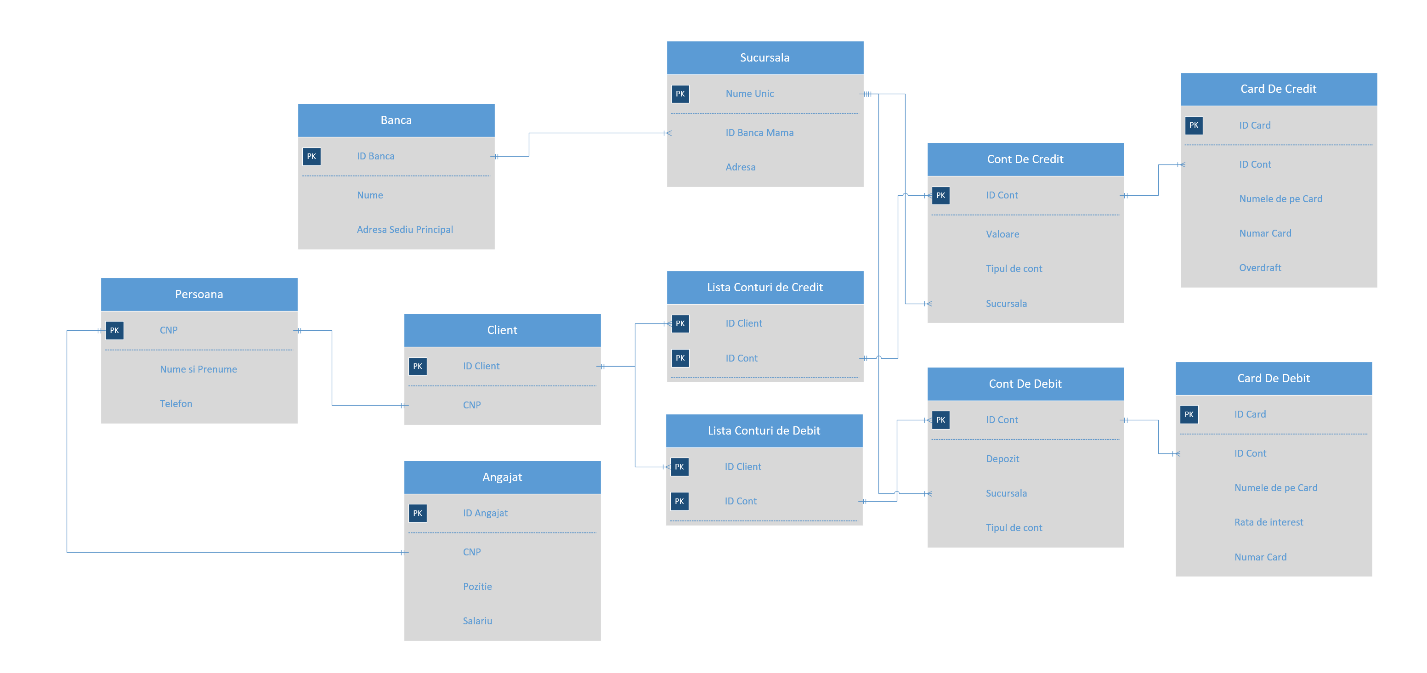
