

Project SGBD Oracle

Gestionarea unui service auto

☺ Zilnic avem de-a face cu date și informații pe care dorim să le gestionăm și să le organizăm cât mai ușor posibil.

☺ Bazele de date reprezintă colecții de informații logic organizate, astfel încât să ne ușureze și să ne eficientizeze diverse acțiuni. În acest proiect am dorit să evidențiez importanța unei baze de date într-un service auto.

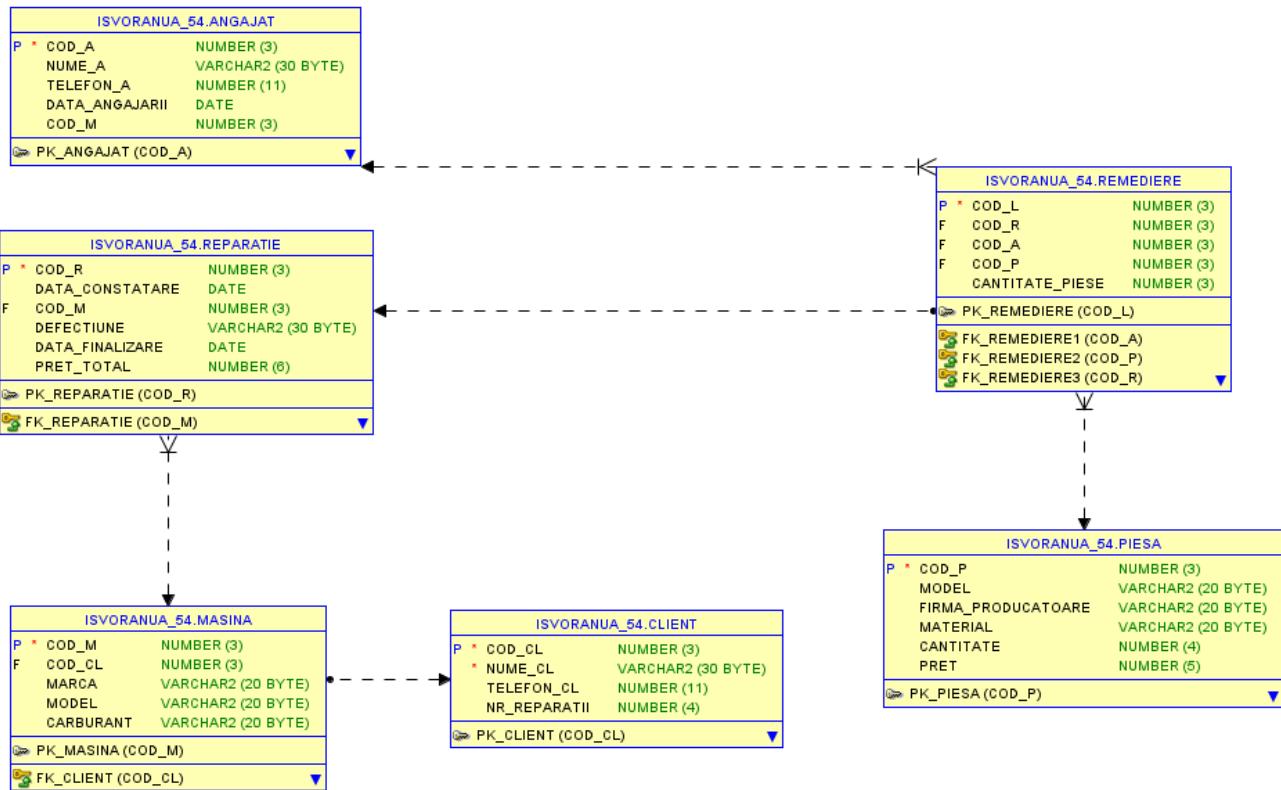
☺ Consider că prin baza de date creată este mult mai ușor de ținut evidența clientilor, pieselor existente în depozit și lista tuturor reparațiilor aduse unei mașini.

☺ În această afacere, piesele vin ambalate de la firma producătoare în depozitul service-ului, unde sunt sortate în funcție de modelul acestora.

Acest model este conceput pentru un singur service auto, dar el poate fi extins.

☺ Clientii sunt conștienți că datele furnizate angajaților sunt confidențiale și sunt de acord cu acest lucru.

☺ Datele despre o mașină intrată în service vor fi introduse într-o bază de date ce va furniza istoricul reparațiilor ale acesteia (problemele remediate, piesele înlocuite și data reparației).



Isvoranu Alexandru-Florian

```
DROP TABLE Remediere;
DROP TABLE Angajat;
DROP TABLE Reparatie;
DROP TABLE Piesa;
DROP TABLE Masina;
DROP TABLE Client;

CREATE TABLE Client
(cod_cl NUMBER(3) CONSTRAINT pk_client PRIMARY KEY,
nume_cl VARCHAR2(30) NOT NULL,
telefon_cl NUMBER(11),
nr_reparatii NUMBER(4));

CREATE TABLE Masina
(cod_m NUMBER(3) CONSTRAINT pk_masina PRIMARY KEY,
cod_cl NUMBER(3),
CONSTRAINT fk_client FOREIGN KEY (cod_cl) REFERENCES Client(cod_cl),
marca VARCHAR2(20),
model VARCHAR2(20),
carburant VARCHAR2(20));

CREATE TABLE Piesa
(cod_p NUMBER(3) CONSTRAINT pk_piesa PRIMARY KEY,
model VARCHAR2(20),
firma_producatoare VARCHAR2(20),
material VARCHAR2(20),
cantitate NUMBER(4),
pret NUMBER(5));

CREATE TABLE Reparatie
(cod_r NUMBER(3) CONSTRAINT pk_reparatie PRIMARY KEY,
```

Isvoranu Alexandru-Florian

```
data_constatare date,  
cod_m NUMBER(3),  
CONSTRAINT fk_reparatie FOREIGN KEY (cod_m) REFERENCES Masina(cod_m),  
defectiune VARCHAR2(30),  
data_finalizare date,  
pret_total number(6));
```

```
CREATE TABLE Angajat  
(cod_a NUMBER(3) CONSTRAINT pk_angajat PRIMARY KEY,  
nume_a VARCHAR2(30),  
telefon_a number(11),  
data_angajarii date,  
cod_m NUMBER(3));
```

```
CREATE TABLE Remediere  
(cod_l NUMBER(3) CONSTRAINT pk_remediere PRIMARY KEY,  
cod_r NUMBER(3),  
CONSTRAINT fk_remediere3 FOREIGN KEY (cod_r) REFERENCES Reparatie(cod_r),  
cod_a NUMBER(3),  
CONSTRAINT fk_remediere1 FOREIGN KEY (cod_a) REFERENCES Angajat(cod_a),  
cod_p NUMBER(3),  
CONSTRAINT fk_remediere2 FOREIGN KEY (cod_p) REFERENCES Piesa(cod_p),  
cantitate_piese NUMBER(3));  
--populez tabelele
```

```
insert into Client(cod_cl,nume_cl,telefon_cl,nr_reparati) values  
(101,'Vasile Popescu',718427392,3);  
insert into Client(cod_cl,nume_cl,telefon_cl,nr_reparati) values  
(102,'Lucian Popa',713035219,1);  
insert into Client(cod_cl,nume_cl,telefon_cl,nr_reparati) values  
(103,'Stefan Diaconescu',783488883,13);
```

Isvoranu Alexandru-Florian

```
insert into Client(cod_cl,nume_cl,telefon_cl,nr_reparatii) values  
(104,'Elena Sonda',787749800,4);  
insert into Client(cod_cl,nume_cl,telefon_cl,nr_reparatii) values  
(105,'Carina Pavel',785583573,27);  
insert into Client(cod_cl,nume_cl,telefon_cl,nr_reparatii) values  
(106,'Alexandru-Florian Isvoranu',717821345,0);  
insert into Client(cod_cl,nume_cl,telefon_cl,nr_reparatii) values  
(107,'Vicentiu Negoita',757521976,10);  
insert into Client(cod_cl,nume_cl,telefon_cl,nr_reparatii) values  
(108,'Vlad Catalin',731224216,4);  
insert into Client(cod_cl,nume_cl,telefon_cl,nr_reparatii) values  
(109,'Livia Craciun',752385222,0);  
insert into Client(cod_cl,nume_cl,telefon_cl,nr_reparatii) values  
(110,'Catalin Diaconu',714530592,21);  
insert into Client(cod_cl,nume_cl,telefon_cl,nr_reparatii) values  
(111,'Vlad Dobre',714563192,2);
```

```
insert into Masina (cod_m,cod_cl,marca,model,carburant) values  
(501,101,'Skoda','Octavia','Motorina');  
insert into Masina (cod_m,cod_cl,marca,model,carburant) values  
(502,101,'VolksWagen','Passat','Motorina');  
insert into Masina (cod_m,cod_cl,marca,model,carburant) values  
(503,102,'Dacia','Logan','Benzina');  
insert into Masina (cod_m,cod_cl,marca,model,carburant) values  
(504,103,'Seat','Leon','Motorina');  
insert into Masina (cod_m,cod_cl,marca,model,carburant) values  
(505,104,'Toyota','Corolla','Benzina');  
insert into Masina (cod_m,cod_cl,marca,model,carburant) values  
(506,105,'Skoda','Fabia','Benzina');  
insert into Masina (cod_m,cod_cl,marca,model,carburant) values  
(507,105,'Kia','Ceed','Benzina');
```

