

Proiect baze de date:

„Proiectarea și implementarea unei baze de date în domeniul economic, utilizând Oracle Databases”

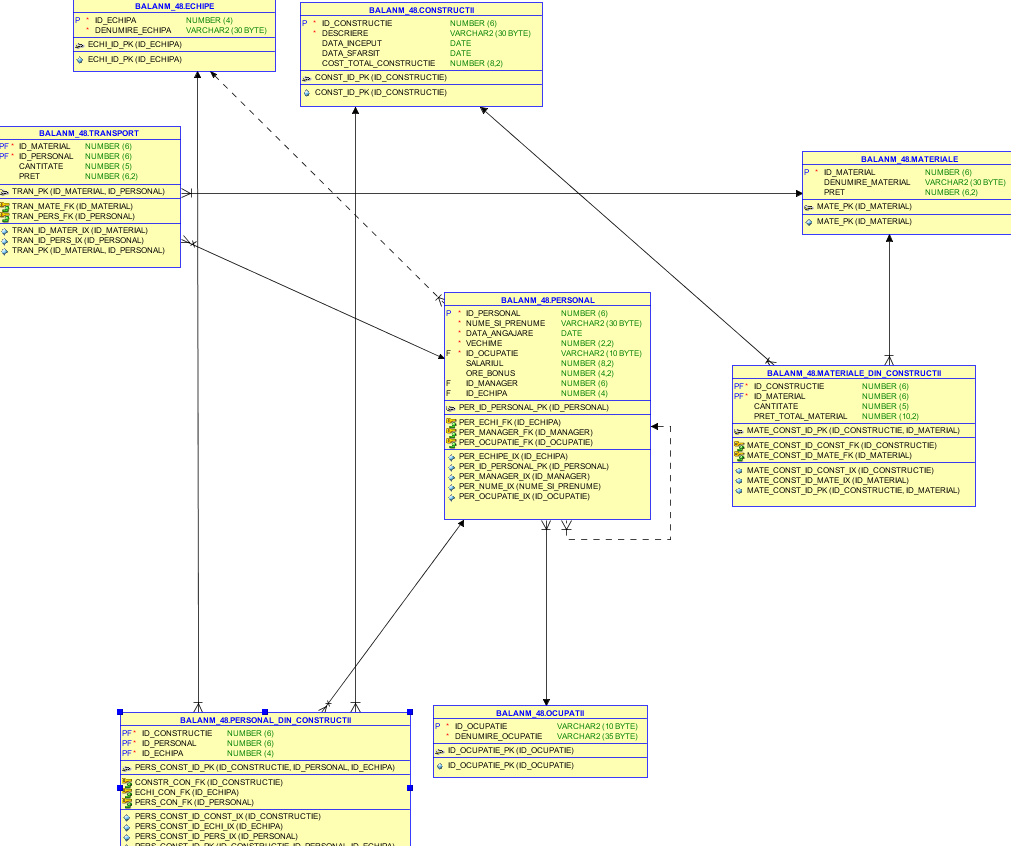
Scris de studentul: Bălan Mircea-George

Seria C, Grupa 1048

Sub îndrumarea profesorului: COPACEANU ANDREEA-MARIA

**DESCRIEREA TEMEI**

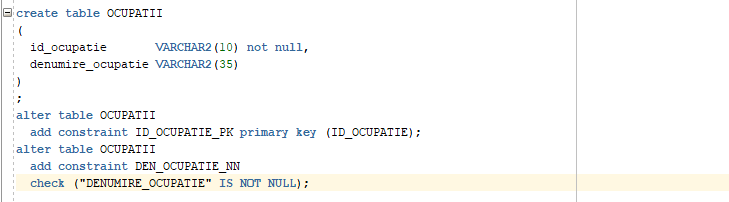
* Am ales să creez o baza de date pentru o firmă de construcții. Această baza cuprinde 8 tabele, și anume: ECHIPE, FUNCTII, PERSONAL, CONSTRUCTII, PERSONAL\_DIN\_CONSTRUCTII, MATERIALE, MATERIALE\_DIN\_CONSTRUCTII, TRANSPORT. Aceste tabele conțin date despre personalul angajat, proiectele și echipele în care fiecare angajat ia parte, materialele folosite în fiecare construcție împreună prețul acestora, șoferii care aduc anumite transporturi de materiale si multe altele
* Asocierile dintre entitățile tabelei se stabilesc în funcție de modul în care se desfășoară activitatea descrisă.
* Fiecare angajat are o funcție. 1-1
* Fiecare echipă are cel puțin un angajat, însă nu și contrariul, un angajat nu are neapărat nevoie să fie într-o echipă. 1-M
* O construcție poate avea mai multe echipe lucrătoare și o echipa poate lucra la mai multe construcții. M-M
* O construcție poate dispune de mai multe materiale și un material poate fi folosit pentru mai multe construcții M-M
* O funcție poate avea mai mulți angajați, însă un angajat poate avea doar o funcție 1-M



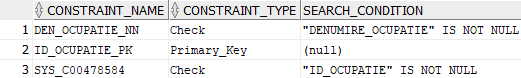
* **Crearea si popularea tabelelor**

1.Tabela Ocupații

Această tabelă cuprinde totalitatea ocupațiilor deținute de personalul ce lucrează la firma de construcții.



Restricțiile:



Popularea:

insert into ocupatii values('PM','Project Manager');

insert into ocupatii values('SIE','Sef Inginer Electronist');

insert into ocupatii values('A','Arhitect');

insert into ocupatii values('SF','Sef Finisaje');

insert into ocupatii values('SS','Sef Structurist');

insert into ocupatii values('FI','Fierar');

insert into ocupatii values('D','Dulgher');

insert into ocupatii values('FA','Faiantier');

insert into ocupatii values('IM','Instalator Machete');

insert into ocupatii values('IG','Instalator Gipscarton');

insert into ocupatii values('B','Betonist');

insert into ocupatii values('SI','Sef Instalatii');

insert into ocupatii values('L','Lacatus');

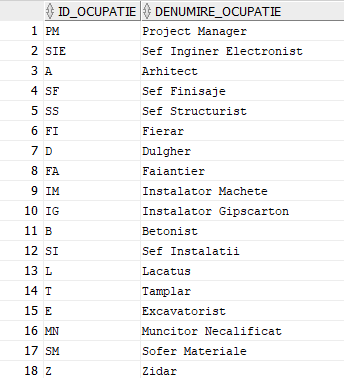
insert into ocupatii values('T','Tamplar');

insert into ocupatii values('E','Excavatorist');

insert into ocupatii values('MN','Muncitor Necalificat');

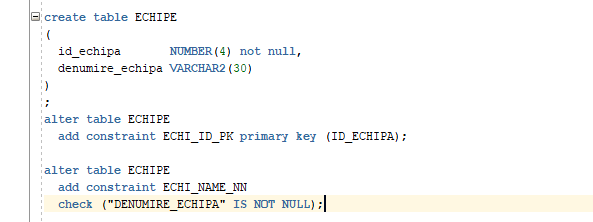
insert into ocupatii values('SM','Sofer Materiale');

insert into ocupatii values('Z','Zidar');

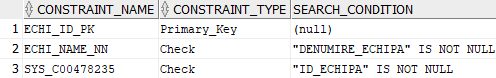


2.Tabela Echipe

Această tabelă cuprinde echipele formate din personal care lucrează la diverse construcții.



