

Academia de Studii Economice
Facultatea de Cibernetică, Statistică și Informatică Economică

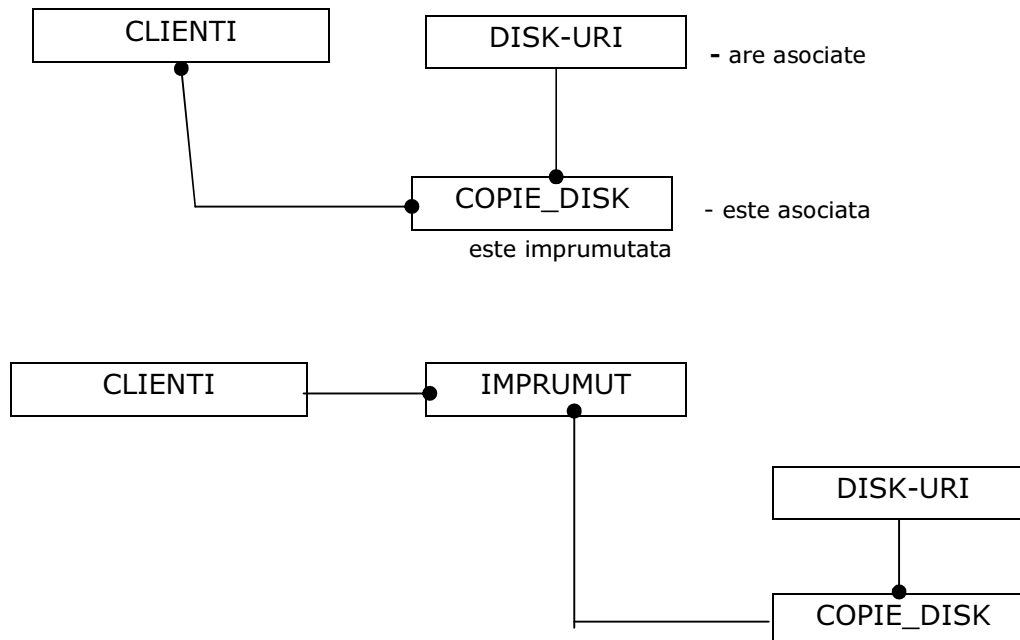
Proiect

- Baze de Date -

1. Tema proiectului :

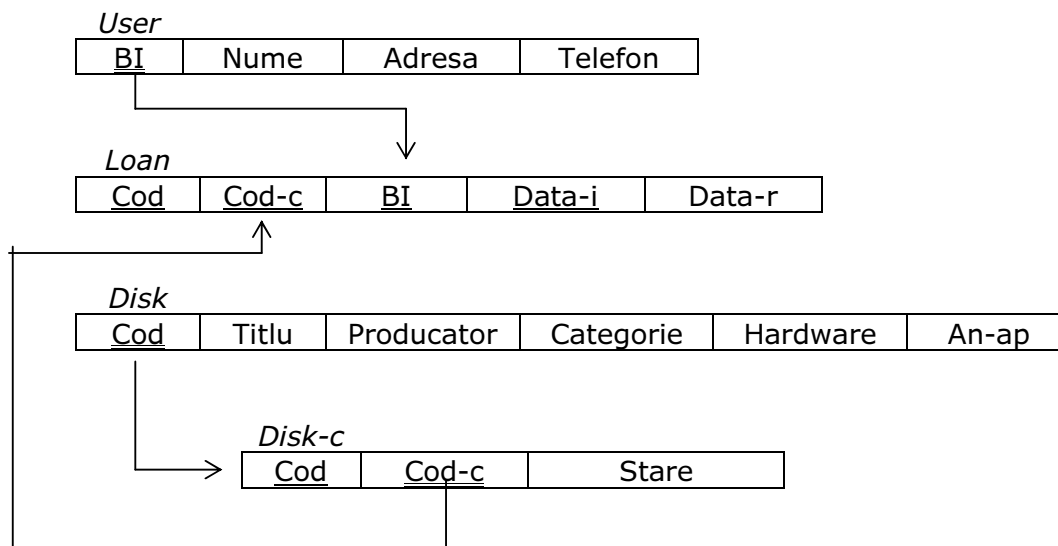
Sa se realizeze un program Fox-Pro impreuna cu bazele de date necesare pentru gestionarea unui centru de inchirieri de CD-uri cu jocuri.

2. Modelul entitate-asociat



Exista relatii m:m care vor fi transformate in relatii 1:m sau m:1.