Ivoranu Alexandru-Florian

```
insert into Masina (cod_m,cod_cl,marca,model,carburant) values
(508,106,'Audi','A6','Motorina');

insert into Masina (cod_m,cod_cl,marca,model,carburant) values
(509,107,'BMW','X3','Motorina');

insert into Masina (cod_m,cod_cl,marca,model,carburant) values
(510,108,'Dacia','Duster','Motorina');

insert into Masina (cod_m,cod_cl,marca,model,carburant) values
(511,109,'Mercedes-Benz','C220','Motorina');

insert into Masina (cod_m,cod_cl,marca,model,carburant) values
(512,110,'BMW','M3','Benzina');

insert into Masina (cod_m,cod_cl,marca,model,carburant) values
(513,110,'Audi','A4','Motorina');

insert into Masina (cod_m,cod_cl,marca,model,carburant) values
(514,111,'Mazda','3','Motorina');

insert into Masina (cod_m,cod_cl,marca,model,carburant) values
(515,111,'Nissan','Qashqai','Benzina');

insert into Piesa(cod_p,model,firma_producatoare,material,cantitate,pret) values
(901,'Amortizor','Bosch','Plastic',10,150);

insert into Piesa(cod_p,model,firma_producatoare,material,cantitate,pret) values
(902,'Placute de frana','Motrio','Otel',30,90);

insert into Piesa(cod_p,model,firma_producatoare,material,cantitate,pret) values
(903,'Filtru de ulei','Mann','Otel',35,25);

insert into Piesa(cod_p,model,firma_producatoare,material,cantitate,pret) values
(904,'Radiator','Bosch','Metal',0,300);

insert into Piesa(cod_p,model,firma_producatoare,material,cantitate,pret) values
(905,'Filtru de aer','Mann','Plastic',20,25);

insert into Piesa(cod_p,model,firma_producatoare,material,cantitate,pret) values
(906,'Disc de frana','Motrio','Otel',15,120);

insert into Piesa(cod_p,model,firma_producatoare,material,cantitate,pret) values
(907,'Planetara','QWP','Metal',8,240);
```

```
insert into Piesa(cod_p,model,firma_producatoare,material,cantitate,pret) values  
(908,'Capat de bara','FEBI BILSTEIN','Metal',10,150);  
  
insert into Piesa(cod_p,model,firma_producatoare,material,cantitate,pret) values  
(909,'Filtru de motorina','Motrio','Metal',15,80);  
  
insert into Piesa(cod_p,model,firma_producatoare,material,cantitate,pret) values  
(910,'Filtru de benzina','Motrio','Metal',15,70);  
  
insert into Piesa(cod_p,model,firma_producatoare,material,cantitate,pret) values  
(911,'Capat de bara','RINGER','Metal',10,190);  
  
insert into Piesa(cod_p,model,firma_producatoare,material,cantitate,pret) values  
(912,'Pompa de apa','RINGER','Metal',7,200);  
  
insert into Piesa(cod_p,model,firma_producatoare,material,cantitate,pret) values  
(913,'Planetara','RINGER','Metal',5,300);  
  
  
insert into Reparatie(cod_r,data_constatare,cod_m,defectiune,data_finalizare,pret_total) values  
(301,to_date ('17-06-2021','dd-mm-yyyy'),502,'Schimb placute',to_date ('17-06-2021','dd-mm-yyyy'),250);  
insert into Reparatie(cod_r,data_constatare,cod_m,defectiune,data_finalizare,pret_total) values  
(302,to_date ('23-05-2021','dd-mm-yyyy'),501,'Schimb pradiator',to_date ('25-06-2021','dd-mm-yyyy'),670);  
insert into Reparatie(cod_r,data_constatare,cod_m,defectiune,data_finalizare,pret_total) values  
(303,to_date ('10-09-2021','dd-mm-yyyy'),503,'Schimb filtre',to_date ('10-06-2021','dd-mm-yyyy'),400);  
insert into Reparatie(cod_r,data_constatare,cod_m,defectiune,data_finalizare,pret_total) values  
(304,to_date ('6-04-2021','dd-mm-yyyy'),504,'Schimb planetara',to_date ('7-04-2021','dd-mm-yyyy'),420);  
insert into Reparatie(cod_r,data_constatare,cod_m,defectiune,data_finalizare,pret_total) values  
(305,to_date ('24-10-2021','dd-mm-yyyy'),504,'Schimb filtre',to_date ('25-10-2021','dd-mm-yyyy'),380);  
insert into Reparatie(cod_r,data_constatare,cod_m,defectiune,data_finalizare,pret_total) values  
(306,to_date ('29-04-2021','dd-mm-yyyy'),505,'Schimb amortizor',to_date ('30-04-2021','dd-mm-yyyy'),390);  
insert into Reparatie(cod_r,data_constatare,cod_m,defectiune,data_finalizare,pret_total) values  
(307,to_date ('19-12-2021','dd-mm-yyyy'),506,'Schimb capete de bara',to_date ('20-12-2021','dd-mm-yyyy'),510);  
insert into Reparatie(cod_r,data_constatare,cod_m,defectiune,data_finalizare,pret_total) values  
(308,to_date ('15-01-2022','dd-mm-yyyy'),507,'Schimb frane',to_date ('15-01-2021','dd-mm-yyyy'),630);  
insert into Reparatie(cod_r,data_constatare,cod_m,defectiune,data_finalizare,pret_total) values  
(309,to_date ('02-02-2022','dd-mm-yyyy'),508,'Schimb filtre',to_date ('02-02-2022','dd-mm-yyyy'),400);
```

Ivoranu Alexandru-Florian

```
insert into Reparatie(cod_r,data_constatare,cod_m,defectiune,data_finalizare,pret_total) values  
(310,to_date ('23-03-2022','dd-mm-yyyy'),508,'Schimb planetara',to_date ('24-03-2022','dd-mm-yyyy'),680);  
  
insert into Angajat(cod_a,nume_a,telefon_a,data_angajarii,cod_m) values  
(1,'Zoe Staruiala',7442210155,to_date ('27-05-2019','dd-mm-yyyy'),0);  
insert into Angajat(cod_a,nume_a,telefon_a,data_angajarii,cod_m) values  
(2,'Pavel Andreeescu',770422322,to_date ('20-01-2020','dd-mm-yyyy'),1);  
insert into Angajat(cod_a,nume_a,telefon_a,data_angajarii,cod_m) values  
(3,'Eugen Rusu',786292674,to_date ('01-06-2019','dd-mm-yyyy'),1);  
insert into Angajat(cod_a,nume_a,telefon_a,data_angajarii,cod_m) values  
(4,'Cosmin Barbu',711189206,to_date ('17-08-2020','dd-mm-yyyy'),1);  
insert into Angajat(cod_a,nume_a,telefon_a,data_angajarii,cod_m) values  
(5,'Bianca Ciocîrlan',767195054,to_date ('12-10-2020','dd-mm-yyyy'),2);  
insert into Angajat(cod_a,nume_a,telefon_a,data_angajarii,cod_m) values  
(6,'Alberto Buse',764419047,to_date ('27-01-2021','dd-mm-yyyy'),2);  
insert into Angajat(cod_a,nume_a,telefon_a,data_angajarii,cod_m) values  
(7,'Dorin Stan',748814187,to_date ('27-02-2021','dd-mm-yyyy'),2);  
insert into Angajat(cod_a,nume_a,telefon_a,data_angajarii,cod_m) values  
(8,'Dragoa Calinescu',710530458,to_date ('03-12-2019','dd-mm-yyyy'),3);  
insert into Angajat(cod_a,nume_a,telefon_a,data_angajarii,cod_m) values  
(9,'Iulian Popa',728225525,to_date ('14-09-2020','dd-mm-yyyy'),3);  
insert into Angajat(cod_a,nume_a,telefon_a,data_angajarii,cod_m) values  
(10,'Iulian Fratila',736211143,to_date ('31-10-2019','dd-mm-yyyy'),1);  
insert into Angajat(cod_a,nume_a,telefon_a,data_angajarii,cod_m) values  
(11,'Petre Marin',738204143,to_date ('14-09-2018','dd-mm-yyyy'),1);  
insert into Angajat(cod_a,nume_a,telefon_a,data_angajarii,cod_m) values  
(12,'Dragos Stoichici',736159336,to_date ('08-10-2018','dd-mm-yyyy'),2);  
insert into Remediere(cod_l,cod_r,cod_a,cod_p,cantitate_piese) values  
(701,301,1,902,4);  
insert into Remediere(cod_l,cod_r,cod_a,cod_p,cantitate_piese) values  
(702,302,2,904,1);
```

Ivoranu Alexandru-Florian

```
insert into Remediere(cod_l,cod_r,cod_a,cod_p,cantitate_piese) values  
(703,303,3,903,1);  
insert into Remediere(cod_l,cod_r,cod_a,cod_p,cantitate_piese) values  
(704,303,3,905,1);  
insert into Remediere(cod_l,cod_r,cod_a,cod_p,cantitate_piese) values  
(705,303,3,910,1);  
insert into Remediere(cod_l,cod_r,cod_a,cod_p,cantitate_piese) values  
(706,304,4,907,1);  
insert into Remediere(cod_l,cod_r,cod_a,cod_p,cantitate_piese) values  
(707,305,3,903,1);  
insert into Remediere(cod_l,cod_r,cod_a,cod_p,cantitate_piese) values  
(708,305,3,905,1);  
insert into Remediere(cod_l,cod_r,cod_a,cod_p,cantitate_piese) values  
(709,305,3,909,1);  
insert into Remediere(cod_l,cod_r,cod_a,cod_p,cantitate_piese) values  
(710,306,5,901,1);  
insert into Remediere(cod_l,cod_r,cod_a,cod_p,cantitate_piese) values  
(711,307,6,908,2);  
insert into Remediere(cod_l,cod_r,cod_a,cod_p,cantitate_piese) values  
(712,308,7,902,4);  
insert into Remediere(cod_l,cod_r,cod_a,cod_p,cantitate_piese) values  
(713,308,7,906,4);  
insert into Remediere(cod_l,cod_r,cod_a,cod_p,cantitate_piese) values  
(714,309,8,903,1);  
insert into Remediere(cod_l,cod_r,cod_a,cod_p,cantitate_piese) values  
(715,309,8,905,1);  
insert into Remediere(cod_l,cod_r,cod_a,cod_p,cantitate_piese) values  
(716,309,8,909,1);  
insert into Remediere(cod_l,cod_r,cod_a,cod_p,cantitate_piese) values  
(717,310,9,907,1);  
commit;
```

--1. Sa se afiseze numarul de reparatii aferent fiecarui client.

```
SELECT NUME_CL, NR_REPARATII  
FROM CLIENT  
ORDER BY NR_REPARATII;
```

	NUME_CL	NR_REPARATII
1	Sebastian Dinu	0
2	Livia Craciun	0
3	Lucian Popa	1
4	Vlad Dobre	2
5	Vasile Popescu	3
6	Elena Sonda	4
7	Vlad Catalin	4
8	Vicentiu Negoita	10
9	Stefan Diaconescu	13
10	Catalin Diaconu	21
11	Carina Pavel	27

--2. Sa se afiseze fiecare marca de masina intrata si Service si frecventa lor.

```
SELECT Marca, COUNT (*)  
FROM MASINA  
GROUP BY Marca;
```

	MARCA	cou...
1	VolksWagen	1
2	BMW	2
3	Mazda	1
4	Nissan	1
5	Seat	1
6	Toyota	1
7	Audi	2
8	Dacia	2
9	Mercedes-Benz	1
10	Skoda	2
11	Kia	1