Restricțiile



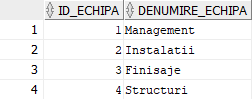
Popularea

insert into echipe values(1,'Management');

insert into echipe values(2,'Instalatii');

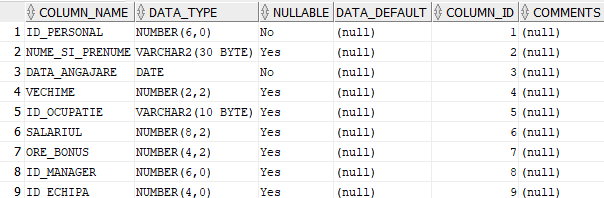
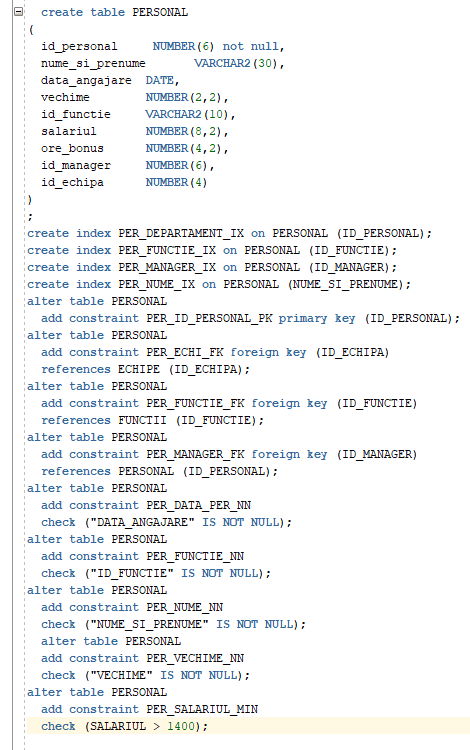
insert into echipe values(3,'Finisaje');

insert into echipe values(4,'Structuri');

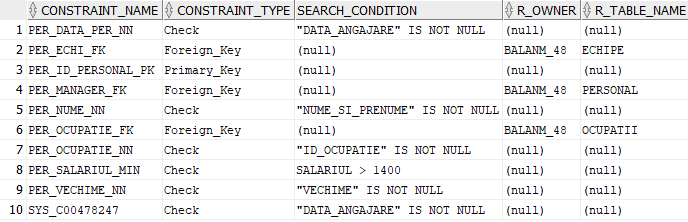


3.Tabela Personal

Această tabelă cuprinde informații despre angajații firmei, precum ocupațiile acestora, echipele în care se află și detalii despre data angajării și salariu.



Restricții



Popularea

insert into personal(id\_personal,nume\_si\_prenume,data\_angajare,id\_ocupatie,

salariul,ore\_bonus,id\_manager,id\_echipa)values(1,'Maiku Rosmin',

to\_date('23-04-2020','dd-mm-yyyy'),'PM',2000,5,(null),1);

insert into personal(id\_personal,nume\_si\_prenume,data\_angajare,id\_ocupatie,

salariul,ore\_bonus,id\_manager,id\_echipa)values(2,'Caraiman Traian',

to\_date('23-04-2020','dd-mm-yyyy'),'A',8000,1,1,1);

insert into personal(id\_personal,nume\_si\_prenume,data\_angajare,id\_ocupatie,

salariul,ore\_bonus,id\_manager,id\_echipa)values(3,'Zidaru Andrei',

to\_date('23-04-2020','dd-mm-yyyy'),'A',8500,2,(null),4);

insert into personal(id\_personal,nume\_si\_prenume,data\_angajare,id\_ocupatie,

salariul,ore\_bonus,id\_manager,id\_echipa)values(4,'Dulgheru Roman',

to\_date('23-04-2020','dd-mm-yyyy'),'SF',7600,0,(null),3);

insert into personal(id\_personal,nume\_si\_prenume,data\_angajare,id\_ocupatie,

salariul,ore\_bonus,id\_manager,id\_echipa)values(5,'Balaiban Roxan',

to\_date('26-08-2019','dd-mm-yyyy'),'SS',7000,10,(null),4);

insert into personal(id\_personal,nume\_si\_prenume,data\_angajare,id\_ocupatie,

salariul,ore\_bonus,id\_manager,id\_echipa)values(6,'Vasili Cosmin',

to\_date('08-10-2019','dd-mm-yyyy'),'FI',2500,5,4,3);

insert into personal(id\_personal,nume\_si\_prenume,data\_angajare,id\_ocupatie,

salariul,ore\_bonus,id\_manager,id\_echipa)values(7,'Tanase Emil',

to\_date('01-12-2019','dd-mm-yyyy'),'D',2700,5,3,4);

insert into personal(id\_personal,nume\_si\_prenume,data\_angajare,id\_ocupatie,

salariul,ore\_bonus,id\_manager,id\_echipa)values(8,'Balan Marius',

to\_date('20-01-2020','dd-mm-yyyy'),'SI',6600,1,(null),2);

insert into personal(id\_personal,nume\_si\_prenume,data\_angajare,id\_ocupatie,

salariul,ore\_bonus,id\_manager,id\_echipa)values(9,'Badin Ramon',

to\_date('11-11-2019','dd-mm-yyyy'),'FA',3000,0,8,2);

insert into personal(id\_personal,nume\_si\_prenume,data\_angajare,id\_ocupatie,

salariul,ore\_bonus,id\_manager,id\_echipa)values(10,'Nita Mihai',

to\_date('29-01-2018','dd-mm-yyyy'),'SI',6000,0,1,1);

insert into personal(id\_personal,nume\_si\_prenume,data\_angajare,id\_ocupatie,

salariul,ore\_bonus,id\_manager,id\_echipa)values(11,'Katalin Talent',

to\_date('17-06-2021','dd-mm-yyyy'),'IM',3900,10,8,2);

insert into personal(id\_personal,nume\_si\_prenume,data\_angajare,id\_ocupatie,

salariul,ore\_bonus,id\_manager,id\_echipa)values(12,'Silviu Faiar',

to\_date('06-09-2021','dd-mm-yyyy'),'IG',3800,7,8,2);

insert into personal(id\_personal,nume\_si\_prenume,data\_angajare,id\_ocupatie,

salariul,ore\_bonus,id\_manager,id\_echipa)values(13,'Rotaru Iulian',

to\_date('20-12-2017','dd-mm-yyyy'),'B',4100,2,3,4);