3. Schema conceptuala a BD



4. Normalizarea Bazei de Date

Initial datele problemei aveau urmatoarele caracteristici:

Client

BI

Date prsonale: - Nume } structura de grup
 - Adresa }
 - Telefon }

Disk-uri

Cod

Cod copie

Caracteristici (titlu,producator,categorie,hardware,an_aparitie)

Stare

Imprumut

Cod

Cod copie

BI

Data_i

Data_r

Cheile identificate sunt: BI pentru *Clienti*, *Cod* si *Cod_copie* pentru *Disk-uri*, BI,Cod,Cod_copie si Data_i pentru *Imprumut*.

Dupa eliminarea campurilor repetitive structura *Disk* se va diviza in doua tabele: *Disk* si *Disk_c*.

DISK: Cod,Titlu,Producator,Categorie,Hardware,An_aparitie;

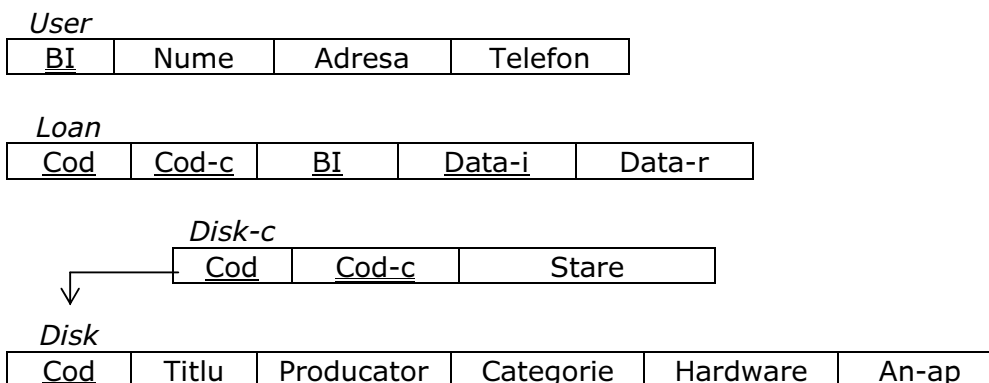
DISK_c:Cod,Cod_c,Stare.

Se elimina structurile de grup din tabela *Client* :

USER:BI,Nume,Adresa,Telefon.

Datorita faptului ca intre campurile tabelelor nu mai exista dependente de nici un alt fel (functionale, incomplete sau tranzitive) putem considera tabelele ca fiind in FN3.

In acest moment schema sistemului arata astfel:



5. Structura tabelelor ce compun BD

Nume	Tip cheie	Val. imp.	Check	Tab. Care ref.	Tip data	Lungime	Not null/unic
------	-----------	-----------	-------	----------------	----------	---------	---------------

USER.DBF

BI	Ch. Externa Ch. Primara	-	-	-	Caracter	10	Not null/ unic
Nume	-	-	-	-	Caracter	30	Not null
Adresa	-	-	-	-	Caracter	30	Not null
Telefon	-	-	-	-	Numeric	9	-

LOAN.DBF

Cod	Ch. Primara	-	-	-	Caracter	5	-
Cod_c	Ch. Primara	-	-	DISK_C	Caracter	5	-
BI	Ch. Primara	-	-	USER	Caracter	10	-
Data_I	Ch. Primara	Date()	-	-	Data	8	-
Data_r	-	Date()+3	-	-	Data	8	-

DISK_C.DBF

Cod	Ch. Primara	-	-	DISK	Caracter	5	-
Cod_c	Ch. Externa Ch. Primara	-	-	-	Caracter	5	Not null/ unic
Stare	-	LIBERA	-	-	Caracter	11	-

DISK.DBF

Cod	Ch. Externa Ch. Primara	-	-	-	Caracter	5	Not null/ unic
Titlu	-	-	-	-	Caracter	30	
Categorie	-	-	-	-	Caracter	15	
Producator	-	-	-	-	Caracter	15	
Hardware	-	-	-	-	Memo	-	
An_ap	-	-	-	-	Numeric	4	