--3. Sa se afiseze fiecare firma producatoare si frecventa lor a pieselor din depozit.

```
SELECT FIRMA_PRODUCATOARE, COUNT(*)  
FROM PIESA  
GROUP BY FIRMA_PRODUCATOARE;
```

	FIRMA_PRODUCATOARE	COUNT(*)
1	FEBI BILSTEIN	1
2	Motrio	4
3	RINGER	3
4	Bosch	2
5	Mann	2
6	QWP	1

--4. Sa se afiseze numarul de reparatii din fiecare luna.

```
select extract(year from DATA_FINALIZARE) as yr, extract(month from DATA_FINALIZARE) as mon,  
COUNT(*)  
FROM Reparatie
```

```
group by extract(year from DATA_FINALIZARE), extract(month from DATA_FINALIZARE)  
order by yr, mon;
```

	YR	MON	COUNT(*)
1	2021	1	1
2	2021	3	1
3	2021	5	1
4	2021	6	3
5	2021	10	1
6	2021	12	1
7	2022	2	1
8	2022	3	1

--5. Sa se afiseze toate datele despre o reparatie.

```
select r.cod_l, c.nume_cl, a.nume_a, p.model, r.cantitate_piese  
from angajat a, piesa p, remediere r, reparatie rp, masina m, client c  
where a.cod_a=r.cod_a and p.cod_p=r.cod_p and rp.cod_m=m.cod_m;
```

	COD_L	NUME_CL	NUME_A	MODEL	CANTITATE_PIESE
1	701	Vasile Popescu	Zoe Staruiala	Placute de frana	4
2	702	Vasile Popescu	Pavel Andreeescu	Radiator	1
3	709	Vasile Popescu	Eugen Rusu	Filtru de motorina	1
4	708	Vasile Popescu	Eugen Rusu	Filtru de aer	1
5	707	Vasile Popescu	Eugen Rusu	Filtru de ulei	1
6	705	Vasile Popescu	Eugen Rusu	Filtru de benzina	1
7	704	Vasile Popescu	Eugen Rusu	Filtru de aer	1
8	703	Vasile Popescu	Eugen Rusu	Filtru de ulei	1
9	706	Vasile Popescu	Cosmin Barbu	Planetara	1
10	710	Vasile Popescu	Bianca Ciocirlan	Amortizor	1
11	711	Vasile Popescu	Alberto Buse	Capat de bara	2
12	713	Vasile Popescu	Dorin Stan	Disc de frana	4
13	712	Vasile Popescu	Dorin Stan	Placute de frana	4
14	716	Vasile Popescu	Dragoa Calinescu	Filtru de motorina	1
15	715	Vasile Popescu	Dragoa Calinescu	Filtru de aer	1
16	714	Vasile Popescu	Dragoa Calinescu	Filtru de ulei	1
17	717	Vasile Popescu	Iulian Popa	Planetara	1

select * from client;

--6. Sa se adauge coloana "Email" in tabelul "Client".

ALTER TABLE Client

ADD email VARCHAR2(30);

Table CLIENT altered.

--7. Sa se stearga coloana "Email" din tabelul "Client".

ALTER TABLE Client

DROP COLUMN email;

Table CLIENT altered.

Isvoranu Alexandru-Florian

--8. Sa se creasca pretul pieselor ce costa mai putin de 100 u.m. cu 10 u.m.

update Piesa

set Pret=Pret+10

where pret<100;

5 rows updated.

--9. Sa se scada cantitatea discurilor cu 4 unitati.

update Piesa

set Cantitate=Cantitate-4

where model like 'Disc%';

1 row updated.

--10. Sa se scada cantitatea placutelor de frana cu 4 unitati.

update Piesa

set Cantitate=Cantitate-4

where model = 'Placute de frana';

1 row updated.

--11. Sa se sterga angajatii prezenti in firma inainte de anul 2019.

delete from Angajat

where data_angajarii<to_date('01-01-2019','dd-mm-yyyy');

2 rows deleted.

--12. Sa se sterga piesele de marca "RINGER".

delete from Piesa

where Firma_producatoare = 'RINGER';

3 rows deleted.

--1. Sa se afiseze angajatii care au efectuat cel putin o reparatie.

select *

from angajat inner join remediere on

angajat.cod_a=remediere.cod_a;

	COD_A	NUME_A	TELEFON_A	DATA_ANGAJARII	COD_M	COD_L	COD_R	COD_A_1	COD_P	CANTITATE_PIESE
1	1	Zoe Staruiala	7442210155	27-MAY-19	1	701	301	1	902	4
2	2	Pavel Andreescu	770422322	20-JAN-20	1	702	302	2	904	1
3	3	Eugen Rusu	786292674	01-JUN-19	1	703	303	3	903	1
4	3	Eugen Rusu	786292674	01-JUN-19	1	704	303	3	905	1
5	3	Eugen Rusu	786292674	01-JUN-19	1	705	303	3	910	1
6	4	Cosmin Barbu	711189206	17-AUG-20	1	706	304	4	907	1
7	3	Eugen Rusu	786292674	01-JUN-19	1	707	305	3	903	1
8	3	Eugen Rusu	786292674	01-JUN-19	1	708	305	3	905	1
9	3	Eugen Rusu	786292674	01-JUN-19	1	709	305	3	909	1
10	5	Bianca Ciocirlan	767195054	12-OCT-20	2	710	306	5	901	1
11	6	Alberto Buse	764419047	27-JAN-21	2	711	307	6	908	2
12	7	Dorin Stan	748814187	27-FEB-21	2	712	308	7	902	4
13	7	Dorin Stan	748814187	27-FEB-21	2	713	308	7	906	4
14	8	Dragoa Calinescu	710530458	03-DEC-19	3	714	309	8	903	1
15	8	Dragoa Calinescu	710530458	03-DEC-19	3	715	309	8	905	1
16	8	Dragoa Calinescu	710530458	03-DEC-19	3	716	309	8	909	1
17	9	Iulian Popa	728225525	14-SEP-20	3	717	310	9	907	1

--2. Sa se afiseze masinile care au fost reparate.

select masina.marca, masina.model

from masina inner join reparatie on

masina.cod_m=reparatie.cod_m;

	MARCA	MODEL
1	Skoda	Octavia
2	VolksWagen	Passat
3	Dacia	Logan
4	Seat	Leon
5	Seat	Leon
6	Toyota	Corolla
7	Skoda	Fabia
8	Kia	Ceed
9	Audi	A6
10	Audi	A6

--3. Sa se afiseze pentru fiecare masina reparatiile.

select *

from masina left join reparatie on

masina.cod_m=reparatie.cod_m;

	COD_M	COD_CL	MARCA	MODEL	CARBURANT	COD_R	DATA_CONSTATARE	COD_M_1	DEFECTIUNE	DATA_FINALIZARE	PRET_TOTAL
1	502	101	VolksWagen	Passat	Motorina	301	17-JUN-21	502	Schimb placute	17-JUN-21	250
2	501	101	Skoda	Octavia	Motorina	302	23-MAY-21	501	Schimb pradiator	25-JUN-21	670
3	503	102	Dacia	Logan	Benzina	303	10-SEP-21	503	Schimb filtre	10-JUN-21	400
4	504	103	Seat	Leon	Motorina	304	06-MAR-21	504	Schimb planetara	07-MAR-21	420
5	504	103	Seat	Leon	Motorina	305	24-OCT-21	504	Schimb filtre	25-OCT-21	380
6	505	104	Toyota	Corolla	Benzina	306	30-APR-21	505	Schimb amortizor	01-MAY-21	390
7	506	105	Skoda	Fabia	Benzina	307	19-DEC-21	506	Schimb capete de bara	20-DEC-21	510
8	507	105	Kia	Ceed	Benzina	308	15-JAN-22	507	Schimb frane	15-JAN-21	630
9	508	106	Audi	A6	Motorina	309	02-FEB-22	508	Schimb filtre	02-FEB-22	400
10	508	106	Audi	A6	Motorina	310	23-MAR-22	508	Schimb planetara	24-MAR-22	680
11	512	110	BMW	M3	Benzina	(null)	(null)	(null)	(null)	(null)	(null)
12	515	111	Nissan	Qashqai	Benzina	(null)	(null)	(null)	(null)	(null)	(null)
13	510	108	Dacia	Duster	Motorina	(null)	(null)	(null)	(null)	(null)	(null)
14	514	111	Mazda	3	Motorina	(null)	(null)	(null)	(null)	(null)	(null)
15	513	110	Audi	A4	Motorina	(null)	(null)	(null)	(null)	(null)	(null)
16	509	107	BMW	X3	Motorina	(null)	(null)	(null)	(null)	(null)	(null)
17	511	109	Mercedes-Benz	C220	Motorina	(null)	(null)	(null)	(null)	(null)	(null)

--4. Sa se afiseze pentru fiecare piesa reparatia care o contine.

select *

from piesa left join remediere on

remediere.cod_p=piesa.cod_p;

Ivoranu Alexandru-Florian

	COD_P	MODEL	FIRMA_PRODUCATOARE	MATERIAL	CANTITATE	PRET	COD_L	COD_R	COD_A	COD_P_1	CANTITATE_PIESE
1	902	Placute de frana	Motrio	Otel	26	100	701	301	1	902	4
2	904	Radiator	Bosch	Metal	5	300	702	302	2	904	1
3	903	Filtru de ulei	Mann	Otel	35	45	703	303	3	903	1
4	905	Filtru de aer	Mann	Plastic	20	45	704	303	3	905	1
5	910	Filtru de benzina	Motrio	Metal	15	90	705	303	3	910	1
6	907	Planetara	QWP	Metal	8	240	706	304	4	907	1
7	903	Filtru de ulei	Mann	Otel	35	45	707	305	3	903	1
8	905	Filtru de aer	Mann	Plastic	20	45	708	305	3	905	1
9	909	Filtru de motorina	Motrio	Metal	15	100	709	305	3	909	1
10	901	Amortizor	Bosch	Plastic	10	150	710	306	5	901	1
11	908	Capat de bara	FEBI BILSTEIN	Metal	10	150	711	307	6	908	2
12	902	Placute de frana	Motrio	Otel	26	100	712	308	7	902	4
13	906	Disc de frana	Motrio	Otel	11	120	713	308	7	906	4
14	903	Filtru de ulei	Mann	Otel	35	45	714	309	8	903	1
15	905	Filtru de aer	Mann	Plastic	20	45	715	309	8	905	1
16	909	Filtru de motorina	Motrio	Metal	15	100	716	309	8	909	1
17	907	Planetara	QWP	Metal	8	240	717	310	9	907	1

--5. Sa se afiseze pentru fiecare reparatie angajatul care a lucrat la ea.