insert into personal(id\_personal,nume\_si\_prenume,data\_angajare,id\_ocupatie,

salariul,ore\_bonus,id\_manager,id\_echipa)values(14,'Tutuianu Eugen',

to\_date('12-10-2018','dd-mm-yyyy'),'L',2000,10,4,3);

insert into personal(id\_personal,nume\_si\_prenume,data\_angajare,id\_ocupatie,

salariul,ore\_bonus,id\_manager,id\_echipa)values(15,'Iuliescu Razvan',

to\_date('26-05-2019','dd-mm-yyyy'),'E',5000,12,3,4);

insert into personal(id\_personal,nume\_si\_prenume,data\_angajare,id\_ocupatie,

salariul,ore\_bonus,id\_manager,id\_echipa)values(16,'Mainea Florin',

to\_date('15-10-2020','dd-mm-yyyy'),'MN',1450,2,3,4);

insert into personal(id\_personal,nume\_si\_prenume,data\_angajare,id\_ocupatie,

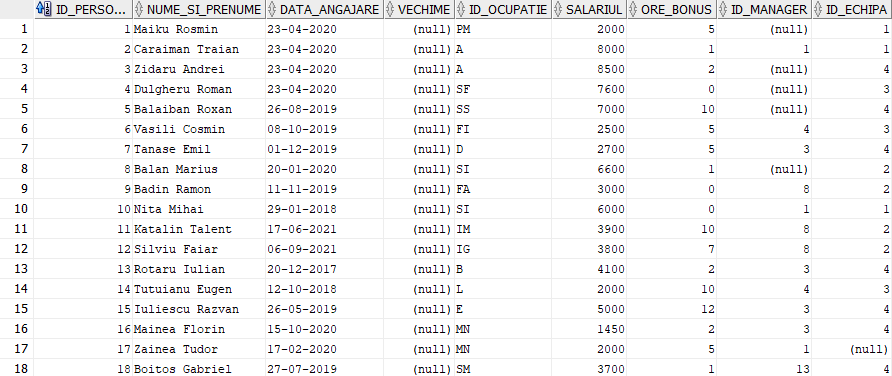
salariul,ore\_bonus,id\_manager)values(17,'Zainea Tudor',

to\_date('17-02-2020','dd-mm-yyyy'),'MN',2000,5,1);

insert into personal(id\_personal,nume\_si\_prenume,data\_angajare,id\_ocupatie,

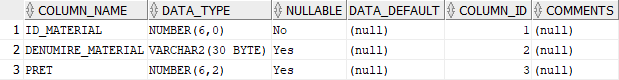
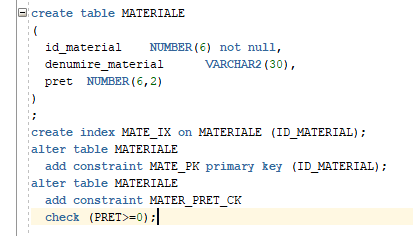
salariul,ore\_bonus,id\_manager,id\_echipa)values(18,'Boitos Gabriel',

to\_date('27-07-2019','dd-mm-yyyy'),'SM',3700,1,13,4);

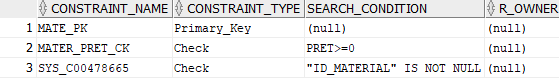


4.Tabela Materiale

Această tabelă cuprinde informații despre materialele folosite în proiectele de construcții, pe baza cărora se va calcula costul total al construcțiilor din tabela construcții.



Restricții



Populare

insert into materiale values(1,'Scandura', 10);

insert into materiale values(2,'Beton simplu', 40);

insert into materiale values(3,'Beton armat', 100);

insert into materiale values(4,'Pietris', 23);

insert into materiale values(5,'Placa polistiren', 46);

insert into materiale values(6,'Caramida', 70);

insert into materiale values(7,'Mortar', 44);

insert into materiale values(8,'Buiandruga', 32);

insert into materiale values(9,'Folie de plastic', 120);

insert into materiale values(10,'Distantier', 200);

insert into materiale values(11,'Placa confata', 300);

insert into materiale values(12,'Grinda din lemn', 100);

insert into materiale values(13,'Popi metalici', 110);

insert into materiale values(14,'Caprior', 97);

insert into materiale values(15,'Barna', 89);

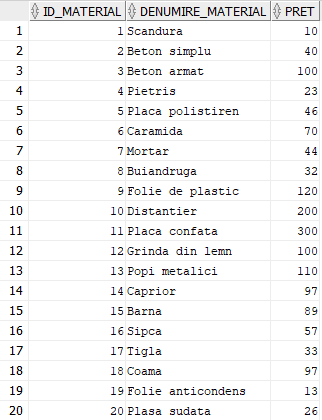
insert into materiale values(16,'Sipca', 57);

insert into materiale values(17,'Tigla', 33);

insert into materiale values(18,'Coama', 97);

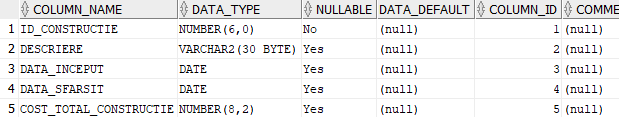
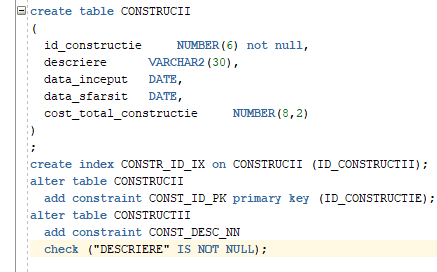
insert into materiale values(19,'Folie anticondens', 13);

insert into materiale values(20,'Plasa sudata', 26);

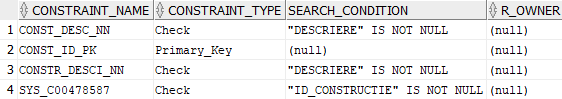


5.Tabela Construcții

Această tabelă cuprinde informații despre proiectele la care lucrează firma: data la care a început proiectul, data la care s-a sfârșit(dacă este cazul) și costul total al proiectului până atunci.