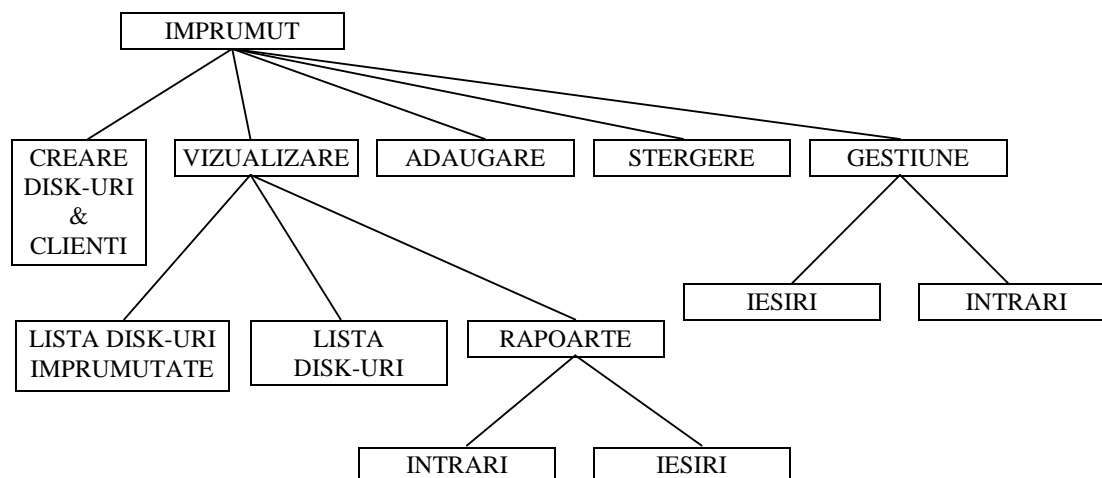
6. Schema interna a BD

USER.DBF;
USER.IDX.

LOAN.DBF;
I1.IDX;I2.IDX;
I3.IDX;I4.IDX.

DISK.DBF;
DISK.IDX.

DISK_C.DBF;
DISK_C1.IDX;
DISK_C2.IDX.



7. Listarea programului

```

close all
deactivate wind command
set deleted on
set talk off
set clock on
clear
set safety off
set procedure to pr.prg
close databases
use disk
index on cod to disk.idx
use disk_c
index on cod to disk_c1.idx
index on cod_c to disk_c2.idx
use user
index on bi to user.idx
use loan
index on bi to il.idx
index on data_i to i2.idx
index on cod_c to i3.idx
index on cod to i4.idx
close databases
define window sis from 1,0 to 24,79 double close
activate window sis
set sysmenu to
define menu _msysmenu

define pad p1 of _msysmenu prompt '\<Actualizare'
define pad p2 of _msysmenu prompt '\<Imprumuturi'
define pad p3 of _msysmenu prompt '\<Restituirii'
define pad p4 of _msysmenu prompt 'Ra\<poarte'
define pad p5 of _msysmenu prompt 'I\<esire'

on pad p1 of _msysmenu activate popup popact
on pad p4 of _msysmenu activate popup poprap
on selection pad p5 of _msysmenu do iesire
on selection pad p2 of _msysmenu do impr
on selection pad p3 of _msysmenu do rest

defi popup poprap
defi bar 1 of poprap prompt '\<Lista Disk-urilor'
defi bar 2 of poprap prompt '\<Lista disk-uri inchiriate'
defi bar 3 of poprap prompt 'L\<ista clienti intarziati'
defi bar 4 of poprap prompt 'Lista \<jocurilor'
on selection bar 1 of poprap do rep5
on selection bar 2 of poprap do rep4
on selection bar 3 of poprap do rep1
on bar 4 of poprap activate popup poprap1

defi popup poprap1
defi bar 1 of poprap1 prompt '\<Dupa categorii'
defi bar 2 of poprap1 prompt '\<Dupa producator'

on selection bar 1 of poprap1 do rep3
on selection bar 2 of poprap1 do rep2

defi popup popact
defi bar 1 of popact prompt '\<Disk-uri '
defi bar 2 of popact prompt 'C\<lienti '

```

```

on bar 1 of popact activate popup cd
on bar 2 of popact activate popup client

define popup cd
defi bar 1 of cd prompt '\<Adaugare date'
defi bar 2 of cd prompt '\<Modificare date '
defi bar 3 of cd prompt '\<Stergere date '

on bar 1 of cd activate popup cd1
on bar 2 of cd activate popup mod
on bar 3 of cd activate popup ste

defi popup cd1
defi bar 1 of cd1 prompt '\<Disk nou'
defi bar 2 of cd1 prompt '\<Copie noua'

on selection bar 1 of cd1 do adaug1
on selection bar 2 of cd1 do adaug12

defi popup mod
defi bar 1 of mod prompt '\<Disk'
defi bar 2 of mod prompt 'C\<opie Disk'

on selection bar 1 of mod do modif1
on selection bar 2 of mod do modif12

defi popup ste
defi bar 1 of ste prompt '\<Disk'
defi bar 2 of ste prompt 'C\<opie Disk'

on selection bar 1 of ste do sterg1
on selection bar 2 of ste do sterg12

define popup client
defi bar 1 of client prompt '\<Adaugare client in BD'
defi bar 2 of client prompt '\<Modificare date client'
defi bar 3 of client prompt '\<Stergere date client'

on selection bar 1 of client do adaug2
on selection bar 2 of client do modif2
on selection bar 3 of client do sterg2

activate menu _msysmenu

*****

procedure iesire
set sysmenu to default
close databases
deactivate window sis
set clock off
clear
return

*****

proc adaug1
clear
use disk
set index to disk.idx
reindex
c=space(5)