select *

from angajat right join remediere on

remediere.cod_a=angajat.cod_a;

	COD_A	NUME_A	TELEFON_A	DATA_ANGAJARII	COD_M	COD_L	COD_R	COD_A_1	COD_P	CANTITATE_PIESE
1	1	Zoe Staruiala	7442210155	27-MAY-19	1	701	301	1	902	4
2	2	Pavel Andreeescu	770422322	20-JAN-20	1	702	302	2	904	1
3	3	Eugen Rusu	786292674	01-JUN-19	1	709	305	3	909	1
4	3	Eugen Rusu	786292674	01-JUN-19	1	708	305	3	905	1
5	3	Eugen Rusu	786292674	01-JUN-19	1	707	305	3	903	1
6	3	Eugen Rusu	786292674	01-JUN-19	1	705	303	3	910	1
7	3	Eugen Rusu	786292674	01-JUN-19	1	704	303	3	905	1
8	3	Eugen Rusu	786292674	01-JUN-19	1	703	303	3	903	1
9	4	Cosmin Barbu	711189206	17-AUG-20	1	706	304	4	907	1
10	5	Bianca Ciocirlan	767195054	12-OCT-20	2	710	306	5	901	1
11	6	Alberto Buse	764419047	27-JAN-21	2	711	307	6	908	2
12	7	Dorin Stan	748814187	27-FEB-21	2	713	308	7	906	4
13	7	Dorin Stan	748814187	27-FEB-21	2	712	308	7	902	4
14	8	Dragoa Calinescu	710530458	03-DEC-19	3	716	309	8	909	1
15	8	Dragoa Calinescu	710530458	03-DEC-19	3	715	309	8	905	1
16	8	Dragoa Calinescu	710530458	03-DEC-19	3	714	309	8	903	1
17	9	Iulian Popa	728225525	14-SEP-20	3	717	310	9	907	1

--6. Sa se afiseze pentru fiecare client masinile sale.

select *

from masina right join client on

masina.cod_cl=client.cod_cl;

Ivoranu Alexandru-Florian

	COD_M	COD_CL	MARCA	MODEL	CARBURANT	COD_CL_1	NUME_CL	TELEFON_CL	NR_REPARATII
1	501	101	Skoda	Octavia	Motorina	101	Vasile Popescu	718427392	3
2	502	101	VolksWagen	Passat	Motorina	101	Vasile Popescu	718427392	3
3	503	102	Dacia	Logan	Benzina	102	Lucian Popa	713035219	1
4	504	103	Seat	Leon	Motorina	103	Stefan Diaconescu	783488883	13
5	505	104	Toyota	Corolla	Benzina	104	Elena Sonda	787749800	4
6	506	105	Skoda	Fabia	Benzina	105	Carina Pavel	785583573	27
7	507	105	Kia	Ceed	Benzina	105	Carina Pavel	785583573	27
8	508	106	Audi	A6	Motorina	106	Sebastian Dinu	717821345	0
9	509	107	BMW	X3	Motorina	107	Vicentiu Negoita	757521976	10
10	510	108	Dacia	Duster	Motorina	108	Vlad Catalin	731224216	4
11	511	109	Mercedes-Benz	C220	Motorina	109	Livia Craciun	752385222	0
12	512	110	BMW	M3	Benzina	110	Catalin Diaconu	714530592	21
13	513	110	Audi	A4	Motorina	110	Catalin Diaconu	714530592	21
14	514	111	Mazda	3	Motorina	111	Vlad Dobre	714563192	2
15	515	111	Nissan	Qashqai	Benzina	111	Vlad Dobre	714563192	2

--7. Sa se afiseze toate masinile si toate reparatiile.

```
select masina.cod_m, masina.marca, reparatie.cod_r, reparatie.defectiune, reparatie.pret_total
from masina left outer join reparatie on
masina.cod_m=reparatie.cod_m
UNION
select masina.cod_m, masina.marca, reparatie.cod_r, reparatie.defectiune, reparatie.pret_total
from masina right outer join reparatie on
masina.cod_m=reparatie.cod_m;
```

	COD_M	MARCA	COD_R	DEFECTIUNE	PRET_TOTAL
1	501	Skoda	302	Schimb pradiator	670
2	502	VolksWagen	301	Schimb placute	250
3	503	Dacia	303	Schimb filtre	400
4	504	Seat	304	Schimb planetara	420
5	504	Seat	305	Schimb filtre	380
6	505	Toyota	306	Schimb amortizor	390
7	506	Skoda	307	Schimb capete de bara	510
8	507	Kia	308	Schimb frane	630
9	508	Audi	309	Schimb filtre	400
10	508	Audi	310	Schimb planetara	680
11	509	BMW	(null)	(null)	(null)
12	510	Dacia	(null)	(null)	(null)
13	511	Mercedes-Benz	(null)	(null)	(null)
14	512	BMW	(null)	(null)	(null)
15	513	Audi	(null)	(null)	(null)
16	514	Mazda	(null)	(null)	(null)
17	515	Nissan	(null)	(null)	(null)

--8. Sa se afiseze toti angajatii si toate reparatiile.

```
select angajat.cod_a, angajat.nume_a, remediere.cod_l, remediere.cod_p, remediere.cantitate_piese
from angajat left outer join remediere on
remediere.cod_a=angajat.cod_a
UNION
select angajat.cod_a, angajat.nume_a, remediere.cod_l, remediere.cod_p, remediere.cantitate_piese
from angajat right outer join remediere on
remediere.cod_a=angajat.cod_a;
```

	COD_A	NUME_A	COD_L	COD_P	CANTITATE_PIESE
1	1	Zoe Staruiala	701	902	4
2	2	Pavel Andreeescu	702	904	1
3	3	Eugen Rusu	703	903	1
4	3	Eugen Rusu	704	905	1
5	3	Eugen Rusu	705	910	1
6	3	Eugen Rusu	707	903	1
7	3	Eugen Rusu	708	905	1
8	3	Eugen Rusu	709	909	1
9	4	Cosmin Barbu	706	907	1
10	5	Bianca Ciocirlan	710	901	1
11	6	Alberto Buse	711	908	2
12	7	Dorin Stan	712	902	4
13	7	Dorin Stan	713	906	4
14	8	Dragoa Calinescu	714	903	1
15	8	Dragoa Calinescu	715	905	1
16	8	Dragoa Calinescu	716	909	1
17	9	Iulian Popa	717	907	1
18	10	Iulian Fratila	(null)	(null)	(null)

--1. Sa se afiseze numarul mediu de reparatii al unui client.

```
select round(avg(nr_reparatii),2) as medie
from client;
```

	MEDIE
1	7.73

--2. Sa se calculeze valoarea totala a pieselor din depozit.

```
select sum(cantitate*pret) as valoare_totala
from piesa;
```

	VALOARE_TOTALA
1	15665

--3. Sa se calculeze cel mai mic cost al unei reparatii.

```
select min(pret_total)  
from reparatie;
```

	MIN(PRET_TOTAL)
1	250

--4. Sa se calculeze cel mai mare cost al unei reparatii.

```
select max(pret_total)  
from reparatie;
```

	MAX(PRET_TOTAL)
1	680

--5. Sa se calculeze numarul mediu de piese folosite la o reparatie.

```
select round(avg(cantitate_piese),2) as medie  
from remediere;
```

	MEDIE
1	1.59

--6. Sa afle data de angajare a primului salariat.

```
select min(data_angajarii) as data  
from angajat;
```

	DATA
1	27-MAY-19

--1. Sa se afiseze angajatii si nivelul ierarhic al acestora pornind de la angajatul cu id-ul 1.

```
select cod_a, nume_a, cod_m, level from angajat  
connect by prior cod_a=cod_m
```

start with cod_a=1

order by level;

	COD_A	NUME_A	COD_M	LEVEL
1	1	Zoe Staruiala	0	1
2	4	Cosmin Barbu	1	2
3	2	Pavel Andreescu	1	2
4	3	Eugen Rusu	1	2
5	10	Iulian Fratila	1	2
6	9	Iulian Popa	3	3
7	8	Dragoa Calinescu	3	3
8	6	Alberto Buse	2	3
9	5	Bianca Ciocirlan	2	3
10	7	Dorin Stan	2	3

--2. Afisati angajatii companiei subordonati inregistrarii radacina sub forma de organigrama.

select level, lpad(' ',level)||nume_a from angajat

where data_angajarii>to_date('01-01-2019','dd-mm-yyyy')

connect by prior cod_a=cod_m

start with cod_a=1;

	LEVEL	LPAD(" ",LEVEL) NUME_A
1	1	Zoe Staruiala
2	2	Pavel Andreescu
3	3	Bianca Ciocirlan
4	3	Alberto Buse
5	3	Dorin Stan
6	2	Eugen Rusu
7	3	Dragoa Calinescu
8	3	Iulian Popa
9	2	Cosmin Barbu
10	2	Iulian Fratila

--3. Sa se selecteze angajatii si gradul de subordonare numai pentru cei angajati dupa 01.01.2020.

select cod_a, nume_a, cod_m, data_angajarii, level from angajat

where data_angajarii>to_date('01-01-2020','dd-mm-yyyy')

connect by prior cod_a=cod_m
start with cod_a=1;

	COD_A	NUME_A	COD_M	DATA_ANGAJARII	LEVEL
1	2	Pavel Andreeescu	1	20-JAN-20	2
2	5	Bianca Ciocirlan	2	12-OCT-20	3
3	6	Alberto Buse	2	27-JAN-21	3
4	7	Dorin Stan	2	27-FEB-21	3
5	9	Iulian Popa	3	14-SEP-20	3
6	4	Cosmin Barbu	1	17-AUG-20	2

--4. Sa se afiseze superiorii angajatilor aflati pe ultimul nivel de subordonare.

select nume_a, level-1 numar_superiori,
sys_connect_by_path(cod_a,'/') cod_superiori
from angajat
start with cod_a=1
connect by prior cod_a=cod_m
order by level desc;

	NUME_A	NUMAR_SUPERIORI	COD_SUPERIORI
1	Dorin Stan		2 / 1 / 2 / 7
2	Iulian Popa		2 / 1 / 3 / 9
3	Dragoa Calinescu		2 / 1 / 3 / 8
4	Alberto Buse		2 / 1 / 2 / 6
5	Bianca Ciocirlan		2 / 1 / 2 / 5
6	Iulian Fratila		1 / 1 / 10
7	Cosmin Barbu		1 / 1 / 4
8	Eugen Rusu		1 / 1 / 3
9	Pavel Andreeescu		1 / 1 / 2
10	Zoe Staruiala		0 / 1