Restricții



Populare

insert into constructii values(1,'Casa plaja Mamaia',

to\_date('17-02-2018','dd-mm-yyyy'),to\_date('17-02-2019','dd-mm-yyyy'),0);

insert into constructii values(2,'Restaurant Braila',

to\_date('19-03-2019','dd-mm-yyyy'),to\_date('23-06-2020','dd-mm-yyyy'),0);

insert into constructii values(3,'Hotel Vama',

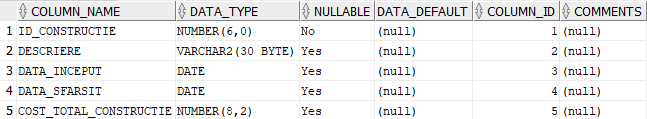
to\_date('17-02-2020','dd-mm-yyyy'),(null),0);

insert into constructii values(4,'Statie de pompieri Constanta',

to\_date('30-10-2017','dd-mm-yyyy'),to\_date('24-12-2021','dd-mm-yyyy'),0);

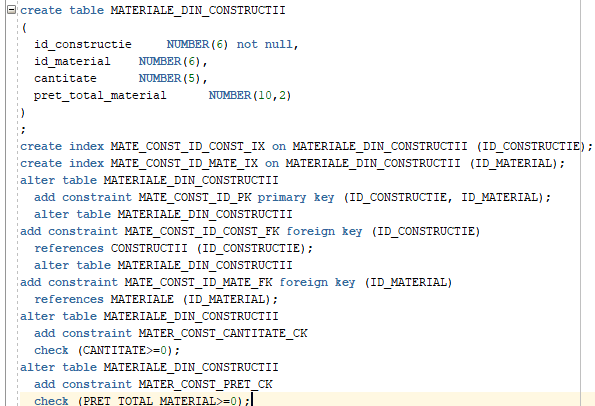
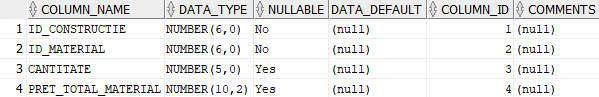
insert into constructii values(5,'Adapost animale',

to\_date('07-01-2022','dd-mm-yyyy'),(null),0);

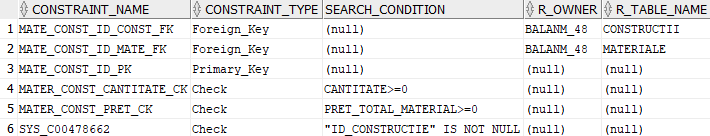


6.Tabela Materiale din constructii

Această tabelă cuprinde informații despre cantitatea și costul materialelor folosite într-un anumit proiect.



Restricții



* **Comenzi LDD(Create/Alter/Drop)**
* **Modificarea Structurii Tabelelor(Alter)**
* **Ștergerea tabelelor(Drop/Truncate)**

Primul pas în crearea unei baze de date este de a construi tabele, pentru a face aceasta, în primul rând avem nevoie de COMANDA „CREATE TABLE”, urmată de numele tabelei pe care dorim să o creăm și apoi conținutul acesteia între două paranteze rotunde. Vom crea tabela „ECHIPE” precum urmează.

create table ECHIPE

(

id\_echipa NUMBER(4) not null,

denumire\_echipa VARCHAR2(30)

)

;



În plus, se pot folosi și restricții de integritate, care pot fi fie in-line, fie out-of-line.

Pentru tabela echipe deja am folosit restricția „NOT NULL” prin metoda in-line. Acum vom transforma coloana „id\_echipa” într-o cheie primară și vom adăuga restricția „NOT NULL” și coloanei „denumire\_echipa” prin metoda out-of-line. Pentru a face aceasta, și anume, să alterăm tabela, după ce deja am creat-o, vom avea nevoie de comanda „ALTER TABLE” și mai apoi de comanda „ADD” pentru a specifica faptul că dorim să adăugam o restricție.

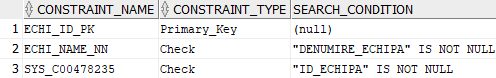
alter table ECHIPE

add constraint ECHI\_ID\_PK primary key (ID\_ECHIPA);

alter table ECHIPE

add constraint ECHI\_NAME\_NN

check ("DENUMIRE\_ECHIPA" IS NOT NULL);



În continuare voi crea o tabela „PERSONAL” cu restricțiile „NOT NULL”, „PRIMARY-KEY”, „FOREIGN-KEY” și „CHECK” precizate on-line pentru a exemplifica utilitățile restricțiilor.

create table PERSONAL

(

id\_personal NUMBER(6) constraint PER\_ID\_PERSONAL\_PK primary key,

nume\_si\_prenume VARCHAR2(30),

data\_angajare DATE not null,

vechime NUMBER(2,2),

id\_functie VARCHAR2(10),

salariul NUMBER(8,2) constraint PER\_SALARIUL\_MIN check (SALARIUL > 1400),

ore\_bonus NUMBER(4,2),

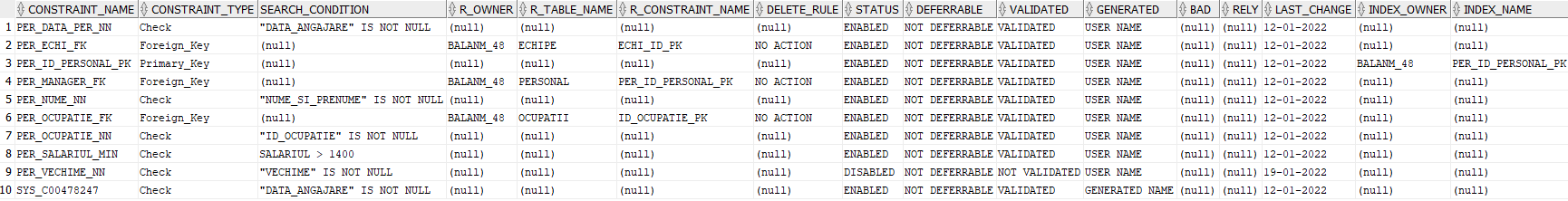
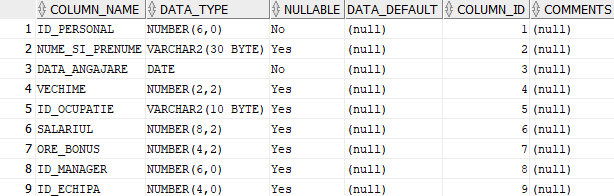
id\_manager NUMBER(6),

id\_echipa NUMBER(4),

constraint PER\_ECHI\_FK FOREIGN KEY(ID\_ECHIPA) references ECHIPE (ID\_ECHIPA)

)

;



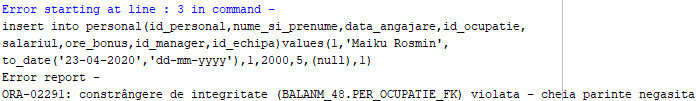
Dorim sa dezactivăm una dintre restricții, si anume, restricția ”NOT NULL” pentru vechime. Coloana, o vom umple mai târziu pe baza alteia, și anume, coloana data\_angajare, însă pentru a face aceasta avem nevoie să populăm coloana data\_angajare și să lăsăm goala coloana vechime. Altfel va da eroare datorita restricției. Prin urmare, vom dezactiva restricția prin comanda ”DISABLE CONSTRAINT”. Astfel putem să reactivăm restricția în caz că o să mai avem nevoie de ea prin comanda ”ENABLE”. Altfel, putem folosi ”DROP” pentru a șterge definitiv restricția.