```

```

a=1
b=2
do while b=2
    @ 2,2 say 'Disk-uri'
    @ 5,5 say 'Cod disk (X-Iesire) ' get c valid not empty(c) error
'Codul nu poate fi nul'
    read
    if upper(c)!='X'
        seek c
        if found()
            @ 7,5 say 'Cod duplicat'
            @ 20,4 get b function '*h \<Iesire;\<Reluare' size 2,12,3
            read
            c=space(5)
        else
            append blank
            scatter memvar
            m.cod=c
            @ 7,5 say 'Tiltu      ' get m.titlu
            @ 8,5 say 'Hardware'
            @ 9,5 edit hardware size 5,21
            @ 15,5 say 'Categorie ' get m.categorie function '^
\<RPG;\<ARCADE;\<3D Shooter;3\D Action;STRATEGIE;DESKTOP' valid not
empty(m.categorie)
            @ 18,5 say 'Producator ' get m.producator
            @ 19,5 say 'An aparitie ' get m.an_ap VALID (m.an_ap>1950)
.and. (m.an_ap<2000) error 'An incorect.Reintroduceti'
            read
            @ 20,4 get a function '*h \<Salvare;\<Anulare' size 2,12,3
            read
            @ 20,4 say '          '

            if a=1
                gather memvar
                do adaug11
                sele 1
                @ 20,4 get b function '*h \<Iesire;\<Continuare' size 2,12,3
            read
            clea
            else
                delete
            endif
            c=space(5)
        endif
    else
        b=1
    endif
    clea
enddo
pack
clea
return

*****

procedure adaug11
clea
n=c
sele 2
use disk_c
set index to disk_c2.idx
reindex
c1=space(5)

```

```

al=1
b1=2
do while b1=2
    @ 1,10 say 'Copii Disk-uri' style 'BI' font 'Courier' , 20
    @ 7,5 say 'Cod Disk :'
    ?? n
    @ 10,5 say 'Cod copie ' get c1 valid not empty(c1) error 'Codul nu
poate fi nul'
    read
        seek c1
        if found()
            @ 17,5 say 'Cod duplicat'
            @ 20,4 get b1 function '*h \<Iesire;\<Reluare' size 2,12,3
            read
            c1=space(5)
        else
            append blank
            scatter memvar
            m.cod_c=c1
            m.cod=n
            @ 12,5 say 'Stare LIBER ' style 'b'
            m.stare='LIBER'
            @ 20,4 get a1 function '*h \<Salvare;\<Anulare' size
2,12,3
            read
            @ 20,4 say ' '
            @ 20,4 get b1 function '*h \<Iesire;\<Continuare' size
2,12,3
            read
            if a1=1
                gather memvar
                clea
            else
                delete
            endif
            c1=space(5)
        endif
    endif
    clea
enddo
pack
clea
return

*****

proc adaug12
clea
use disk
set index to disk.idx
reindex
c=space(5)
a=1
b=2
do while b=2
    @ 1,10 say 'Disk-uri'
    @ 15,5 say 'Cod disk a carui copie doriti sa o introduceti (X-
Iesire) ' get c valid not empty(c) error 'Codul nu poate fi nul'
    read
    if upper(c)!='X'
        seek c
        if not found()
            @ 7,5 say 'Cod inexistent'
            @ 20,4 get b function '*h \<Iesire;\<Reluare' size 2,12,3