--5. Sa se afiseze toti superiorii lui Dorin Stan.

select cod_a, nume_a, cod_m, data_angajarii, level from angajat
connect by prior cod_a=cod_m

start with nume_a = 'Dorin Stan';

	COD_A	NUME_A	COD_M	DATA_ANGAJARII	LEVEL
1	7	Dorin Stan	2	27-FEB-21	1

--6. Sa se afiseze toti subordonatii lui 'Pavel Andreeescu'.

select cod_a, nume_a, cod_m, data_angajarii, level from angajat

connect by prior cod_a=cod_m

start with nume_a = 'Pavel Andreeescu'

order by level;

	COD_A	NUME_A	COD_M	DATA_ANGAJARII	LEVEL
1	2	Pavel Andreeescu	1	20-JAN-20	1
2	6	Alberto Buse	2	27-JAN-21	2
3	7	Dorin Stan	2	27-FEB-21	2
4	5	Bianca Ciocirlan	2	12-OCT-20	2

--1. Sa se calculeze discountul clientilor in functie de numarul de reparatii:

- 5% daca au intre 5 si 20 de reparatii in istoric
- 10% daca au intre 20 si 50 de reparatii in istoric
- 15% daca au cel putin 50 de reparatii in istoric

--altfel, va fi 0.

```
select cod_cl, nume_cl, nr_reparatii,
case
when nr_reparatii>5 and nr_reparatii<20 then 0.05
when nr_reparatii>20 and nr_reparatii<50 then 0.1
when nr_reparatii>50 then 0.15
else 0 end comision
from client;
```

	COD_CL	NUME_CL	NR_REPARATII	COMISION
1	101	Vasile Popescu	3	0
2	102	Lucian Popa	1	0
3	103	Stefan Diaconescu	13	0.05
4	104	Elena Sonda	4	0
5	105	Carina Pavel	27	0.1
6	106	Sebastian Dinu	0	0
7	107	Vicentiu Negoita	10	0.05
8	108	Vlad Catalin	4	0
9	109	Livia Craciun	0	0
10	110	Catalin Diaconu	21	0.1
11	111	Vlad Dobre	2	0

--2. Sa se calculeze bonusul salarial al angajatilor in functie de vechimea in firma:

- 150 lei/luna daca este angajat in anul 2020
 - 250 lei/luna daca este angajat in anul 2019
 - 350 lei/luna daca este angajat mai demult de anul 2019
- altfel, va fi 0.

```

select cod_a, nume_a, data_angajarii,
case
when data_angajarii>to_date('01-01-2020','dd-mm-yyyy') and data_angajarii<to_date('01-01-2021','dd-
mm-yyyy') then 150
when data_angajarii>to_date('01-01-2019','dd-mm-yyyy') and data_angajarii<to_date('01-01-2020','dd-
mm-yyyy') then 250
when data_angajarii<to_date('01-01-2019','dd-mm-yyyy') then 350
else 0 end bonus
from angajat;

```

	COD_A	NUME_A	DATA_ANGAJARII	BONUS
1	1	Zoe Staruiala	27-MAY-19	250
2	2	Pavel Andreeescu	20-JAN-20	150
3	3	Eugen Rusu	01-JUN-19	250
4	4	Cosmin Barbu	17-AUG-20	150
5	5	Bianca Ciocîrlan	12-OCT-20	150
6	6	Alberto Buse	27-JAN-21	0
7	7	Dorin Stan	27-FEB-21	0
8	8	Dragoa Calinescu	03-DEC-19	250
9	9	Iulian Popa	14-SEP-20	150
10	10	Iulian Fratila	31-OCT-19	250

--3. Sa se afiseze pentru fiecare angajat numele managerului.

```
SELECT nume_a,
       DECODE(cod_m, 1, 'Zoe Staruiala',
              2, 'Pavel Andreeescu',
              3, 'Eugen Rusu',
              'Manager') Manager
  FROM Angajat;
```

	NUME_A	MANAGER
1	Zoe Staruiala	Manager
2	Pavel Andreeescu	Zoe Staruiala
3	Eugen Rusu	Zoe Staruiala
4	Cosmin Barbu	Zoe Staruiala
5	Bianca Ciocîrlan	Pavel Andreeescu
6	Alberto Buse	Pavel Andreeescu
7	Dorin Stan	Pavel Andreeescu
8	Dragoa Calinescu	Eugen Rusu
9	Iulian Popa	Eugen Rusu
10	Iulian Fratila	Zoe Staruiala

--4. Sa se afiseze pentru fiecare motorizare pretul unui schimb de filtre.

```
SELECT cod_m, marca, model,
       DECODE(carburant, 'Benzina', 'Filtru de benzina+aer+ulei, 150 lei',
              'Motorina','Filtru de motorina+aer+ulei, 160 lei',
```

'Nu avem filtru') Pret_schimb_filtre

FROM Masina;

	COD_M	MARCA	MODEL	PRET_SCHIMB_FILTRE
1	501	Skoda	Octavia	Filtru de motorina+aer+ulei, 160 lei
2	502	VolksWagen	Passat	Filtru de motorina+aer+ulei, 160 lei
3	503	Dacia	Logan	Filtru de benzina+aer+ulei, 150 lei
4	504	Seat	Leon	Filtru de motorina+aer+ulei, 160 lei
5	505	Toyota	Corolla	Filtru de benzina+aer+ulei, 150 lei
6	506	Skoda	Fabia	Filtru de benzina+aer+ulei, 150 lei
7	507	Kia	Ceed	Filtru de benzina+aer+ulei, 150 lei
8	508	Audi	A6	Filtru de motorina+aer+ulei, 160 lei
9	509	BMW	X3	Filtru de motorina+aer+ulei, 160 lei
10	510	Dacia	Duster	Filtru de motorina+aer+ulei, 160 lei
11	511	Mercedes-Benz	C220	Filtru de motorina+aer+ulei, 160 lei
12	512	BMW	M3	Filtru de benzina+aer+ulei, 150 lei
13	513	Audi	A4	Filtru de motorina+aer+ulei, 160 lei
14	514	Mazda	3	Filtru de motorina+aer+ulei, 160 lei
15	515	Nissan	Qashqai	Filtru de benzina+aer+ulei, 150 lei

--1. Sa se afiseze toti angajatii prezenti in firma, in afara de cei angajati in 2020

select* from angajat

minus

select * from angajat where data_angajarii>to_date('01-01-2020','dd-mm-yyyy') and data_angajarii<to_date('31-12-2020','dd-mm-yyyy');

	COD_A	NUME_A	TELEFON_A	DATA_ANGAJARII	COD_M
1	1	Zoe Staruiala	7442210155	27-MAY-19	0
2	3	Eugen Rusu	786292674	01-JUN-19	1
3	6	Alberto Buse	764419047	27-JAN-21	2
4	7	Dorin Stan	748814187	27-FEB-21	2
5	8	Dragoa Calinescu	710530458	03-DEC-19	3
6	10	Iulian Fratila	736211143	31-OCT-19	1

--2. Sa se afiseze codul pieselor care sunt de marca "Motrio" si sunt folosite in cel putin 4 exemplare la fiecare remediere

select cod_p from piesa

where firma_producatoare = 'Motrio'

intersect

```
select cod_p from remediere  
where cantitate_piese>=2;
```

	COD_P
1	902
2	906

--3. Sa se calculeze distinct bonusul salarial pentru angajati folosind operatorul UNION:

- 150 lei/luna daca este angajat in anul 2020
 - 250 lei/luna daca este angajat in anul 2019
 - 350 lei/luna daca este angajat mai demult de anul 2019
- altfel, va fi 0.

```
select cod_a, nume_a, data_angajarii, 150 valoare_bonus  
from angajat  
where data_angajarii>=to_date('01-01-2020','dd-mm-yyyy') and data_angajarii<to_date('01-01-2021','dd-mm-yyyy')  
union  
select cod_a, nume_a, data_angajarii, 250 valoare_bonus  
from angajat  
where data_angajarii>=to_date('01-01-2019','dd-mm-yyyy') and data_angajarii<to_date('01-01-2020','dd-mm-yyyy')  
union  
select cod_a, nume_a, data_angajarii, 350 valoare_bonus  
from angajat  
where data_angajarii<to_date('01-01-2019','dd-mm-yyyy')  
union  
select cod_a, nume_a, data_angajarii, 0 valoare_bonus  
from angajat  
where data_angajarii>=to_date('01-01-2021','dd-mm-yyyy');
```

	COD_A	NUME_A	DATA_ANGAJARII	VALOARE_BONUS
1	1	Zoe Staruiala	27-MAY-19	250
2	2	Pavel Andreescu	20-JAN-20	150
3	3	Eugen Rusu	01-JUN-19	250
4	4	Cosmin Barbu	17-AUG-20	150
5	5	Bianca Ciocirlan	12-OCT-20	150
6	6	Alberto Buse	27-JAN-21	0
7	7	Dorin Stan	27-FEB-21	0
8	8	Dragoa Calinescu	03-DEC-19	250
9	9	Iulian Popa	14-SEP-20	150
10	10	Iulian Fratila	31-OCT-19	250

```
set serveroutput on;
```

```
--STRUCTURI REPETITIVE SI ALTERNATIVE
```

```
-- CASE
```

```
--1. Sa se calculeze discountul pentru un client citit in functie de numarul de reparatii:
```

- 5% daca au intre 5 si 20 de reparatii in istoric
- 10% daca au intre 20 si 50 de reparatii in istoric
- 15% daca au cel putin 50 de reparatii in istoric

```
--altfel, va fi 0.
```

```
declare
```

```
v_cod_cl NUMBER(4) := &cod;
```

```
v_nume_cl VARCHAR2(30);
```

```
v_nr_reparatii NUMBER(4);
```

```
begin
```

```
select nume_cl, nr_reparatii into v_nume_cl, v_nr_reparatii
```

```
from client
```

```
where cod_cl = v_cod_cl;
```

```
case
```

```
when v_nr_reparatii > 5 and v_nr_reparatii < 20 then
```

```
DBMS_OUTPUT.PUT_LINE('Clientul ' || v_nume_cl || ' are un discount de 5%');
```

Isvoranu Alexandru-Florian

```
when v_nr_reparatii > 20 and v_nr_reparatii < 50 then
    DBMS_OUTPUT.PUT_LINE('Clientul ' || v_nume_cl || ' are un discount de 10%');

when v_nr_reparatii > 50 then
    DBMS_OUTPUT.PUT_LINE('Clientul ' || v_nume_cl || ' are un discount de 15%');

else
    DBMS_OUTPUT.PUT_LINE('Clientul ' || v_nume_cl || ' nu are discount');

end case;

end;
/

```

Clientul Alexandru-Florian Isvoranu nu are discount

-- IF-THEN-ELSE

--2. Sa se calculeze bonusul salarial al unui angajat in functie de vechimea in firma:

- 150 lei/luna daca este angajat in anul 2020
 - 250 lei/luna daca este angajat in anul 2019
 - 350 lei/luna daca este angajat mai demult de anul 2019
- altfel, va fi 0.