ALTER TABLE personal DISABLE CONSTRAINT PER\_VECHIME\_NN;

ALTER TABLE personal ENABLE CONSTRAINT PER\_VECHIME\_NN;

ALTER TABLE personal DROP CONSTRAINT PER\_VECHIME\_NN;

De reținut faptul că restricția “FOREIGN KEY” poate fi folosită doar dacă deja există un alt tabel ce conține acea coloana având restricția „PRIMARY KEY”.



Foreign KEY out-of-line:

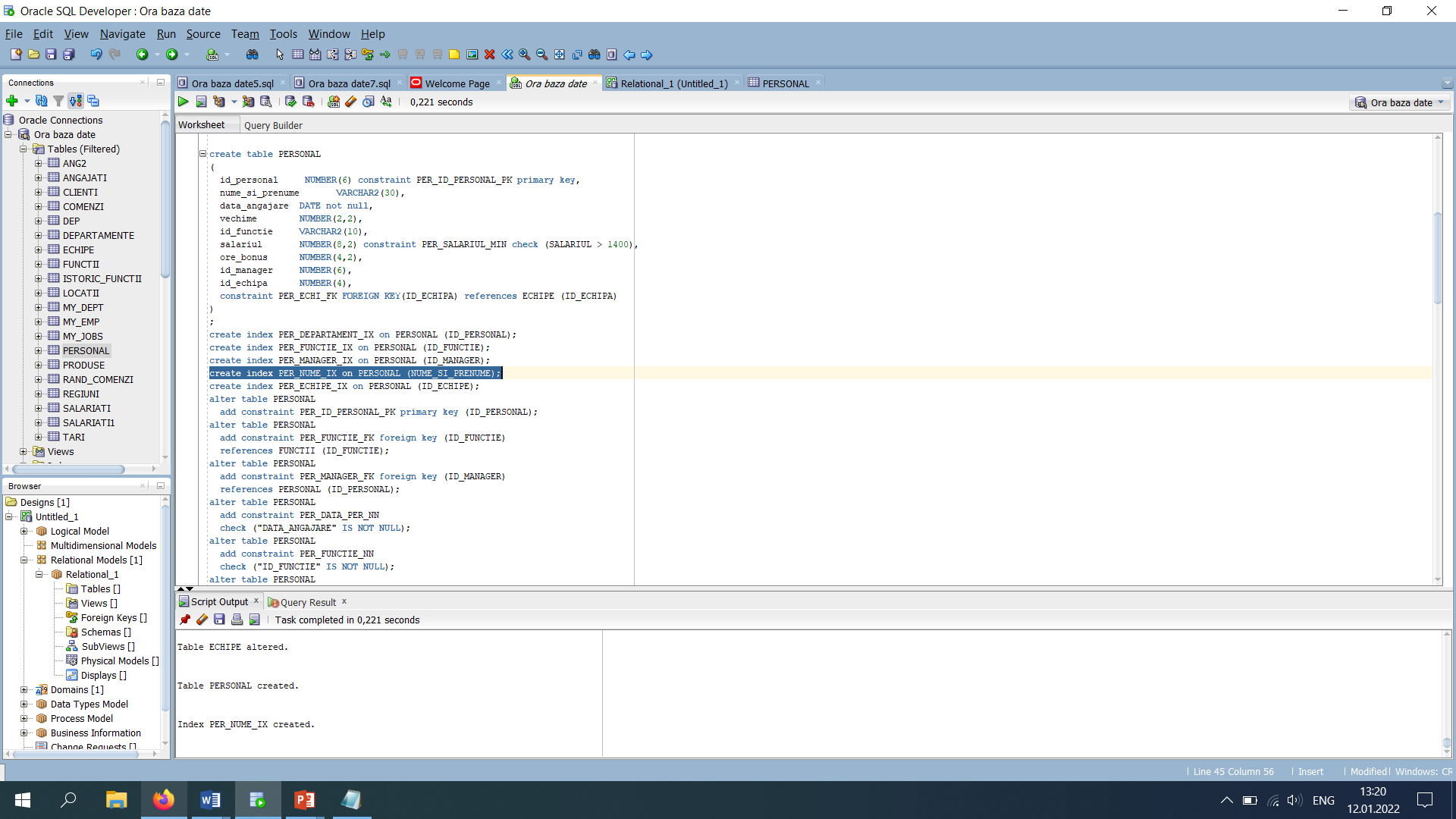
alter table PERSONAL

add constraint PER\_MANAGER\_FK foreign key (ID\_MANAGER)

references PERSONAL (ID\_PERSONAL);

De asemenea, pe lângă comanda „CREATE TABLE” se poate folosi și comanda „CREATE INDEX”. Indecșii ne permit să accesam datele din bază mult mai rapid astfel încât select-urile și proiecțiile să aibă o durata mai mica de execuție. Această comandă are nevoie de un indicator(PER\_NUME\_IX), pentru a putea fi identifică ușor, iar apoi adăugam „ON” înainte de tabela și coloana căreia dorim să îi adăugam un index.

create index PER\_NUME\_IX on PERSONAL (NUME\_SI\_PRENUME);



Uneori facem anumite greșeli pe care mai apoi trebuie să le rectificăm sau poate pur și simplu ne răzgândim. Spre exemplu, doresc să schimb denumirea coloanei id\_functie în id\_ocupatie, aceasta e posibilă cu ajutorul comenzii „RENAME COLUMN” (denumire coloana) TO (noua denumire)”

ALTER TABLE PERSONAL RENAME COLUMN id\_functie TO id\_ocupatie;

Se observă faptul că trebuie specificat ce anume este ceea ce dorim să modificam: fie o coloană, fie o restricție (constraint).

ALTER TABLE PERSONAL RENAME CONSTRAINT PER\_FUNCTIE\_FK TO PER\_OCUPATIE\_FK;

Pentru tabele mai există o posibilitate, și anume, folosim direct „RENAME”, „ALTER TABLE” înțelegându-se de la sine.