```



```

        read
        c=space(5)
    else
        do adaug11
            sele 1
            @ 20,4 get b function '*h \<Iesire;\<Continuare' size 2,12,3
            read
            clea
            c=space(5)
        endif
    else
        b=1
    endif
    clea
enddo
pack
clea
return

*****

proc modif1
clea
use disk
set index to disk.idx
reindex
c=space(5)
a=1
b=2
do while b=2
    @ 1,10 say 'Modificare date disk-uri'
    @ 5,5 say 'Cod disk ' get c valid not empty(c) error 'Codul nu
poate fi nul'
    read
    seek c
    if not found()
        @ 7,5 say 'Cod inexistent'
        @ 20,4 get b function '*h \<Iesire;\<Reluare' size 2,12,3
        read
        c=space(5)
    else
        scatter memvar
        m.cod=c
        @ 4,5 say 'Titlu: '
        ?? titlu
        @ 6,5 say 'Categorie: '
        ?? categorie
        @ 10,5 say 'Producator: '
        ?? producator
        @ 12,5 say 'Anul Aparitiei '
        ?? an_ap
        @ 5,5 say 'Titlul modificat ' color 'w+/b' get m.titlu
valid not empty(m.titlu) error 'Nume vid.Reintroduceti'
        @ 7,5 say 'Categorie modificata ' color 'w+/b' get m.categorie
function '^ \<RPG;\<ARCADE;\<3D Shooter;3\<D Action;STRATEGIE;DESKTOP'
valid not empty(m.categorie)
        @ 11,5 say 'Producator modificat ' color 'w+/b' get
m.producator
        @ 13,5 say 'An aparitie modificat' color 'w+/b' get m.an_ap
VALID (m.an_ap>1950) .and. (m.an_ap<2000) error 'An
incorect.Reintroduceti'
        read
        @ 14,5 say 'Hardware'

```

```

        @ 15,5 edit hardware size 4,21

        @ 20,4 get a function '*h \<Modificare;\<Anulare' size 2,12,3
        read
        @ 20,4 say '
        @ 20,4 get b function '*h \<Iesire;\<Continuare' size 2,12,3
        read

        if a=1
            gather memvar
        endif
        c=space(5)
    endif
    clea
enddo
clea
close databases
return
*****

proc modif12
clea
use disk_c
set index to disk_c2.idx
reindex
c=space(5)
a=1
b=2
do while b=2
    @ 1,10 say 'Modificare date copie disk'
    @ 5,5 say 'Cod copie ' get c valid not empty(c) error 'Codul nu
poate fi nul'
    read
    seek c
    if not found()
        @ 7,5 say 'Cod inexistent'
        @ 20,4 get b function '*h \<Iesire;\<Reluare' size 2,12,3
        read
        c=space(5)
    else
        @ 20,4 say '
        scatter memvar
        m.cod_c=c
        @ 7,5 say 'Starea disk-ului: ' style 'b'
        ?? m.stare
        @ 10,5 say 'Starea modificata ' color 'w+/b' get m.stare
function '^ \<LIBER;\<DISTRUSA;\I\<MPRUMUTATA'
        read
        @ 20,4 get a function '*h \<Modificare;\<Anulare' size 2,12,3
        read
        @ 20,4 say '
        @ 20,4 get b function '*h \<Iesire;\<Continuare' size 2,12,3
        read

        if a=1
            gather memvar
        endif
        c=space(5)
    endif
    clea
enddo
clea
close databases

```

```

return
*****

proc sterg1
close databases
clea
use disk
if eof()
@ 10,10 say 'Nu exista disk-uri introduce in BD'
wait window 'Apasati o tasta'
else
set index to disk.idx
reindex
c=space(5)
a=1
b=2
do while b=2
    sele 1
        @ 1,10 say 'Stergere date disk'
        @ 5,5 say 'Cod disk ' get c valid not empty(c) error 'Codul nu
poate fi nul'
        read
        seek c
        if not found()
            @ 7,5 say 'Cod inexistent'
            @ 20,4 get b function '*h \<Iesire;\<Reluare' size 2,12,3
            read
            c=space(5)
        else
            @ 20,4 say '
            @ 7,5 say 'Titlu: ' color 'w+/b'
            ?? titlu
            @ 10,5 say 'Categorie: ' color 'w+/b'
            ?? categorie
            @ 13,5 say 'Producator: ' color 'w+/b'
            ?? producator
            use
            sele 1
            use disk
            set index to disk.idx
            reindex
            sele 2
            use disk_c
            set index to disk_c1.idx
            sele 1
            set relation to cod into disk_c
            set skip to disk_c
            seek c
            sele 2
            if not eof()
                @ 17,4 say 'Acest disk are copii inregistrate in BD' color
'W+/b'
                @ 18,4 say 'Stergerea lui va determina stergerea copiilor!!!'
color 'w+/b'
                @ 20,4 get a function '*h \<Stergere;\<Anulare' size 2,12,3
                read
                @ 20,4 get b function '*h \<Iesire;\<Continuare' size 2,12,3
                read
                if a=1
                    go top
                    do while (not eof()) .and. (cod=c)
                        delete
                        skip 1