DECLARE

```
v_cod_a NUMBER(4) := &cod;
v_nume_a VARCHAR2(30);
v_data_angajarii date;

BEGIN
    SELECT cod_a, nume_a, data_angajarii INTO v_cod_a, v_nume_a, v_data_angajarii
    FROM Angajat
    WHERE cod_a = v_cod_a;
```

IF v_data_angajarii > to_date('01-01-2020', 'dd-mm-yyyy') and v_data_angajarii < to_date('01-01-2021', 'dd-mm-yyyy') THEN

```
    DBMS_OUTPUT.PUT_LINE('Angajatul ' || v_nume_a || ' are bonusul salarial de 150.');
```

Ivoranu Alexandru-Florian

```
ELSIF v_data_angajarii>to_date('01-01-2019','dd-mm-yyyy') and v_data_angajarii<to_date('01-01-2020','dd-mm-yyyy') THEN
    DBMS_OUTPUT.PUT_LINE('Angajatul ' || v_nume_a || ' are bonusul salarial de 250.');
ELSIF v_data_angajarii<to_date('01-01-2019','dd-mm-yyyy') then
    DBMS_OUTPUT.PUT_LINE('Angajatul ' || v_nume_a || ' are bonusul salarial de 350.');
ELSE
    DBMS_OUTPUT.PUT_LINE('Angajatul ' || v_nume_a || ' nu are bonus salarial.');
END IF;
END;
/
| Angajatul Zoe Staruiala are bonusul salarial de 250.
```

```
-- LOOP
--3. Sa se afiseze numarul de reparatii pentru fiecare client
DECLARE
    v_cod_cl NUMBER(3) := 101;
    v_nr_reparatii NUMBER(4) := 0;
BEGIN
    LOOP
        BEGIN
            SELECT nr_reparatii INTO v_nr_reparatii
            FROM Client
            WHERE cod_cl = v_cod_cl;
            DBMS_OUTPUT.PUT_LINE('Clientul cu codul ' || v_cod_cl || ' a efectuat ' || v_nr_reparatii || ' reparatii.');
            v_cod_cl := v_cod_cl + 1;
        EXCEPTION
            WHEN NO_DATA_FOUND THEN
                DBMS_OUTPUT.PUT_LINE('Nu exist? nicio înregistrare în tabelul Client cu cod_cl ' || v_cod_cl);
                v_cod_cl := v_cod_cl + 1;
        END;
    END;
```

```
    EXIT WHEN v_cod_cl > 111;  
END LOOP;  
END;  
/  
  
Clientul cu codul 101 a efectuat 3 reparatii.  
Clientul cu codul 102 a efectuat 1 reparatii.  
Clientul cu codul 103 a efectuat 13 reparatii.  
Clientul cu codul 104 a efectuat 4 reparatii.  
Clientul cu codul 105 a efectuat 27 reparatii.  
Clientul cu codul 106 a efectuat 0 reparatii.  
Clientul cu codul 107 a efectuat 10 reparatii.  
Clientul cu codul 108 a efectuat 4 reparatii.  
Clientul cu codul 109 a efectuat 0 reparatii.  
Clientul cu codul 110 a efectuat 21 reparatii.  
Clientul cu codul 111 a efectuat 2 reparatii.
```

--WHILE

-- 4. Sa se afiseze angajati prezenți în firma începând cu anul 2020

DECLARE

v_data DATE := '31-DEC-2019';

v_id NUMBER(4) := 1;

v_nume VARCHAR2(30);

v_data_angajarii date;

BEGIN

WHILE v_id IS NOT NULL LOOP

BEGIN

SELECT nume_a, data_angajarii

INTO v_nume, v_data_angajarii

FROM Angajat

WHERE cod_a = v_id;

IF v_data_angajarii > v_data THEN

Ivoranu Alexandru-Florian

```
DBMS_OUTPUT.PUT_LINE(v_nume || ',' || v_data_angajarii);

END IF;

v_id := v_id + 1;

EXCEPTION

WHEN NO_DATA_FOUND THEN

v_id := NULL;

END;

END LOOP;

END;
```

/

```
Pavel Andreescu, 20-JAN-20
Cosmin Barbu, 17-AUG-20
Bianca Ciocirlan, 12-OCT-20
Alberto Buse, 27-JAN-21
Dorin Stan, 27-FEB-21
Iulian Popa, 14-SEP-20
```

```
-- COLECTII DE DATE

--INDEX-BY TABLE:

DECLARE

TYPE cod_m_index IS TABLE OF Masina.cod_m%TYPE INDEX BY PLS_INTEGER;

v_cod_m cod_m_index;

BEGIN

v_cod_m(1) := 501;
v_cod_m(2) := 502;
v_cod_m(3) := 503;

DBMS_OUTPUT.PUT_LINE('Coduri de masina: ');

FOR i IN v_cod_m.FIRST .. v_cod_m.LAST LOOP

DBMS_OUTPUT.PUT_LINE(v_cod_m(i));
```

Isvoranu Alexandru-Florian

```
END LOOP;  
END;  
  
/  
  
Coduri de masina:  
501  
502  
503
```

```
--NESTED TABLE:  
  
DECLARE  
  
  TYPE masini_type IS TABLE OF Masina%ROWTYPE;  
  v_masini masini_type := masini_type();  
  
BEGIN  
  
  v_masini.EXTEND(3);  
  v_masini(1).cod_m := 100;  
  v_masini(1).cod_cl := 1;  
  v_masini(1).marca := 'Ford';  
  v_masini(1).model := 'Focus';  
  v_masini(1).carburant := 'Benzina';  
  
  v_masini(2).cod_m := 101;  
  v_masini(2).cod_cl := 2;  
  v_masini(2).marca := 'Opel';  
  v_masini(2).model := 'Astra';  
  v_masini(2).carburant := 'Motorina';  
  
  v_masini(3).cod_m := 102;  
  v_masini(3).cod_cl := 3;  
  v_masini(3).marca := 'Renault';  
  v_masini(3).model := 'Clio';  
  v_masini(3).carburant := 'Benzina';
```

```
DBMS_OUTPUT.PUT_LINE('Masini: ');

FOR i IN v_masini.FIRST .. v_masini.LAST LOOP

    DBMS_OUTPUT.PUT_LINE(v_masini(i).cod_m || ' ' || v_masini(i).marca || ' ' || v_masini(i).model || ' ' ||
v_masini(i).carburant);

    END LOOP;

END;

/
```

```
Masini:
100 Ford Focus Benzina
101 Opel Astra Motorina
102 Renault Clio Benzina
```

```
--VARRAY

DECLARE

TYPE masina_type IS VARRAY(2) OF Masina%ROWTYPE;

v_masini masina_type := masina_type();

BEGIN

SELECT * BULK COLLECT INTO v_masini FROM Masina WHERE cod_m = 501 OR cod_m = 502;

DBMS_OUTPUT.PUT_LINE('Masini: ');

FOR i IN 1 .. v_masini.COUNT LOOP

    DBMS_OUTPUT.PUT_LINE(v_masini(i).cod_m || ' ' || v_masini(i).cod_cl || ' ' || v_masini(i).marca || ' ' ||
v_masini(i).model || ' ' || v_masini(i).carburant);

    END LOOP;

END;

/

Masini:
501 101 Skoda Octavia Motorina
502 101 VolksWagen Passat Motorina
```

```
-- EXCEPTII IMPLICITE
```

Ivoranu Alexandru-Florian

```
-- 1. Generarea unei exceptii implice (TOO_MANY_ROWS)

DECLARE
    v_nume_angajat VARCHAR2(30);
BEGIN
    SELECT nume_a INTO v_nume_angajat
    FROM Angajat
    WHERE cod_a > 5;
    DBMS_OUTPUT.PUT_LINE('Numele angajatului este: ' || v_nume_angajat);
EXCEPTION
    WHEN TOO_MANY_ROWS THEN
        DBMS_OUTPUT.PUT_LINE('Prea multe înregistrari returnate!');
END;
/
```

PL/SQL procedure successfully completed.

Prea multe înregistrari returnate!

```
--2. Generarea unei exceptii implice (VALUE_ERROR)

declare
    v_nume_client varchar2(20);
    v_nr_reparatii number(3);
    v_result NUMBER;
begin
    select nume_cl, nr_reparatii INTO v_nume_client, v_nr_reparatii from client
    where cod_cl=110;
    v_result:=v_nume_client+v_nr_reparatii;
    DBMS_OUTPUT.put_line(v_result);
EXCEPTION
    WHEN VALUE_ERROR THEN
```

Isvoranu Alexandru-Florian

```
DBMS_OUTPUT.PUT_LINE('A aparut o eroare de valoare');

end;

/
```

PL/SQL procedure successfully completed.