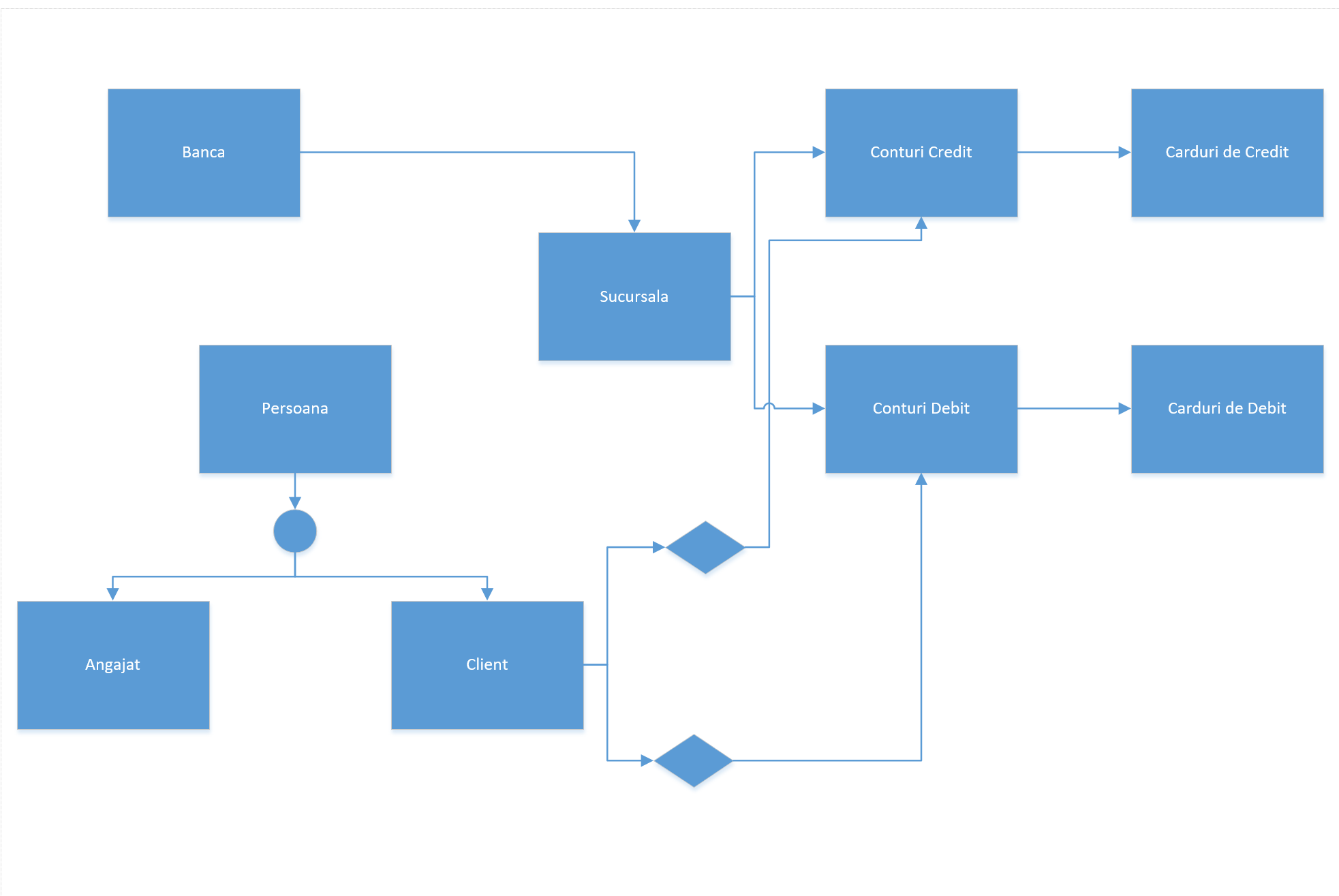
**Database Design Project 25**