RENAME ANGAJATI TO PERSONAL;

Odată ce am creat tabelele este nevoie să le și populăm. Aceasta se face prin comanda “INSERT INTO” și există două posibilități prin care putem duce comanda la bun sfârșit. Fie scriem numele tabelei urmată de colanele unde dorim să adăugăm valori(1), fie scriem doar numele tabelei daca dorim să adăugăm valori tuturor coloanelor(2). După numele tabelei (și a coloanelor) trebuie specificate valorile, dar nu înainte de a scrie “VALUES”. În continuare vom popula tabelele prin ambele metode:

OCUPATII:

(1) insert into ocupatii(id\_ocupatie,denumire\_ocupatie)values('PM','Project Manager');

(2) insert into ocupatii values('SIE','Sef Inginer Electronist');

insert into ocupatii values('A','Arhitect');

insert into ocupatii values('SF','Sef Finisaje');

insert into ocupatii values('SS','Sef Structurist');

insert into ocupatii values('FI','Fierar');

insert into ocupatii values('D','Dulgher');

insert into ocupatii values('FA','Faiantier');

insert into ocupatii values('IM','Instalator Machete');

insert into ocupatii values('IG','Instalator Gipscarton');

insert into ocupatii values('B','Betonist');

insert into ocupatii values('SI','Sef Instalatii');

insert into ocupatii values('L','Lacatus');

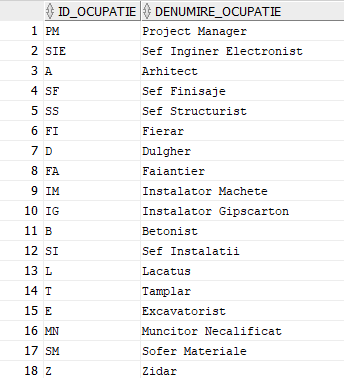
insert into ocupatii values('T','Tamplar');

insert into ocupatii values('E','Excavatorist');

insert into ocupatii values('MN','Muncitor Necalificat');

insert into ocupatii values('SM','Sofer Materiale');

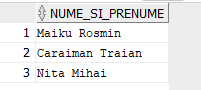
insert into ocupatii values('Z','Zidar');



Odată creată o tabelă, aceasta poate fi folosită drept bază pentru crearea unei alte tabele. Trebuie avut în vedere, însă, faptul că repetarea informației într-o bază de date poate fi redundantă. Putem face o nouă tabelă tot cu ajutorul „CREATE TABLE” însă pentru a baza noua tabela pe o alta mai veche folosim comenzile „AS” și „SELECT”, chiar și „WHERE” daca este o anumita condiție ce o dorim satisfăcută pentru intrările din noul tabel. Este important de notat faptul că, datorita acestui mod de a crea o tabela, adică cu ajutorul comenzii „SELECT”, aceasta se va și popula cu intrările celei de bază care îndeplinesc condițiile impuse de „WHERE”. Vom crea tabela „Manageri” formată din personalul din echipa Management.

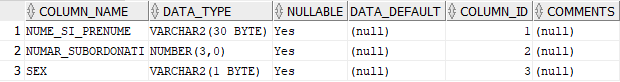
create table Manageri as select p.nume\_si\_prenume from personal p, echipe e

where p.id\_echipa=e.id\_echipa and denumire\_echipa like '%Management%';



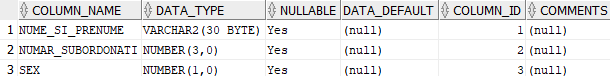
Să adăugăm tabelei Manageri coloanele “numar\_subordonati” și “sex”.

alter table Manageri add (numar\_subordonati number(3), sex varchar2(1));



Sa modificam tipul de data al coloanei “gen”, apoi sa o dezactivam si in cele din urma sa ștergem coloanele inactive.

alter table Manageri modify(sex number (1));



alter table Manageri set unused column sex;

alter table Manageri drop unused columns;



Comanda “DROP TABLE” permite ștergerea unei tabele, pe când comanda “FLASHBACK TABLE” ne permite sa recuperam tabela ștearsă.

DROP TABLE Manageri;

FLASHBACK TABLE Manageri TO BEFORE DROP;

Însă comanda “TRUNCATE TABLE” șterge înregistrările tabelei si eliberează și spațiul alocat acestora.

TRUNCATE TABLE Manageri;

* **Comenzi LMD(Insert/Update/Delete/Merge/Select)**

Inserarea înregistrărilor se poate face prin trei modalități.

Fie prin precizarea explicită a valorilor, însă trebuie precizate valori pentru toate coloanele tabelei.

insert into personal values(19,'Grecu Fanel', to\_date('27-02-2021','dd-mm-yyyy'),'MN',2100,6,1,1);

Fie prin adăugarea înregistrărilor pe baza valorilor din alte tabele:

INSERT INTO Manageri Select nume from personal where id\_echipa=1;

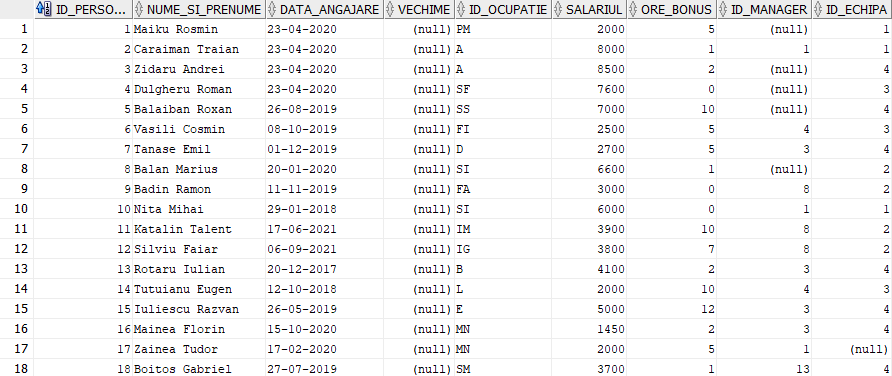
Fie prin utilizarea variabilelor de substituție pentru adăugarea înregistrărilor pe baza valorilor introduse de utilizator de la tastatură:

insert into personal(id\_personal,nume\_si\_prenume,data\_angajare,id\_ocupatie,salariul,

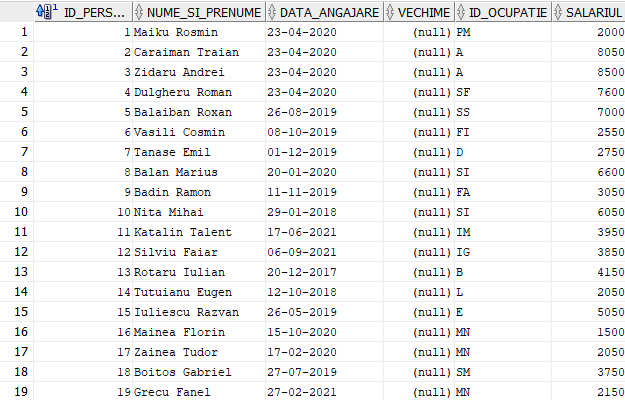
ore\_bonus,id\_manager,id\_echipa)values(19,'Grecu Fanel', to\_date('27-02-2021','dd-mm-yyyy'),'MN',2100,6,1,1);

Cât despre comanda „UPDATE”, în principiu este nevoie și de comenzile „SET” pentru a specifica cum dorim să schimbăm și „WHERE” pentru a specifica ce anume dorim să schimbăm. Astfel, vom crește salariul cu 50 tuturor angajaților care sunt subordonați cuiva.