```

```

        enddo
        pack
        sele 1
        seek c
        delete
        pack
        else
        sele 1
        endif

else
@ 20,4 say 'Aceast disk nu mai are nici o copie in BD' color
'W+/B'
@ 20,4 get a function '*h \<Stergere;\<Anulare' size 2,12,3
read
@ 20,4 say '
@ 20,4 get b function '*h \<Iesire;\<Continuare' size 2,12,3
read

        if a=1
        delete
        pack
        endif
        endif
        c=space(5)
endif
clea

enddo
endif
clea
close databases
return

*****

proc sterg12
close databases
clea
use disk_c
set index to disk_c2.idx
reindex
c=space(5)
a=1
b=2
do while b=2
@ 1,10 say 'Stergere date copii disk-uri'
@ 5,5 say 'Cod copie disk ' get c valid not empty(c) error 'Codul
nu poate fi nul'
read
seek c
if not found()
@ 7,5 say 'Cod inexistent'
@ 20,4 get b function '*h \<Iesire;\<Reluare' size 2,12,3
read
c=space(5)
else
@ 20,4 say '
@ 10,5 say 'Cod disk:' color 'w+/b'
?? cod style 'b'
@ 11,5 say 'Stare: ' color 'w+/b'
?? stare style 'b'
@ 20,4 get a function '*h \<Stergere;\<Anulare' size 2,12,3
read

```

```

        @ 20,4 say '
        @ 20,4 get b function '*h \<Iesire;\<Continuare' size 2,12,3
        read

        if a=1
            delete
            pack
        endif
        c=space(5)
    endif
    clea
enddo
clea
close databases
return

*****

proc adaug2
clea
clea
use user
set index to user.idx
reindex
c=space(5)
a=1
b=2
do while b=2
    @ 1,10 say 'Clienti' style 'BI'
    @ 5,5 say 'Serie buletin ' get c valid not empty(c) error 'Seria nu
poate fi nula'
    read
    seek c
    if found()
        @ 7,5 say 'Serie existenta'
        @ 20,4 get b function '*h \<Iesire;\<Reluare' size 2,12,3
        read
        c=space(5)
    else
        @ 20,4 say '
        append blank
        scatter memvar
        m.bi=c
        @ 7,5 say 'Nume ' get m.nume valid not empty(m.nume)
error 'Nume vid.Reintroduceti'
        @ 9,5 say 'Adresa ' get m.adresa valid not
empty(m.adresa) error 'Adresa vida.Reintroduceti'
        @ 11,5 say 'Telefon ' get m.telefon
        read
        @ 20,4 get a function '*h \<Salvare;\<Anulare' size 2,12,3
        read
        @ 20,4 say '
        @ 20,4 get b function '*h \<Iesire;\<Continuare' size 2,12,3
        read

        if a=1
            gather memvar
        else
            delete
        endif
        c=space(5)
    endif
    clea

```

```

enddo
pack
clea
close databases
return

*****

proc modif2
clea
use user
set index to user.idx
reindex
c=space(5)
a=1
b=2
do while b=2
    @ 1,10 say 'Modificare date clienti'
    @ 5,5 say 'Serie buletin ' get c valid not empty(c) error 'Seria nu
poate fi nula'
    read
    seek c
    if not found()
        @ 7,5 say 'Serie inexistentă'
        @ 20,4 get b function '*h \<Iesire;\<Reluare' size 2,12,3
        read
        c=space(5)
    else
        @ 20,4 say '
scatter memvar
m.bi=c
@ 7,5 say 'Nume: '
?? nume
@ 11,5 say 'Adresa: '
?? adresa
@ 15,5 say 'Telefon: '
?? telefon
        @ 8,5 say 'Numele modificat ' color 'w+/b' get m.nume valid
not empty(m.nume) error 'Nume vid.Reintroduceti' default ''
        @ 12,5 say 'Adresa modificata' color 'w+/b' get m.adresa
valid not empty(m.adresa) error 'Adresa vida.Reintroduceti' default ''
        @ 16,5 say 'Telefon modificat' color 'w+/b' get m.telefon
default 0
        read
        @ 20,4 get a function '*h \<Modificare;\<Anulare' size 2,12,3
        read
        @ 20,4 say '
@ 20,4 get b function '*h \<Iesire;\<Continuare' size 2,12,3
        read

        if a=1
            gather memvar
        endif
        c=space(5)
    endif
    clea
enddo
clea
close databases
return

*****