**Student: Turcu Gabriel**

**CEN4.S**

**Teacher: Liana Stănescu**

**Diagrams**

****

**Create Statements**

**Angajat:**

CREATE TABLE "ANGAJAT"

( "ID\_ANGAJAT" NUMBER,

"CNP" NUMBER,

"POZITIE" VARCHAR2(30),

"SALARIU" NUMBER,

CONSTRAINT "ANGAJAT\_PK" PRIMARY KEY ("ID\_ANGAJAT") ENABLE

) ;ALTER TABLE "ANGAJAT" ADD CONSTRAINT "ANGAJAT\_FK" FOREIGN KEY ("CNP")

REFERENCES "PERSOANA" ("CNP") ENABLE;

CREATE OR REPLACE TRIGGER "BI\_ANGAJAT"

before insert on "ANGAJAT"

for each row

begin

if :NEW."ID\_ANGAJAT" is null then

select "ANGAJAT\_SEQ".nextval into :NEW."ID\_ANGAJAT" from dual;

end if;

end;

/

ALTER TRIGGER "BI\_ANGAJAT" ENABLE;

**Banca:**

CREATE TABLE "BANCA"

( "ID\_BANCA" NUMBER,

"NUME" VARCHAR2(30),

"ADRESA\_SEDIU\_PRINCIPAL" VARCHAR2(500),

CONSTRAINT "BANCA\_PK" PRIMARY KEY ("ID\_BANCA") ENABLE

) ;

CREATE INDEX "BANCA\_IDX1" ON "BANCA" ("ID\_BANCA", "NUME")

;

CREATE OR REPLACE TRIGGER "BI\_BANCA"

before insert on "BANCA"

for each row

begin

if :NEW."ID\_BANCA" is null then

select "BANCA\_SEQ".nextval into :NEW."ID\_BANCA" from dual;

end if;

end;

/

ALTER TRIGGER "BI\_BANCA" ENABLE;

**Card De Credit:**

CREATE TABLE "CARD\_DE\_CREDIT"

( "ID\_CARD" NUMBER,

"ID\_CONT" NUMBER,

"NUMELE\_DE\_PE\_CARD" VARCHAR2(30),

"NUMAR\_CARD" NUMBER,

"OVERDRAFT" NUMBER,

CONSTRAINT "CARD\_DE\_CREDIT\_PK" PRIMARY KEY ("ID\_CARD") ENABLE

) ;ALTER TABLE "CARD\_DE\_CREDIT" ADD CONSTRAINT "CARD\_DE\_CREDIT\_FK" FOREIGN KEY ("ID\_CONT")

REFERENCES "CONT\_DE\_CREDIT" ("ID\_CONT") ENABLE;

CREATE OR REPLACE TRIGGER "BI\_CARD\_DE\_CREDIT"

before insert on "CARD\_DE\_CREDIT"

for each row

begin

if :NEW."ID\_CARD" is null then

select "CARD\_DE\_CREDIT\_SEQ".nextval into :NEW."ID\_CARD" from dual;

end if;

end;

/

ALTER TRIGGER "BI\_CARD\_DE\_CREDIT" ENABLE;

**Card De Debit:**

CREATE TABLE "CARD\_DE\_DEBIT"

( "ID\_CARD" NUMBER,

"ID\_CONT" NUMBER,

"NUMELE\_DE\_PE\_CARD" VARCHAR2(30),

"RATA\_DE\_INTEREST" NUMBER,

"NUMAR\_CARD" NUMBER,

CONSTRAINT "CARD\_DE\_DEBIT\_PK" PRIMARY KEY ("ID\_CARD") ENABLE

) ;ALTER TABLE "CARD\_DE\_DEBIT" ADD CONSTRAINT "CARD\_DE\_DEBIT\_FK" FOREIGN KEY ("ID\_CONT")

REFERENCES "CONT\_DE\_DEBIT" ("ID\_CONT") ENABLE;

CREATE OR REPLACE TRIGGER "BI\_CARD\_DE\_DEBIT"

before insert on "CARD\_DE\_DEBIT"

for each row

begin

if :NEW."ID\_CARD" is null then

select "CARD\_DE\_DEBIT\_SEQ".nextval into :NEW."ID\_CARD" from dual;

end if;

end;

/

ALTER TRIGGER "BI\_CARD\_DE\_DEBIT" ENABLE;

**Client:**

CREATE TABLE "CLIENT"

( "ID\_CLIENT" NUMBER,

"CNP" NUMBER,

CONSTRAINT "CLIENT\_PK" PRIMARY KEY ("ID\_CLIENT") ENABLE

) ;ALTER TABLE "CLIENT" ADD CONSTRAINT "CLIENT\_FK" FOREIGN KEY ("CNP")

REFERENCES "PERSOANA" ("CNP") ENABLE;

CREATE INDEX "CLIENT\_IDX1" ON "CLIENT" ("ID\_CLIENT", "CNP")

;

CREATE OR REPLACE TRIGGER "BI\_CLIENT"

before insert on "CLIENT"

for each row

begin

if :NEW."ID\_CLIENT" is null then

select "CLIENT\_SEQ".nextval into :NEW."ID\_CLIENT" from dual;

end if;

end;

/

ALTER TRIGGER "BI\_CLIENT" ENABLE;

**Cont De Credit:**

CREATE TABLE "CONT\_DE\_CREDIT"

