS = ' '

```

```
XTMIN = 0.00
XTMOUT = 0.00
IN = 0.00
OUT = 0.00
DIFF= 0.00
```

```
@23,0 clear
@ 23,0 say 'Wait, .... doing time calculations'
NOE=1
```

```
** Total extra / short hours calculation
```

```
do while .T.
```

```
  XDATE = MONTH + '01'
  M = substr(XDATE, 3, 2)
  Y = substr(XDATE, 1, 2)
  do GETDAYS with Y, M, DAYS
  @23,40 say 'No of emp :'+str(NOE,3)
```

```
  select B
  seek A->EMP_NO
```

```
  TOTXTRA = 0.00
  TOTSHRT = 0.00
  LATEHRS = 0.00
  LATECNT = 0
  CNT = 0
```

```
** calculate the daily extra/early/late times for A or B shift
```

```
if SHIFT = 'A'
```

```
  do while .T.
```

```
    XTRAHR = 0.00
    SHRTHR = 0.00
    LATEIN = 0.00
```

```
    select A
    DT = substr(XDATE, 5, 2)
    XIN = 'IN' + DT
    XOUT = 'OUT' + DT
    IN = &XIN
    OUT = &XOUT
```

```
    if (IN>0.00 .and. OUT>0.00)
      do TIMEADIFF with IN, OUT, XTRAHR, SHRTHR, LATEIN
      if LATEIN > 0.00
        CNT = CNT + 1
      endif
    endif
    LATEHRS = LATEHRS + LATEIN
    if CNT > 3 .or. LATEHRS > 1.00
      LATECNT = LATECNT + 1
    endif
```

```
    TOTXTRA = TOTXTRA + XTRAHR
    TOTSHRT = TOTSHRT + SHRTHR
```

```
    if (DT > '09' .and. DT < '28')
      NEWD = str(val(DT)+1, 2)
```

```

        XDATE = substr(XDATE, 1, 4)+NEWD
    else
        do TMNEXTDT with XDATE, DAYS
    endif
    if substr(XDATE, 5, 2) > DAYS
        exit
    endif
enddo
endif

select B
if SHIFT = 'B'

    do while .T.
        XTRAHR = 0.00
        SHRTHR = 0.00
        LATEIN = 0.00

        select A
        DT = substr(XDATE, 5, 2)
        XIN = 'IN' + DT
        XOUT = 'OUT' + DT
        IN = &XIN
        OUT = &XOUT

        if (IN>0.00 .and. OUT>0.00)
            do TIMEBDIFF with IN, OUT, XTRAHR, SHRTHR, LATEIN
            if LATEIN > 0.00
                CNT = CNT + 1
            endif
        endif
        LATEHRS = LATEHRS + LATEIN
        if CNT > 3 .or. LATEHRS > 1.00
            LATECNT = LATECNT + 1
        endif

        TOTXTRA = TOTXTRA + XTRAHR
        TOTSHRT = TOTSHRT + SHRTHR

        if (DT > '09' .and. DT < '28')
            NEWD = str(val(DT)+1, 2)
            XDATE = substr(XDATE, 1, 4)+NEWD
        else
            do TMNEXTDT with XDATE, DAYS
        endif
        if substr(XDATE, 5, 2) > DAYS
            exit
        endif
    enddo
endif

```

** calculate nett and store in string form

```

select A
if TOTXTRA >= TOTSHRT
    NETTHR = TOTXTRA - TOTSHRT
    NETTSTR = substr(str(NETTHR,6,2),5,2)
    NETTMIN = val(NETTSTR) * 60
    NETTIME = '+' + substr(str(NETTHR,6,2),1,3) + ':' + substr(str(NETTMIN,4),1,2)
    replace NETT with NETTIME
else
    NETTHR = TOTSHRT - TOTXTRA

```

```

NETTSTR = substr(str(NETTTHR,6,2),5,2)
NETTMIN = val(NETTSTR) * 60
NETTIME = '-' + substr(str(NETTTHR,6,2),1,3) + ':' + substr(str(NETTMIN,4),1,2)
replace NETT with NETTIME
endif

```

** convert time into string form and store in time register

```

TOTXTST = substr(str(TOTXTRA,6,2),5,2)
TOTXTMIN = val(TOTXTST) * 60
TOTXTHRS = substr(str(TOTXTRA,6,2),1,3) + ':' + substr(str(TOTXTMIN,4),1,2)
TOTSHST = substr(str(TOTSHRT,6,2),5,2)
TOTSHMIN = val(TOTSHST) * 60
TOTSHHRS = substr(str(TOTSHRT,6,2),1,3) + ':' + substr(str(TOTSHMIN,4),1,2)
TOTLTST = substr(str(LATEHRS,6,2),5,2)
TOTLTMIN = val(TOTLTST) * 60
TOTLTHRS = substr(str(LATEHRS,6,2),1,3) + ':' + substr(str(TOTLTMIN,4),1,2)

```