```

```

proc sterg2
close databases
clea
use user
set index to user.idx
reindex
c=space(5)
a=1
b=2
do while b=2
    @ 1,10 say 'Stergere date clienti'
    @ 5,5 say 'Serie buletin ' get c valid not empty(c) error 'Seria nu
poate fi nula'
    read
    seek c
    if not found()
        @ 7,5 say 'Serie inexistentă'
        @ 20,4 get b function '*h \<Iesire;\<Reluare' size 2,12,3
        read
        c=space(5)
    else
        @ 20,4 say '
        @ 7,5 say 'Nume: ' color 'w+/b'
        ?? nume style 'b'
        @ 11,5 say 'Adresa: ' color 'w+/b'
        ?? adresa style 'b'
        @ 15,5 say 'Telefon: ' color 'w+/b'
        ?? telefon style 'b'
        use
        sele 1
        use user
        set index to user.idx
        reindex
        sele 2
        use loan
        set index to il.idx
        reindex
        sele 1
        set relation to bi into loan
        set skip to loan
        seek c
        sele 2
        if not eof()
            @ 20,4 say 'Acest client are disk-uri imprumutate' color 'W+/b'
            @ 23,4 say 'Mergeti intai la restituiiri' color 'w+/b'
            wait window 'Apasati o tasta'
            b=1
        else
            @ 20,4 get a function '*h \<Stergere;\<Anulare' size 2,12,3
            read
            @ 20,4 say '
            @ 20,4 get b function '*h \<Iesire;\<Continuare' size 2,12,3
            read

            if a=1
                delete
                pack
            endif
            endif
            c=space(5)
        endif
        clea
    enddo
enddo

```

```

clea
close databases

return
*****

proc impr
clea
sele 1
use user
set index to user.idx
reindex
sele 2
use loan
set index to il.idx
reindex
sele 3
use disk
set index to disk.idx
reindex
sele 4
use disk_c
set index to disk_c1.idx
reindex

c=space(5)
c1=space(5)
a=1
b=2
do while b=2
    sele 1
        @ 1,10 say 'Imprumuturi'
        @ 5,5 say 'Serie buletin client ' get c valid not empty(c) error
'Seria nu poate fi nula'
        read
        reindex
        seek c
        if not found()
            @ 7,5 say 'Serie inexistentă'
            @ 8,5 say 'Introduceti intai datele clientului'
            @ 20,4 get b function '*h \<Iesire;\<Reluare introducere' size
2,12,3
            read
            c=space(5)
        else
            @ 7,5 say 'Cod disk' get c1 valid not empty(c1) error 'Codul
nu poate fi nul'
            read
            sele 3
            reindex
            seek c1
            if not found()
                @ 7,5 say '
                @ 7,5 say 'Cod inexistent'
                @ 20,4 get b function '*h \<Iesire;\<Reluare introducere'
size 2,12,3
            read
            else
                sele 3
                set relation to cod into disk_c
                set skip to disk_c
                seek c1
                sele 4

```



```

if eof()
  @ 12,5 say 'Nu exista copii pentru aceast disk'
else
  go top
  j=0
  for i=1 to reccount()
    go i
    if cod=c1 .and. stare='LIBER'
      j=1
      nr=recno()
      i=reccount()
    endif
  endfor
  if j!=0
    go nr
    c1=cod
    c2=cod_c
    sele 2
    appe blank
    scatter memvar
    m.cod_c=c2
    m.cod=c1
    m.bi=c
    m.data_i=date()
    m.data_r=date()+3
    @ 10,5 say 'Cod copie:      '
    ??m.cod_c
    @ 12,5 say 'Data imprumut:  '
    ??m.data_i
    @ 14,5 say 'Data restituire:'
    ??m.data_r style 'b'
    @ 20,4 get a function '*h \<Salvare;\<Anulare' size
2,12,3
    read
    @ 20,4 say '
    @ 20,4 get b function '*h \<Iesire;\<Continuare' size
2,12,3
    read
    if a=1
      gather memvar
      sele 4
      go nr
      replace stare with 'IMPRUMUTAT'
    else
      delete
      pack
    endif
  else
    @ 10,5 say 'Nu exista copii disponibile'
    @ 20,4 get b function '*h \<Iesire;\<Continuare' size
2,12,3
    read
  endif
endif
endif
endif
c=space(5)
  clea
enddo
clea
close databases
return