( "ID\_CONT" NUMBER,

"VALOARE" NUMBER,

"TIPUL\_DE\_CONT" VARCHAR2(30),

"SUCURSALA" VARCHAR2(30),

CONSTRAINT "CONT\_DE\_CREDIT\_PK" PRIMARY KEY ("ID\_CONT") ENABLE

) ;ALTER TABLE "CONT\_DE\_CREDIT" ADD CONSTRAINT "CONT\_DE\_CREDIT\_FK" FOREIGN KEY ("SUCURSALA")

REFERENCES "SUCURSALA" ("NUME\_UNIC") ENABLE;

CREATE OR REPLACE TRIGGER "BI\_CONT\_DE\_CREDIT"

before insert on "CONT\_DE\_CREDIT"

for each row

begin

if :NEW."ID\_CONT" is null then

select "CONT\_DE\_CREDIT\_SEQ".nextval into :NEW."ID\_CONT" from dual;

end if;

end;

/

ALTER TRIGGER "BI\_CONT\_DE\_CREDIT" ENABLE;

**Cont De Debit:**

CREATE TABLE "CONT\_DE\_DEBIT"

( "ID\_CONT" NUMBER,

"DEPOZIT" NUMBER,

"SUCURSALA" VARCHAR2(30),

"TIPUL\_DE\_CONT" VARCHAR2(30),

CONSTRAINT "CONT\_DE\_DFBIT\_PK" PRIMARY KEY ("ID\_CONT") ENABLE

) ;ALTER TABLE "CONT\_DE\_DEBIT" ADD CONSTRAINT "CONT\_DE\_DFBIT\_FK" FOREIGN KEY ("SUCURSALA")

REFERENCES "SUCURSALA" ("NUME\_UNIC") ENABLE;

CREATE OR REPLACE TRIGGER "BI\_CONT\_DE\_DFBIT"

before insert on "CONT\_DE\_DEBIT"

for each row

begin

if :NEW."ID\_CONT" is null then

select "CONT\_DE\_DFBIT\_SEQ".nextval into :NEW."ID\_CONT" from dual;

end if;

end;

/

ALTER TRIGGER "BI\_CONT\_DE\_DFBIT" ENABLE;

CREATE OR REPLACE TRIGGER "CONT\_DE\_DEBIT\_T1"

BEFORE

insert on "CONT\_DE\_DEBIT"

for each row

begin

:new.depozit := :new.depozit +10;

end;

/

ALTER TRIGGER "CONT\_DE\_DEBIT\_T1" ENABLE;

**Lista Conturi De Credit:**

CREATE TABLE "LISTA\_CONTURI\_DE\_CREDIT"

( "ID\_CLIENT" NUMBER,

"ID\_CONT" NUMBER,

CONSTRAINT "LISTA\_CONTURI\_DE\_CREDIT\_PK" PRIMARY KEY ("ID\_CLIENT", "ID\_CONT") ENABLE

) ;ALTER TABLE "LISTA\_CONTURI\_DE\_CREDIT" ADD CONSTRAINT "LISTA\_CONTURI\_DE\_CREDIT\_FK" FOREIGN KEY ("ID\_CLIENT")

REFERENCES "CLIENT" ("ID\_CLIENT") ENABLE;ALTER TABLE "LISTA\_CONTURI\_DE\_CREDIT" ADD CONSTRAINT "LISTA\_CONTURI\_DE\_CREDIT\_FK2" FOREIGN KEY ("ID\_CONT")

REFERENCES "CONT\_DE\_CREDIT" ("ID\_CONT") ENABLE;

**Lista Conturi de Debit:**

CREATE TABLE "LISTA\_CONTURI\_DE\_DEBIT"

( "ID\_CLIENT" NUMBER,

"ID\_CONT" NUMBER,

CONSTRAINT "LISTA\_CONTURI\_DE\_DEBIT\_PK" PRIMARY KEY ("ID\_CLIENT", "ID\_CONT") ENABLE

) ;ALTER TABLE "LISTA\_CONTURI\_DE\_DEBIT" ADD CONSTRAINT "LISTA\_CONTURI\_DE\_DEBIT\_FK2" FOREIGN KEY ("ID\_CONT")