INAINTE:



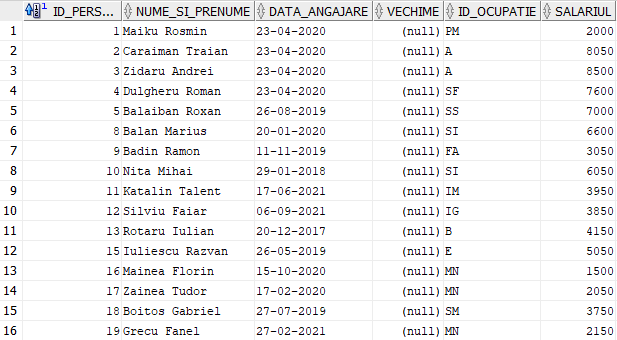
DUPĂ:



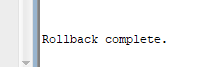
Ștergerea unor intrări se face foarte ușor cu ajutorul comenzii „DELETE”. În caz că nu dorim această schimbare putem folosi comanda „ROLLBACK” pentru a ne întoarce înainte de „DELETE”. În caz contrar, când vrem să salvam schimbările făcute folosim comanda „COMMIT”.

DELETE FROM PERSONAL

WHERE data\_angajare<TO\_DATE('01-01-2020','DD-MM-YYYY') and salariul<3000;



ROLLBACK;

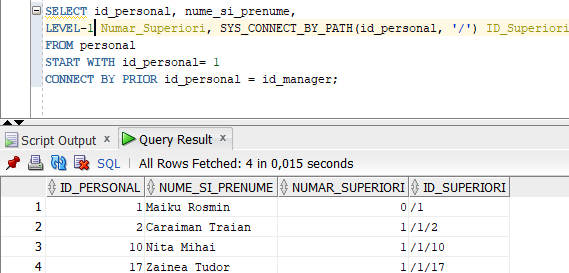


COMMIT;

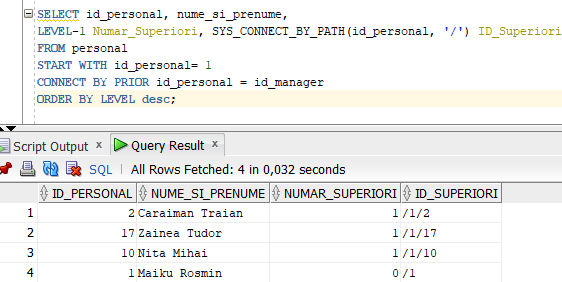
* **Structuri Ierarhice**

Parcurgerea acestor structuri se poate face prin trei modalități:

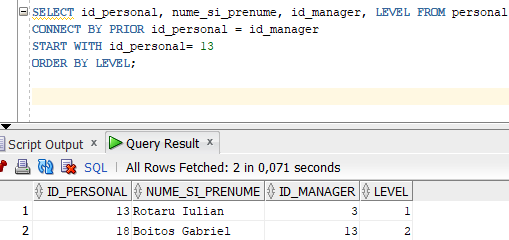
I.TOP-BOTTOM unde specificam care este „topul”



II.BOTTOM UP care este opusul lui TOP-BOTTOM



III. Parcurgerea arborelui începând cu alt nivel decât primul sau ultimul



* **Decode and Case**

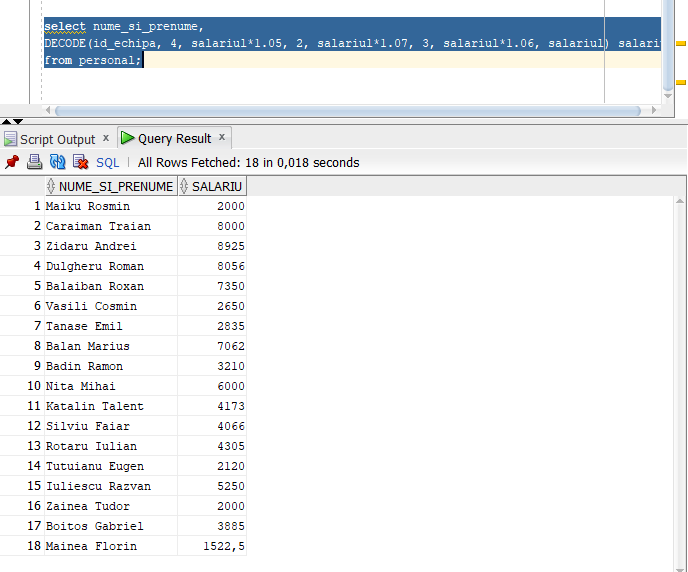
Să se afișeze numele personalului și să se mărească salariile în funcție de echipă astfel:

- Daca id\_echipa= 4 atunci creșterea sa fie de 5%

- Daca id\_echipa= 2 atunci creșterea sa fie de 7%

- Daca id\_echipa= 3 atunci creșterea sa fie de 6%

- In rest sa nu se aplice creșteri salariale.



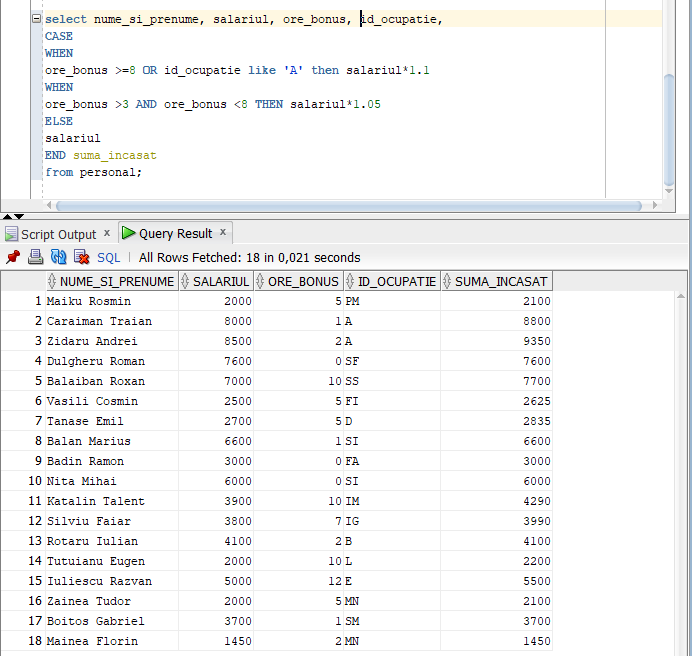
CASE ne permite să fim mult mai detaliați în condiții în comparație cu DECODE

Să se afișeze numele, salariul și funcția personalului, precum și suma de încasat astfel:

- daca ore\_bonus>8 suma\_incasat = salariul\*10% sau id\_ocupatie=’A’