```

```

*****

proc rest
close databases
clea
sele 1
use loan
set index to il.idx
reindex
sele 2
use disk_c
set index to disk_c2.idx
reindex
sele 1
b=2
c=space(5)
do while b=2
    @ 1,10 say 'Restituiri'
    @ 5,5 say 'Serie buletin client ' get c valid not empty(c) error
'Seria nu poate fi nula'
    read
    seek c
    if not found()
        @ 14,5 say 'Acest client nu are imprumutat nici un disk'
        @ 20,4 get b function '*h \<Iesire;\<Reluare introducere' size
2,12,3
        read
        c=space(5)
        clea
    else
        @ 7,5 say 'Acest client a imprumutat:'
        for i=1 to reccount()
            sele 1
            set index to il.idx
            go i
            if bi=c
                cl=cod_c
                @ 9,5 say 'Cod disk: '
                ??cod
                @ 11,5 say 'Cod copie: '
                ??cod_c
                cc=cod_c
                nr=recno()
                @ 13,5 say 'Data imprumut: '
                ??data_i
                @ 15,5 say 'Data la care trebuia restituita: '
                ??data_r
                pen=date()-data_r
                if pen>0
                    @ 17,5 say 'Clientul trebuie sa plateasca penalizari de:'
style 'b'
                    ??pen*5000,' lei' style 'b'
                else
                    @ 17,5 say '
,
                    @ 17,5 say 'Clientul nu are penalizari de platit'
endif
                @ 19,5 say 'Restituirea a fost inregistrata'
                use
                use loan
                index on cod_c to i3.idx
                use

```

```

        use loan
        set index to i3.idx
        sele 2
        use
        use disk_c
        index on cod_c to disk_c2
        set index to disk_c2.idx
        sele 1
        set relation to cod_c into disk_c
        seek c1
        sele 2
        repl stare with 'LIBER'
        sele 1
        delete
        wait window
    endif
endfor
@ 21,4 get b function '*h \<Iesire;\<Continuare' size 2,12,3
read
endif
c=space(5)
clea
enddo
sele 1
pack
clea
close databas
return

*****

proc repl
clea
use loan
j=0
for i=1 to reccount()
go i
if date(>data_r
j=1
i=reccount()
endif
endif
if j=0
clea
@ 10,10 say 'Nu exista clienti intarziati !'
else
repo form loan.frx for date(>data_r
endif
wait window 'O tasta pentru stergere'
clea
close databases
return

*****

proc rep2
close databases
clea
use disk
c2=space(15)
a=1
b=2
do while b=2

```

```

        @ 1,10 say 'Urmeaza lista jocurilor'
        @ 5,5 say 'Introduceti producatorul dorit : ' get c2 valid not
empty(c2) error 'Introducere incorecta'
        read
        locate for producator=c2
        if not found()
            @ 7,5 say 'Producatorul acesta nu exista in BD !'
            @ 20,4 get b function '*h \<Iesire;\<Reluare' size 2,12,3
            read
            c2=space(15)
        else
            clea
            go top
            repo form disk.frx for producator=c2
            wait window 'Tasta pentru stergere'
            clea
            b=1
        endif
        clea
    enddo
    clea
    close databases
    return

*****

proc rep3
close databases
clea
use disk
c=space(10)
b=2
do while b=2
    @ 1,10 say 'Urmeaza lista jocurilor'
    @ 15,5 say 'Categoria dorita ' get c function '^ \<RPG;\<ARCADE;\<3D
Shooter;3\<D Action;STRATEGIE;DESKTOP' valid not empty(c)
    read
    locate for categorie=c
    if not found()
        @ 7,5 say 'Categoria aceasta nu exista in BD'
        @ 20,4 get b function '*h \<Iesire;\<Reluare' size 2,12,3
        read
        c=space(10)
    else
        clea
        go top
        repo form disk.frx for categorie=c
        wait window 'Tasta pentru stergere'
        clea
        b=1
    endif
    clea
enddo
clea
close databases
return

*****

proc rep4
clea
sele 1
use disk

```

```

set index to disk.idx
reindex
sele 2
use disk_c
set index to disk_cl.idx
reindex
set relation to cod into disk
locate for stare='IMPRUMUTAT'
if found()
@ 2,10 say 'Disk-uri imprumutate'
@ 4,10 say '
? '-----'
-----'
? '| Cod disk | Cod copie | Titlu |
Categorie |'
? '-----'
-----'
for i=1 to reccount()
go i
if stare='IMPRUMUTAT'
? '| ',cod,' | ',cod_c,' | ',a.titlu,'|',a.categorie,'|'
? '-----'
-----'

endif
endifor
wait window 'Tasta pentru stergere'
else
@ 10,10 say 'Nu exista disk-uri imprumutate !!!!!'
wait window 'Tasta pentru iesire'
endif
clea
close databases
return

*****
proc rep5
close databases
clea
use disk
repo form disk.frx
wait window 'Tasta pentru stergere'
clea
close databases
return
*****

```