REFERENCES "CONT\_DE\_DEBIT" ("ID\_CONT") ENABLE;ALTER TABLE "LISTA\_CONTURI\_DE\_DEBIT" ADD CONSTRAINT "LISTA\_CONTURI\_DE\_DEBIT\_FK3" FOREIGN KEY ("ID\_CLIENT")

REFERENCES "CLIENT" ("ID\_CLIENT") ENABLE;

**Persoana:**

CREATE TABLE "PERSOANA"

( "CNP" NUMBER,

"NUME\_SI\_PRENUME" VARCHAR2(50),

"TELEFON" VARCHAR2(15),

CONSTRAINT "PERSOANA\_PK" PRIMARY KEY ("CNP") ENABLE

) ;

CREATE OR REPLACE TRIGGER "BI\_PERSOANA"

before insert on "PERSOANA"

for each row

begin

if :NEW."CNP" is null then

select "PERSOANA\_SEQ".nextval into :NEW."CNP" from dual;

end if;

end;

/

ALTER TRIGGER "BI\_PERSOANA" ENABLE;

**Sucursala:**

CREATE TABLE "SUCURSALA"

( "NUME\_UNIC" VARCHAR2(30),

"ID\_BANCA\_MAMA" NUMBER,

"ADRESA" VARCHAR2(500),

CONSTRAINT "SUCURSALA\_PK" PRIMARY KEY ("NUME\_UNIC") ENABLE

) ;ALTER TABLE "SUCURSALA" ADD CONSTRAINT "SUCURSALA\_FK" FOREIGN KEY ("ID\_BANCA\_MAMA")

REFERENCES "BANCA" ("ID\_BANCA") ENABLE;

CREATE OR REPLACE TRIGGER "BI\_SUCURSALA"

before insert on "SUCURSALA"

for each row

begin

if :NEW."NUME\_UNIC" is null then

select "SUCURSALA\_SEQ".nextval into :NEW."NUME\_UNIC" from dual;

end if;

end;

/

ALTER TRIGGER "BI\_SUCURSALA" ENABLE;

**Queries:**

1. **Lista detalii Angajati**

select "PERSOANA"."NUME\_SI\_PRENUME" as "NUME\_SI\_PRENUME",

"PERSOANA"."TELEFON" as "TELEFON",

"ANGAJAT"."ID\_ANGAJAT" as "ID\_ANGAJAT",

"ANGAJAT"."CNP" as "CNP",

"ANGAJAT"."POZITIE" as "POZITIE",

"ANGAJAT"."SALARIU" as "SALARIU"

from "ANGAJAT" "ANGAJAT",

"PERSOANA" "PERSOANA"

where "ANGAJAT"."CNP"="PERSOANA"."CNP"

1. **Lista detalii Credite**

select "SUCURSALA"."NUME\_UNIC" as "NUME\_UNIC",

"CONT\_DE\_CREDIT"."TIPUL\_DE\_CONT" as "TIPUL\_DE\_CONT",

"CONT\_DE\_CREDIT"."VALOARE" as "VALOARE"

from "CONT\_DE\_CREDIT" "CONT\_DE\_CREDIT",

"SUCURSALA" "SUCURSALA"

where "CONT\_DE\_CREDIT"."SUCURSALA"="SUCURSALA"."NUME\_UNIC"

1. **Lista Angajati care sunt si Clienti**

select "CLIENT"."ID\_CLIENT" as "ID\_CLIENT",

"PERSOANA\_2"."CNP" as "CNP",

"ANGAJAT\_1"."ID\_ANGAJAT" as "ID\_ANGAJAT",

"PERSOANA\_2"."NUME\_SI\_PRENUME" as "NUME\_SI\_PRENUME"

from "ANGAJAT" "ANGAJAT\_1",

"CLIENT" "CLIENT",

"PERSOANA" "PERSOANA\_2"

where "PERSOANA\_2"."CNP"="ANGAJAT\_1"."CNP"

and "PERSOANA\_2"."CNP"="CLIENT"."CNP"

1. **Informatii sucursale**

select "SUCURSALA"."NUME\_UNIC" as "NUME\_UNIC",

"BANCA"."NUME" as "NUME",

"SUCURSALA"."ADRESA" as "ADRESA"

from "SUCURSALA" "SUCURSALA",

"BANCA" "BANCA"

where "BANCA"."ID\_BANCA"="SUCURSALA"."ID\_BANCA\_MAMA"