```

replace TOT_XTRAHR with TOTXTHRS, TOT_SHRTHR with TOTSHHRS
replace LATE_HRS with TOTLTHRS, LATE_OCCNS with LATECNT

```

** store extra/short hours in temporary file for department

```

select C
locate for DEPT = B->DEPT
if eof()
    append blank
    replace DIV with B->DIV, DEPT with B->DEPT
    replace TOT_XTRAHR with TOTXTHRS, TOT_SHRTHR with TOTSHHRS
    replace NETT with NETTIME
else
    TOTXTST = substr(TOT_XTRAHR,6,2)
    TOTXTMIN = val(TOTXTST) * 60
    TOTXTAHR = val(substr(TOT_XTRAHR,1,4)) + TOTXTMIN + TOTXTRA
    TOTHRSTR = substr(str(TOTXTAHR,7,2),6,2)
    TOTHRMIN = val(TOTHRSTR) * 60
    TOTHRXTA = substr(str(TOTXTAHR,7,2),1,4) + ':' + substr(str(TOTHRMIN,4),1,2)

    TOTSHST = substr(TOT_SHRTHR,6,2)
    TOTSHMIN = val(TOTSHST) * 60
    TOTSRTHR = val(substr(TOT_SHRTHR,1,4)) + TOTSHMIN + TOTSHRT
    TOTHRSHS = substr(str(TOTSRTHR,7,2),6,2)
    TOTHRSMN = val(TOTHRSHS) * 60
    TOTHRSHT = substr(str(TOTSRTHR,7,2),1,4) + ':' + substr(str(TOTHRSMN,4),1,2)

    replace TOT_XTRAHR with TOTHRXTA, TOT_SHRTHR with TOTHRSHT

    if TOTXTAHR >= TOTSRTHR
        NETTHR = TOTXTAHR - TOTSRTHR
        NETTSTR = substr(str(NETTHR,6,2),5,2)
        NETTMIN = (NETTSTR) * 60
        NETTIME = '+' + substr(str(NETTHR,6,2),1,3) + ':' + substr(str(NETTMIN,4),1,2)
        replace NETT with NETTIME
    else
        NETTHR = TOTSRTHR - TOTXTAHR
        NETTSTR = substr(str(NETTHR,6,2),5,2)
        NETTMIN = val(NETTSTR) * 60
        NETTIME = '-' + substr(str(NETTHR,6,2),1,3) + ':' + substr(str(NETTMIN,4),1,2)
        replace NETT with NETTIME
    endif
endif
endif

```

```
select A
skip

if PERIOD <> MONTH .or. eof()
    exit
endif
NOE=NOE+1
enddo

@24,0
@24,0 say 'Daily Time calculation complete.'
close database
return
```

```

** CHKDATE.PRG
** Program for checking the correctness of given date

parameter DATE, OK
OK = 1

** checking the year **

if substr( DATE, 1, 2 ) < '88'
    OK = 0
endif

** checking the month **

if substr( DATE, 3, 2 ) < '01' .OR. substr( DATE, 3, 2 ) > '12'
    OK = 0
endif

** checking the date **

DAYS = ' '
do GETDAYS with substr( DATE, 1, 2 ), substr( DATE, 3, 2 ), DAYS
if substr( DATE, 5, 2 ) < '01' .or. substr( DATE, 5, 2 ) > DAYS
    OK = 0
endif

```

```

* VALIDPERS.PRG
* Program to validate various parameters in persinfo file

parameters XEMP_NO, XCATEGORY, XDIV, XDEPT, XSHIFT

STATUS = ' '

* EMP_NO starts with 'Y', 'R' and greater than 300

if (substr(XEMP_NO, 1, 1) <> 'R' .and. substr(XEMP_NO, 1, 1) <> 'Y');
    .or. substr(XEMP_NO, 2, 3) < '300' .or. XEMP_NO = '
    @ 20,0 say 'Error in Emp. no. ,give >= R300 or Y300 '
    STATUS = 'L'
endif

* CATEGORY's first character is 'G', 'W', 'T', 'M' or 'N'

XCATGY = substr(XCATEGORY, 1, 1)
if (XCATGY <> 'G' .and. XCATGY <> 'M' .and. XCATGY <> 'T' .and. XCATGY <> 'W' .and.
    @ 21,0 say "Error in Category type, give first figure only as 'G','W','T','N' or
    STATUS = 'L'
endif

* DIV is 'R', 'Y' and DIV= 1st character in EMP_NO

if (XDIV <> 'R' .and. XDIV <> 'Y') .or. XDIV = ' ' .or. XDIV <> substr(XEMP_NO, 1,
    @ 22,0 say "Error in Division type, give only 'R' or 'Y' and same as in Emp. no.
    STATUS = 'L'
endif

* DEPT starts with 'R' or 'Y' only

if substr(XDEPT,1, 1) <> substr(XEMP_NO, 1, 1) .or. XDEPT = '
    @ 23,0 say "Error in Dept. type, give first letter same as in Emp. no."
    STATUS = 'L'
endif