- daca ore\_bonus>3 dar <8 atunci suma\_incasat = salariul\*5%

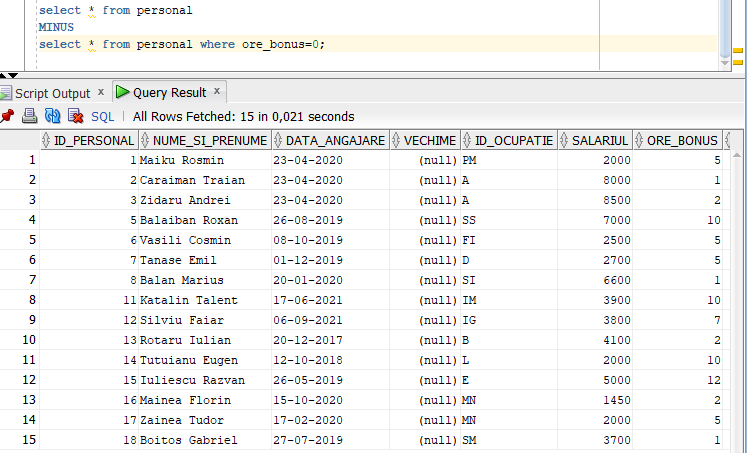
- pentru celelalte funcții suma\_incasat = salariul



* **UNION, INTERSECT, MINUS**

Operatorul MINUS e alcătuit din doua selecturi și ne permite să extragem ce se află în prima interogare și care nu se afla în a doua.

Să se afișeze numărul orelor bonus pentru fiecare angajat și folosind operatorul MINUS să se elimine angajații care nu au stat peste program.



UNION reunește toate interogările din selecturi

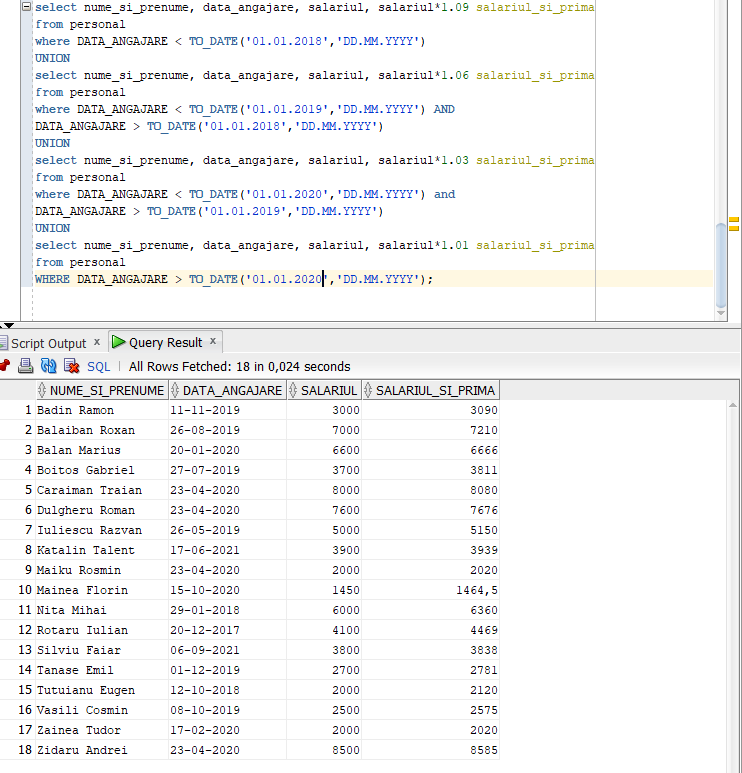
Să se acorde prime fiecărui angajat în funcție de vechime astfel:

- Pentru cei angajați înainte de 1 ianuarie 2018 prima=9%\*salariu

- Pentru cei angajați înainte de 1 ianuarie 2019 prima=6%\*salariu

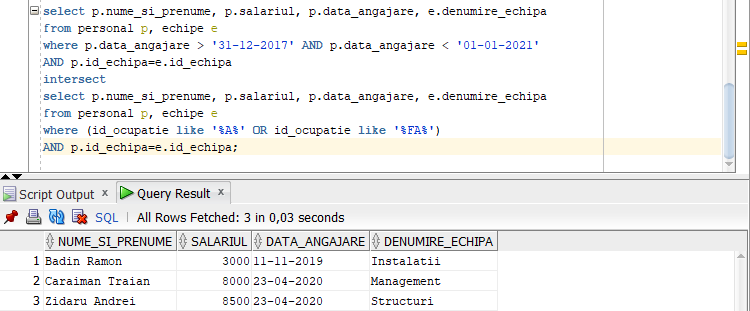
- Pentru cei angajați înainte de 1 ianuarie 2020 prima=3%\*salariu

- Pentru ceilalți angajați prima = 1%\*salariu



INTERSECT returnează doar ce se găsește atât în prima interogare, cât și în a doua

Să se afișeze numele și prenumele, salariul, data angajării, denumirea echipei pentru angajații care au funcția Arhitect sau Faiantier și au fost angajați între 2018 și 2020.



* **TABELE VIRTUALE(CREATE/REPLACE VIEW)**

View-urile ne permit să accesam mult mai ușor selecturile ce sunt folosite des. De asemenea datorită comenzii “WITH READ ONLY” putem proteja baza de orice schimbări neautorizate. La final putem șterge view-urile cu ajutorul comenzii “DROP VIEW”.

CREAREA UNUI VIEW (READ ONLY OPTIONAL)

CREATE OR REPLACE VIEW prima\_salariu

AS SELECT id\_personal, ore\_bonus, salariul\*(0.1\*ore\_bonus) salariu FROM personal;

(WITH READ ONLY;)

Updatarea unui view(doar daca nu este read-only)

UPDATE prima\_salariu

SET salariu = salariu+100;

Crearea unui trigger

CREATE OR REPLACE TRIGGER t\_prima\_salariu

INSTEAD OF UPDATE ON prima\_salariu

BEGIN

UPDATE personal SET salariul = salariul+100;

end;

Ștergerea unui view

DROP VIEW prima\_salariu;

* **SINONIME**

Acestea sunt nume alternative pentru referirea obiectelor unei baze de date. Însă crearea acestora necesită anumite privilegii.

CREATE SYNONYM mdc FOR material\_din\_constructii;

Putem folosi și comanda “SELECT […] FROM USER\_SYNONIMS” pentru a vedea toate sinonimele curente.

SELECT \* FROM USER\_SYNONYMS;

DROP SYNONYM mdc;

* **PARTITII SI CLUSTERE**

Acestea aranjează informația din tabele, pe mici grupulețe egale, în așa fel încât să poată fi accesată mult mai ușor.

CREATE TABLE accidente(data\_accident DATE, nume\_accidentat VARCHAR2(50), zile\_spitalizat NUMBER(2))

PARTITION BY RANGE(zile\_spitalizat)

(PARTITION P1 VALUES LESS THAN 1, PARTITION P2 VALUES MORE THAN 30);