1. **Lista carduri de credit la BRD Craiova**

select "CARD\_DE\_CREDIT"."ID\_CONT" as "ID\_CONT",

"CARD\_DE\_CREDIT"."ID\_CARD" as "ID\_CARD",

"CARD\_DE\_CREDIT"."NUMELE\_DE\_PE\_CARD" as "NUMELE\_DE\_PE\_CARD",

"CONT\_DE\_CREDIT"."SUCURSALA" as "SUCURSALA",

"CARD\_DE\_CREDIT"."NUMAR\_CARD" as "NUMAR\_CARD"

from "CARD\_DE\_CREDIT" "CARD\_DE\_CREDIT",

"CONT\_DE\_CREDIT" "CONT\_DE\_CREDIT"

where "CARD\_DE\_CREDIT"."ID\_CONT"="CONT\_DE\_CREDIT"."ID\_CONT"

and "CONT\_DE\_CREDIT"."SUCURSALA" = 'BRD Craiova'

1. **Suma totala credite acordate**

select sum(CONT\_DE\_CREDIT.VALOARE) as "VALOARE"

from "LISTA\_CONTURI\_DE\_CREDIT" "LISTA\_CONTURI\_DE\_CREDIT",

"CONT\_DE\_CREDIT" "CONT\_DE\_CREDIT",

"CLIENT" "CLIENT"

where "CLIENT"."ID\_CLIENT"="LISTA\_CONTURI\_DE\_CREDIT"."ID\_CLIENT"

and "CONT\_DE\_CREDIT"."ID\_CONT"="LISTA\_CONTURI\_DE\_CREDIT"."ID\_CONT"

1. **Lista personae cont de debit si cont de credit**

select "CLIENT\_1"."CNP" as "CNP",

"CONT\_DE\_CREDIT"."SUCURSALA" as "SUCURSALA",

"CONT\_DE\_DEBIT"."SUCURSALA" as "SUCURSALA",

"PERSOANA"."NUME\_SI\_PRENUME" as "NUME\_SI\_PRENUME",

"CONT\_DE\_DEBIT"."ID\_CONT" as "ID\_CONT",

"CONT\_DE\_CREDIT"."ID\_CONT" as "ID\_CONT"

from "LISTA\_CONTURI\_DE\_DEBIT" "LISTA\_CONTURI\_DE\_DEBIT",

"LISTA\_CONTURI\_DE\_CREDIT" "LISTA\_CONTURI\_DE\_CREDIT",

"CONT\_DE\_DEBIT" "CONT\_DE\_DEBIT",

"CONT\_DE\_CREDIT" "CONT\_DE\_CREDIT",

"CLIENT" "CLIENT\_1",

"PERSOANA" "PERSOANA"

where "CLIENT\_1"."ID\_CLIENT"="LISTA\_CONTURI\_DE\_CREDIT"."ID\_CLIENT"

and "CLIENT\_1"."ID\_CLIENT"="LISTA\_CONTURI\_DE\_DEBIT"."ID\_CLIENT"

and "CONT\_DE\_CREDIT"."ID\_CONT"="LISTA\_CONTURI\_DE\_CREDIT"."ID\_CONT"

and "LISTA\_CONTURI\_DE\_DEBIT"."ID\_CONT"="CONT\_DE\_DEBIT"."ID\_CONT"

1. **Group by conturi de debit pe fiecare sucursala**

select count(SUCURSALA\_1.NUME\_UNIC) as "NUME\_UNIC",

"CONT\_DE\_DEBIT"."SUCURSALA" as "SUCURSALA"

from "CONT\_DE\_DEBIT" "CONT\_DE\_DEBIT",

"SUCURSALA" "SUCURSALA\_1"

where "CONT\_DE\_DEBIT"."SUCURSALA"="SUCURSALA\_1"."NUME\_UNIC"

group by SUCURSALA\_1.NUME\_UNIC

1. **Info conturi debit cu dobanda 15%**

select "CONT\_DE\_DEBIT"."ID\_CONT" as "ID\_CONT",

"CONT\_DE\_DEBIT"."DEPOZIT" as "DEPOZIT",

"CONT\_DE\_DEBIT"."SUCURSALA" as "SUCURSALA",

"CONT\_DE\_DEBIT"."TIPUL\_DE\_CONT" as "TIPUL\_DE\_CONT"

from "CONT\_DE\_DEBIT" "CONT\_DE\_DEBIT"

where "CONT\_DE\_DEBIT"."TIPUL\_DE\_CONT" ='Dobanda 15%'