A aparut o eroare de valoare

--3. Exceptie implicita pentru incalcarea cheii primare (DUP_VAL_ON_INDEX)

```
DECLARE
v_cod_m NUMBER(3) := 501;
BEGIN
INSERT INTO Masina (cod_m, cod_cl, marca, model, carburant)
VALUES (v_cod_m, 999, 'Marca', 'Model', 'Benzina');
EXCEPTION
WHEN DUP_VAL_ON_INDEX THEN
DBMS_OUTPUT.PUT_LINE('Nu se poate adauga o masina cu cheia primara ' || v_cod_m || ' deoarece deja exista in
tabelul Masina.');
END;
/
PL/SQL procedure successfully completed.

Nu se poate adauga o masina cu cheia primara 501 deoarece deja exista in tabelul Masina.
```

--EXCEPTII EXPLICITE

--1. Exceptie pentru cazul in care se incercă actualizarea cantitatii pieselor cu o valoare negativa:

```
set serveroutput on;
DECLARE
negative_quantity EXCEPTION;
PRAGMA EXCEPTION_INIT(negative_quantity, -20001);
```

Ivoranu Alexandru-Florian

```
v_cantitate NUMBER := -5;  
BEGIN  
    IF v_cantitate < 0 THEN  
        RAISE negative_quantity;  
    ELSE  
        UPDATE Piesa SET cantitate = v_cantitate WHERE cod_p = 911;  
    END IF;  
EXCEPTION  
    WHEN negative_quantity THEN  
        DBMS_OUTPUT.PUT_LINE('Nu se poate actualiza cantitatea cu o valoare negativa!');  
END;  
/
```

```
| PL/SQL procedure successfully completed.  
| Nu se poate actualiza cantitatea cu o valoare negativa!
```

--2. Exceptie pentru cazul în care se încearca actualizarea
--numarului de telefon al unui client cu un numar diferit de 9 cifre:

```
DECLARE  
    numar_invalid EXCEPTION;  
    PRAGMA EXCEPTION_INIT(numar_invalid, -20002);  
    v_telefon_cl NUMBER := 83655172;  
BEGIN  
    IF LENGTH(v_telefon_cl) != 9 THEN  
        RAISE numar_invalid;  
    ELSE  
        UPDATE Client SET telefon_cl = v_telefon_cl WHERE cod_cl = 105;  
    END IF;  
EXCEPTION  
    WHEN numar_invalid THEN
```

Ivoranu Alexandru-Florian

```
DBMS_OUTPUT.PUT_LINE('Numarul de telefon nu contine exact 9 cifre!');

END;

/

PL/SQL procedure successfully completed.

Numarul de telefon nu contine exact 9 cifre!
```

--3. Exceptie pentru cazul în care se încearca crearea unei tabele deja existente:

```
DECLARE

tabela_exista exception;

pragma exception_init(tabela_exista,-00955);

begin

execute immediate 'create table preturi(nr number)';

execute immediate 'insert into preturi values (100)';

exception

when tabela_exista then

execute immediate 'drop table preturi';

DBMS_OUTPUT.PUT_LINE('Tabela a fost stearsa!');

end;

/

PL/SQL procedure successfully completed.

Tabela a fost stearsa!
```

-- CURSORI IMPLICITI

-- 1. Sa se creasca pretul pieselor ce costa mai putin de 100 u.m. cu 5 u.m.

```
declare

begin

update Piesa set Pret=Pret+5 where pret<100;

if sql%rowcount>0 then
```

Isvoranu Alexandru-Florian

```
dbms_output.put_line('S-au marit preturile!');

else

dbms_output.put_line('Nu s-au marit preturile!');

end if;

exception when others then

dbms_output.put_line('Nu s-a marit niciun pret.'||sqlerrm);

end;

/
```

```
PL/SQL procedure successfully completed.
```

```
S-au marit preturile!
```

--2. Sa se sterga angajatii prezenti in firma inainte de anul 2019.

```
declare

begin

delete from Angajat

where data_angajarii<to_date('01-01-2019','dd-mm-yyyy');

if sql%rowcount>0 then

dbms_output.put_line('S-au sters angajatii!');

else

dbms_output.put_line('Nu s-au sters angajatii!');

end if;

EXCEPTION when others then

dbms_output.put_line('Nu s-a sters niciun angajat!'||sqlerrm);

end;

/
```

```
PL/SQL procedure successfully completed.
```

```
S-au sters angajatii!
```

Ivoranu Alexandru-Florian

--CURSORI EXPLICITI

--1. Sa se afiseze pentru primele 5 reparatii defectiunea si pretul final.

declare

cursor c is select defectiune, pret_total

from reparatie

order by pret_total;

r c%ROWTYPE;

begin

if not c%isopen then

open c;

end if;

loop

fetch c into r;

exit when c%notfound or c%rowcount>5;

dbms_output.put('Reparatia'||c%rowcount);

dbms_output.put(':||r.defectiune||', pret total:'||r.pret_total);

dbms_output.new_line;

end loop;

close c;

end;

/

```
| Reparatia 1: Schimb placute, pret total: 250
| Reparatia 2: Schimb filtre, pret total: 380
| Reparatia 3: Schimb amortizor, pret total: 390
| Reparatia 4: Schimb filtre, pret total: 400
| Reparatia 5: Schimb filtre, pret total: 400
```

--2. Sa se afiseze pentru fiecare angajat numarul de zile de cand e prezent in firma

declare

cursor c is select nume_a, trunc((SYSDATE-data_angajarii),0) AS numar_zile

```
from angajat;
r c%ROWTYPE;

begin
if not c%isopen then
open c;
end if;
loop
fetch c into r;
exit when c%notfound;
dbms_output.put('Angajatul'||c%rowcount);
dbms_output.put(':||r.nume_a||', este prezent de'||r.numar_zile||' zile');
dbms_output.new_line;
end loop;
close c;
end;
/
```

```
Angajatul 1: Zoe Staruiala, este prezent de 1447 zile
Angajatul 2: Pavel Andreeescu, este prezent de 1209 zile
Angajatul 3: Eugen Rusu, este prezent de 1442 zile
Angajatul 4: Cosmin Barbu, este prezent de 999 zile
Angajatul 5: Bianca Ciocirlan, este prezent de 943 zile
Angajatul 6: Alberto Buse, este prezent de 836 zile
Angajatul 7: Dorin Stan, este prezent de 805 zile
Angajatul 8: Dragoa Calinescu, este prezent de 1257 zile
Angajatul 9: Iulian Popa, este prezent de 971 zile
Angajatul 10: Iulian Fratila, este prezent de 1290 zile
```

--3. Sa se afiseze pentru fiecare marca numarul de intrari in Service;

```
declare
cursor c is SELECT Marca, COUNT (*) as total
```

Ivoranu Alexandru-Florian

FROM MASINA

GROUP BY Marca;

```
r c%ROWTYPE;
begin
if not c%isopen then
open c;
end if;
loop
fetch c into r;
exit when c%notfound;
if r.total>1 then
dbms_output.put('Marca '|r.Marca| '| a intrat in service de '|r.total| '| ori.');
dbms_output.new_line;
else
dbms_output.put('Marca '|r.Marca| '| a intrat in service o data.');
dbms_output.new_line;
end if;
end loop;
close c;
end;
/
```

```
Marca VolksWagen a intrat in service o data.  
Marca BMW a intrat in service de 2 ori.  
Marca Mazda a intrat in service o data.  
Marca Nissan a intrat in service o data.  
Marca Seat a intrat in service o data.  
Marca Toyota a intrat in service o data.  
Marca Audi a intrat in service de 2 ori.  
Marca Dacia a intrat in service de 2 ori.  
Marca Mercedes-Benz a intrat in service o data.  
Marca Skoda a intrat in service de 2 ori.  
Marca Kia a intrat in service o data.
```

--Cursor explicit cu parametri:

--4. Afisati toate detaliile pieselor cu o anumita firma producatoare si un anumit model:

```
set serveroutput on;
```

```
DECLARE
```

```
CURSOR c_piese (v_firma VARCHAR2, v_model VARCHAR2) IS
```

```
    SELECT *
```

```
    FROM piesa
```

```
    WHERE firma_producatoare = v_firma
```

```
    AND model = v_model;
```

```
    v_cod_p piesa.cod_p%TYPE;
```

```
    v_firma_producatoare piesa.firma_producatoare%TYPE;
```

```
    v_model piesa.model%TYPE;
```

```
    v_pret piesa.pret%TYPE;
```

```
BEGIN
```

```
FOR r_piese IN c_piese('Motrio', 'Placute de frana') LOOP
```

```
    v_cod_p := r_piese.cod_p;
```

```
    v_firma_producatoare := r_piese.firma_producatoare;
```

```
    v_model := r_piese.model;
```

```
    v_pret := r_piese.pret;
```

```
    DBMS_OUTPUT.PUT_LINE('Cod piesa: ' || v_cod_p);
```

Isvoranu Alexandru-Florian

```
DBMS_OUTPUT.PUT_LINE('Firma producatoare: ' || v_firma_producatoare);
DBMS_OUTPUT.PUT_LINE('Model: ' || v_model);
DBMS_OUTPUT.PUT_LINE('Pret: ' || v_pret);
END LOOP;
END;
/
```

```
PL/SQL procedure successfully completed.
```

```
Cod piesa: 902
Firma producatoare: Motrio
Model: Placute de frana
Pret: 95
```

--FUNCTII

--1. Functia care afiseaza numarul total de reparatii efectuate într-un anumit an si luna:

```
CREATE OR REPLACE FUNCTION f_nr_reparatii_an_luna(v_an IN NUMBER, v_luna IN NUMBER) RETURN NUMBER
AS
```

```
v_nr_reparatii NUMBER;
BEGIN
  SELECT COUNT(*)
  INTO v_nr_reparatii
  FROM reparatie
  WHERE EXTRACT(YEAR FROM data_finalizare) = v_an
    AND EXTRACT(MONTH FROM data_finalizare) = v_luna;
  RETURN v_nr_reparatii;
END;
/
```

```
DECLARE
```

```
v_count NUMBER;
```

Ivoranu Alexandru-Florian

```
BEGIN  
    v_count := f_nr_reparatii_an_luna(2021, 4);  
    DBMS_OUTPUT.PUT_LINE('Numarul de reparatii in acea luna este: ' || v_count);  
END;  
  
/  
  
Function F_NR_REPARATII_AN_LUNA compiled  
  
Numarul de reparatii in acea luna este: 2
```

--2. Functia care afiseaza numarul total de reparatii efectuate în fiecare an:

```
CREATE OR REPLACE FUNCTION nr_reparatii_an(  
    anul IN NUMBER  
) RETURN NUMBER  
  
IS  
    nr_reparatii NUMBER;  
  
BEGIN  
    SELECT COUNT(*)  
    INTO nr_reparatii  
    FROM reparatie  
    WHERE EXTRACT(YEAR FROM data_finalizare) = anul;  
    RETURN nr_reparatii;  
  
END;  
  
/  
  
set serveroutput on;  
  
DECLARE  
    v_count NUMBER;  
  
BEGIN  
    v_count := nr_reparatii_an(2021);  
    DBMS_OUTPUT.PUT_LINE('Numarul de reparatii in acel an este: ' || v_count);  
END;
```

/

```
Function NR_REPARATII_AN compiled  
Numarul de reparatii in acel an este: 8
```