* SHIFT is 'A' or 'B' only

if (XSHIFT <> 'A' .and. XSHIFT <> 'B') .or. XSHIFT = ' '
    @ 24,0 say "Error in Shift type, give only as 'A' or 'B'"
    STATUS = 'L'
endif

return

```

```
XCATGY <> 'N')
'M' "
```

** GETNXTDT.PRG

** Program for changing the date according to the calender

parameters XDATE, DAYS

D = substr(XDATE, 5, 2)

M = substr(XDATE, 3, 2)

Y = substr(XDATE, 1, 2)

VAL_D = val(D)+1

V_DAYS = val(DAYS)

if VAL_D > V_DAYS

VAL_D = 1

VAL_M = val(M) + 1

if VAL_M = 13

VAL_M = 1

VAL_Y = val(Y) + 1

Y = str(VAL_Y, 2)

endif

if VAL_M < 10

M = '0'+str(VAL_M,1)

else

M = str(VAL_M, 2)

endif

endif

if VAL_D < 10

D = '0'+str(VAL_D,1)

else

D = str(VAL_D, 2)

endif

XDATE = Y+M+D

return

```

** TIMEADIFF.PRG
** Program to calculate the difference in time with standard time

parameters IN, OUT, HRXTRA, HRSHT, HRSLATE

** initialising the variables

HRXTRA = 0.00
HRSHT = 0.00
HRSLATE= 0.00
XTRAHR = 0.00
ERLYIN = 0.00
ERLYOUT= 0.00

STRIN = str(IN, 5, 2)
MININ = val(substr(STRIN, 4, 2))
HRSIN = val(substr(STRIN, 1, 2))
VLMIN = MININ/60.0
TIMEIN= HRSIN + VLMIN

if TIMEIN <= 7.75
    ERLYIN = 7.75 - TIMEIN
else
    if TIMEIN > 7.92
        HRSLATE = TIMEIN - 7.75
    endif
endif

OUTSTR = str(OUT, 5, 2)
OUTMIN = val(substr(OUTSTR, 4, 2))
OUTHR = val(substr(OUTSTR, 1, 2))
VALMIN = OUTMIN/60.0
TIMEOUT= OUTHR + VALMIN

if TIMEOUT >= 16.25
    XTRAHR = TIMEOUT - 16.25
else
    ERLYOUT = 16.25 - TIMEOUT
endif
HRXTRA = ERLYIN+XTRAHR
HRSHT = HRSLATE+ERLYOUT
return

```

```

** TIMEBDIFF.PRG
** Program to calculate the difference in time with standard time
** for shift B employees

parameters IN, OUT, HRXTRA, HRSHT, HRSLATE

** initialising the variables

HRXTRA = 0.00
HRSHT = 0.00
HRSLATE= 0.00
XTRAHR = 0.00
ERLYIN = 0.00
ERLYOUT= 0.00

** taking the value and checking with standard time

STRIN = str(IN, 5, 2)
MININ = val(substr(STRIN, 4, 2))
HRSIN = val(substr(STRIN, 1, 2))
VLMIN = MININ/60.0
TIMEIN= HRSIN + VLMIN

if TIMEIN < 9.00
    ERLYIN = 9.00 - TIMEIN
else
    if TIMEIN > 9.17
        LATEIN = TIMEIN - 9.00
        HRSLATE= LATEIN
    endif
endif

OUTSTR = str(OUT, 5, 2)
OUTMIN = val(substr(OUTSTR, 4, 2))
OUTHR = val(substr(OUTSTR, 1, 2))
VALMIN = OUTMIN/60.0
TIMEOUT= OUTHR + VALMIN

if TIMEOUT >= 17.50
    XTRAHR = TIMEOUT - 17.50
else
    ERLYOUT = 17.50 - TIMEOUT
endif

HRXTRA = ERLYIN+XTRAHR
HRSHT = HRSLATE+ERLYOUT
return

```

```
** TMNEXTDT.PRG
** Program for changing the date according to the calender
```

```
parameters XDATE, DAYS
```

```
D = substr(XDATE, 5, 2)
```

```
M = substr(XDATE, 3, 2)
```

```
Y = substr(XDATE, 1, 2)
```

```
VAL_D = val(D)+1
```

```
V_DAYS = val(DAYS)
```

```
if VAL_D<10
```

```
    D = '0'+str(VAL_D,1)
```

```
else
```

```
    D = str( VAL_D, 2 )
```

```
endif
```

```
VAL_M = val(M)
```

```
if VAL_M<10
```

```
    M = '0'+str(VAL_M,1)
```

```
else
```

```
    M = str( VAL_M, 2 )
```

```
endif
```

```
XDATE = Y+M+D
```

```
return
```

Chaper Institute of Engg. & Tech.
PATIAL-147001
CENTRAL LIBRARY

Acc. No. 90461 Dt. 27-5-94