1. **Info credite peste 10.000**

select "CONT\_DE\_CREDIT"."ID\_CONT" as "ID\_CONT",

"CONT\_DE\_CREDIT"."VALOARE" as "VALOARE",

"CONT\_DE\_CREDIT"."TIPUL\_DE\_CONT" as "TIPUL\_DE\_CONT",

"CONT\_DE\_CREDIT"."SUCURSALA" as "SUCURSALA"

from "CONT\_DE\_CREDIT" "CONT\_DE\_CREDIT"

where "CONT\_DE\_CREDIT"."VALOARE" >=10000

1. **Informatii credite de nevoi**

select "CONT\_DE\_CREDIT"."ID\_CONT" as "ID\_CONT",

"CONT\_DE\_CREDIT"."VALOARE" as "VALOARE",

"CONT\_DE\_CREDIT"."TIPUL\_DE\_CONT" as "TIPUL\_DE\_CONT",

"CONT\_DE\_CREDIT"."SUCURSALA" as "SUCURSALA"

from "CONT\_DE\_CREDIT" "CONT\_DE\_CREDIT"

where "CONT\_DE\_CREDIT"."TIPUL\_DE\_CONT" ='Credit de nevoi'

1. **Informatii conturi ale caror carduri au overdraft peste 5000**

select "CARD\_DE\_CREDIT"."ID\_CARD" as "ID\_CARD",

"CARD\_DE\_CREDIT"."NUMELE\_DE\_PE\_CARD" as "NUMELE\_DE\_PE\_CARD",

"CARD\_DE\_CREDIT"."NUMAR\_CARD" as "NUMAR\_CARD",

"CARD\_DE\_CREDIT"."OVERDRAFT" as "OVERDRAFT",

"CONT\_DE\_DEBIT"."DEPOZIT" as "DEPOZIT",

"CONT\_DE\_DEBIT"."SUCURSALA" as "SUCURSALA",

"CONT\_DE\_DEBIT"."TIPUL\_DE\_CONT" as "TIPUL\_DE\_CONT"

from "CONT\_DE\_DEBIT" "CONT\_DE\_DEBIT",

"CARD\_DE\_CREDIT" "CARD\_DE\_CREDIT"

where "CONT\_DE\_DEBIT"."ID\_CONT"="CARD\_DE\_CREDIT"."ID\_CONT"

and "CARD\_DE\_CREDIT"."OVERDRAFT" >5000

1. **Count group by dobanda la contul de debit**

select "CONT\_DE\_DEBIT"."TIPUL\_DE\_CONT" as "TIPUL\_DE\_CONT",

count(CONT\_DE\_DEBIT.ID\_CONT) as "ID\_CONT"

from "CONT\_DE\_DEBIT" "CONT\_DE\_DEBIT"

group by CONT\_DE\_DEBIT.TIPUL\_DE\_CONT

1. **Full outer join info angajati**

select "CLIENT"."ID\_CLIENT" as "ID\_CLIENT",

"PERSOANA\_2"."CNP" as "CNP",

"ANGAJAT\_1"."ID\_ANGAJAT" as "ID\_ANGAJAT",

"PERSOANA\_2"."NUME\_SI\_PRENUME" as "NUME\_SI\_PRENUME"

from "ANGAJAT" "ANGAJAT\_1",

"CLIENT" "CLIENT",

"PERSOANA" "PERSOANA\_2"

where "PERSOANA\_2"."CNP"="ANGAJAT\_1"."CNP"(+)

and "PERSOANA\_2"."CNP"="CLIENT"."CNP"(+)

1. **Lista Info Conturi de Credit cu Right Inner Join pe Carduri de Credit**

select "CARD\_DE\_CREDIT"."ID\_CONT" as "ID\_CONT",

"CARD\_DE\_CREDIT"."ID\_CARD" as "ID\_CARD",

"CARD\_DE\_CREDIT"."NUMELE\_DE\_PE\_CARD" as "NUMELE\_DE\_PE\_CARD",

"CONT\_DE\_CREDIT"."SUCURSALA" as "SUCURSALA",

"CARD\_DE\_CREDIT"."NUMAR\_CARD" as "NUMAR\_CARD",

"CONT\_DE\_CREDIT"."ID\_CONT" as "ID\_CONT"

from "CARD\_DE\_CREDIT" "CARD\_DE\_CREDIT",

"CONT\_DE\_CREDIT" "CONT\_DE\_CREDIT"

where "CONT\_DE\_CREDIT"."ID\_CONT"="CARD\_DE\_CREDIT"."ID\_CONT"(+)

and "CONT\_DE\_CREDIT"."SUCURSALA" = 'BRD Craiova'

**Procedures:**

**Procedure 1(Add an employee’s salary to an account):**

create or replace procedure "ADDSALARYTOACCOUNT"  
(employeeid IN NUMBER,  
idcontdebit IN NUMBER)  
is  
    rez number;  
    salary number;  
begin  
SELECT salariu into salary FROM ANGAJAT WHERE id\_angajat = employeeid;  
SELECT depozit into rez FROM cont\_de\_debit WHERE id\_cont=idcontdebit;  
dbms\_output.put\_line('Valoarea depozitului inainte de adaugarea salariului este:'||rez);  
rez := rez+ salary;  
dbms\_output.put\_line('Valoarea depozitului dupa adaugarea salariului este:'||rez);  
UPDATE cont\_de\_debit  
SET depozit = rez  
WHERE id\_cont=idcontdebit;  
end;​

**Procedure 2(Add yearly interest to an account):**

create or replace procedure "ADDYEARLYINTERESTTOACCOUNT"  
(idcont IN NUMBER)  
is  
    rez number;  
begin  
rez := CALCULATEINTERESTAFTERNYEARS(idcont,0);  
dbms\_output.put\_line('Depozitul inainte de adaos: '|| rez);   
rez := CALCULATEINTERESTAFTERNYEARS(idcont,1);  
dbms\_output.put\_line('Depozitul dupa adaos: '|| rez);  
UPDATE cont\_de\_debit  
SET depozit = rez  
WHERE id\_cont=idcont;  
end;​

**Functions:**

**Function 1(Calculate interest sum after N years):**

create or replace function "CALCULATEINTERESTAFTERNYEARS"  
      
(idcont in NUMBER,  
nryears in NUMBER)  
return NUMBER  
is  
    sum\_c number;  
    tip VARCHAR2(30);  
    rez number;  
begin  
  
  
SELECT depozit, tipul\_de\_cont into sum\_c, tip FROM CONT\_DE\_DEBIT WHERE id\_cont = idcont;  
  
  
IF tip = 'Dobanda 10%' THEN  
   rez := sum\_c + nryears \* 0.1 \* sum\_c;  
ELSE IF tip = 'Dobanda 15%' THEN  
    rez := sum\_c + nryears \* 0.15 \* sum\_c;  
ELSE IF tip = 'Dobanda 5%' THEN  
    rez := sum\_c + nryears \* 0.05 \* sum\_c;  
END IF;  
END IF;  
END IF;  
  
RETURN rez;  
end;

​**Function 2(Calculate yearly salary of an employee):**

create or replace function "CALCULATEYEARLYSALLARY"  
(employeeid in NUMBER)  
return NUMBER  
is  
    rez number;  
    salary number;  
begin  
SELECT salariu into salary FROM ANGAJAT WHERE id\_angajat = employeeid;  
  
rez:= salary \*12;  
  
RETURN rez;  
end;​

**Trigger(Gives a bonus starting amount to a new Debit account):**

create or replace trigger "CONT\_DE\_DEBIT\_T1"  
BEFORE  
insert on "CONT\_DE\_DEBIT"  
for each row  
begin  
:new.depozit := :new.depozit +10;  
end;​