--3. Functia care calculeaza pretul pieselor pentru fiecare remediere:

```
CREATE OR REPLACE FUNCTION calc_pret_piese (p_cod_I IN Remediere.cod_I%TYPE)  
RETURN NUMBER  
IS  
    v_cantitate_piese Remediere.cantitate_piese%TYPE;  
    v_cod_p Piesa.cod_p%TYPE;  
    v_pret_piese NUMBER := 0;  
  
BEGIN  
    FOR c IN (SELECT cod_p, cantitate_piese FROM Remediere WHERE cod_I = p_cod_I) LOOP  
        v_cantitate_piese := c.cantitate_piese;  
        v_cod_p := c.cod_p;  
        SELECT pret INTO v_pret_piese FROM Piesa WHERE cod_p = v_cod_p;  
        v_pret_piese := v_pret_piese * v_cantitate_piese;  
    END LOOP;  
    RETURN v_pret_piese;  
END;  
/  
SELECT cod_I, calc_pret_piese(cod_I) AS pret_piese FROM Remediere;
```

	COD_P	PRET_PIESE
1	701	380
2	702	300
3	703	30
4	704	30
5	705	75
6	706	240
7	707	30
8	708	30
9	709	85
10	710	150
11	711	300
12	712	380
13	713	480
14	714	30
15	715	30
16	716	85
17	717	240

--PROCEDURI

--1. Procedura care actualizeaza cantitatea de piese disponibile în functie de o anumita cantitate utilizata într-o reparatie:

```

CREATE OR REPLACE PROCEDURE actualizeaza_cantitate_piese(
    cod_piesa IN NUMBER,
    cantitate_utilizata IN NUMBER
)
IS
BEGIN
    UPDATE Piesa
    SET Cantitate = Cantitate - cantitate_utilizata
    WHERE Cod_p = cod_piesa;
    COMMIT;
    DBMS_OUTPUT.PUT_LINE('Cantitatea piesei cu codul ' || cod_piesa || ' a fost actualizata cu succes.');
END;
/
BEGIN
    actualizeaza_cantitate_piese(905, 2);

```

```
END;  
/
```

```
Procedure ACTUALIZEAZA_CANTITATE_PIESE compiled  
Cantitatea piesei cu codul 905 a fost actualizata cu succes.
```

--2. Procedura care primește un cod de la o piesă și mărește prețul de lista cu 20 u.m:

```
drop procedure mreste_pret_categorie;  
create or replace procedure mreste_pret_categorie(p_cod number, p_cate OUT number) is  
begin  
update piesa set pret=pret+20  
where cod_p=p_cod;  
p_cate:=sql%rowcount;  
end mreste_pret_categorie;  
/  
set serveroutput on  
declare  
v_cate number;  
begin  
mreste_pret_categorie(901, v_cate);  
dbms_output.put_line('S-au modificate '|>v_cate);  
end;  
/
```

```
Procedure MARESTE_PRET_CATEGORIE compiled  
S-au modificate 1
```

--3. Procedura care afiseaza numele si prenumele angajatului care a efectuat cele mai multe reparatii într-un anumit an:

```
CREATE OR REPLACE PROCEDURE cel_mai_productiv_angajat(
    anul IN NUMBER
)
IS
    nume_angajat VARCHAR2(50);
BEGIN
    FOR record IN (
        SELECT a.nume_a, COUNT(*) as nr_reparatii
        FROM angajat a
        JOIN remediere r ON a.cod_a = r.cod_a
        JOIN reparatie rp ON r.cod_r = rp.cod_r
        WHERE EXTRACT(YEAR FROM rp.data_finalizare) = anul
        GROUP BY a.nume_a
        ORDER BY nr_reparatii DESC
    ) LOOP
        nume_angajat := record.nume_a;
        EXIT;
    END LOOP;
    DBMS_OUTPUT.PUT_LINE('Cel mai productiv angajat in ' || anul || ':' || nume_angajat);
END;
/
BEGIN
    cel_mai_productiv_angajat(2021);
END;
/
Procedure CEL_MAI_PRODUCTIV_ANGAJAT compiled
Cel mai productiv angajat in 2021: Eugen Rusu
```

--Declansatoare la nivel de rand:

--1. Crearea unui declansator dupa instructiunea UPDATE a tablei "Piesa",

--care sa verifice daca cantitatea de piese ramase în stoc este mai mica

--decât 5 si sa emita un mesaj de avertizare daca este adevarat:

```
CREATE OR REPLACE TRIGGER piesa_stoc
```

```
AFTER UPDATE ON Piesa
```

```
FOR EACH ROW
```

```
BEGIN
```

```
IF :NEW.Cantitate < 5 THEN
```

```
    DBMS_OUTPUT.PUT_LINE('Atentie! Cantitatea de piese ramasa în stoc este mai mica decât 5!');
```

```
END IF;
```

```
END;
```

```
/
```

```
UPDATE Piesa
```

```
SET Cantitate = 2
```

```
WHERE Cod_p = 913;
```

```
Trigger PIESA_STOC compiled
```

```
Atentie! Cantitatea de piese ramasa în stoc este mai mica decât 5!
```

--2. Crearea unui declansator dupa instructiunea INSERT a tablei "Piesa",

--care sa verifice daca numarul pieselor din depozit este mai mare sau

-- egal cu numarul pieselor care urmeaza sa fie inlocuite

```
CREATE OR REPLACE TRIGGER verificare_cantitate_piese
```

```
BEFORE INSERT ON Remediere
```

```
FOR EACH ROW
```

Ivoranu Alexandru-Florian

```
DECLARE
    cantitate_piese_disponibile Piesa.cantitate%TYPE;
BEGIN
    SELECT cantitate INTO cantitate_piese_disponibile
    FROM Piesa
    WHERE cod_p = :new.cod_p;
    IF cantitate_piese_disponibile < :new.cantitate_piese THEN
        DBMS_OUTPUT.PUT_LINE('Nu sunt suficiente piese disponibile pentru aceasta remediere!');
    END IF;
END;
/
insert into Remediere(cod_l,cod_r,cod_a,cod_p,cantitate_piese) values
(718,305,3,905,30);
Trigger VERIFICARE_CANTITATE_PIESE compiled
Nu sunt suficiente piese disponibile pentru aceasta remediere!
```

--Declansatoare la nivel de instructiune:

--1. Crearea unui declansator care sa verifice daca pretul total al reparatiei
--este mai mare de 500 si, in caz afirmativ, sa afiseze un mesaj.

```
CREATE OR REPLACE TRIGGER check_pret_total
AFTER INSERT ON Reparatie
FOR EACH ROW
BEGIN
    IF :new.pret_total > 500 THEN
        DBMS_OUTPUT.PUT_LINE('Este nevoie de factura!');
    END IF;
END;
/
```

Isvoranu Alexandru-Florian

```
set serveroutput on;
insert into Reparatie(cod_r,data_constatare,cod_m,defectiune,data_finalizare,pret_total) values
(314,to_date ('23-02-2023','dd-mm-yyyy'),503,'Schimb planetare',to_date ('25-02-2023','dd-mm-yyyy'),540);
```

```
Trigger CHECK_PRET_TOTAL compiled
```

```
Este nevoie de factura!
```

--2. Crearea unui declansator care afiseaza un mesaj corespunzator
--daca durata unei reparatii este mai mare de 3 zile.

```
CREATE OR REPLACE TRIGGER check_pret_total
```

```
AFTER INSERT ON Reparatie
```

```
FOR EACH ROW
```

```
BEGIN
```

```
IF :new.data_finalizare->:new.data_constatare > 3 THEN
```

```
DBMS_OUTPUT.PUT_LINE('Reparatie de lunga durata.');
```

```
END IF;
```

```
END;
```

```
/
```

```
insert into Reparatie(cod_r,data_constatare,cod_m,defectiune,data_finalizare,pret_total) values
(315,to_date ('02-05-2023','dd-mm-yyyy'),506,'Schimb radiator',to_date ('07-05-2023','dd-mm-yyyy'),670);
```

```
Trigger CHECK_PRET_TOTAL compiled
```

```
Reparatie de lunga durata.
```

--PACHET

```
CREATE OR REPLACE PACKAGE gestionare_reparatii AS  
-- Functie care afiseaza toate datele despre o reparatie  
FUNCTION afisare_reparatie(cod_reparatie IN NUMBER) RETURN REPARATIE%ROWTYPE;
```

```
-- Functie care afiseaza cantitatea unei piese  
FUNCTION afisare_cantitate_piesa(cod_piesa IN NUMBER) RETURN NUMBER;
```

--Procedura pentru a scadea pretul total al unei reparatii cu 10%, pe baza codului de reparatie primit ca parametru:

```
PROCEDURE scadere_pret_reparatie(v_cod_r IN NUMBER);
```

--Procedura pentru a creste numarul de reparatii ale unui client cu 1, pe baza codului de client primit ca parametru:

```
PROCEDURE creste_nr_reparatii_client(cod_cl IN NUMBER);
```

```
END gestionare_reparatii;
```

```
/
```

```
CREATE OR REPLACE PACKAGE BODY gestionare_reparatii AS
```

-- Functie care afiseaza toate datele despre o reparatie

```
FUNCTION afisare_reparatie(cod_reparatie IN NUMBER) RETURN REPARATIE%ROWTYPE IS  
reparatie_row REPARATIE%ROWTYPE;
```

```
BEGIN
```

```
SELECT * INTO reparatie_row FROM REPARATIE WHERE cod_r = cod_reparatie;
```

```
RETURN reparatie_row;
```

```
END;
```

-- Functie care afiseaza cantitatea unei piese

```
FUNCTION afisare_cantitate_piesa(cod_piesa IN NUMBER) RETURN NUMBER IS  
cantitate NUMBER(4);
```

```
BEGIN
```

Isvoranu Alexandru-Florian

```
SELECT cantitate INTO cantitate FROM Piesa WHERE cod_p = cod_piesa;  
RETURN cantitate;  
END;
```

--Procedura pentru a scadea pretul total al unei reparatii cu 10%, pe baza codului de reparatie primit ca parametru:

```
PROCEDURE scadere_pret_reparatie(v_cod_r IN NUMBER) IS  
    v_pret_total NUMBER(6);  
BEGIN  
    SELECT pret_total INTO v_pret_total FROM Reparatie WHERE cod_r = v_cod_r;  
    v_pret_total := v_pret_total * 0.9;  
    UPDATE Reparatie SET pret_total = v_pret_total WHERE cod_r = v_cod_r;  
    DBMS_OUTPUT.PUT_LINE('Pretul total a scazut cu 10%.' );  
END;
```

--Procedura pentru a creste numarul de reparatii ale unui client cu 1, pe baza codului de client primit ca parametru:

```
PROCEDURE creste_nr_reparatii_client(cod_cl IN NUMBER) IS  
BEGIN  
    UPDATE Client SET nr_reparatii = nr_reparatii + 1 WHERE cod_cl = cod_cl;  
    DBMS_OUTPUT.PUT_LINE('Numarul de reparatii a crescut cu 1.' );  
END;  
END gestionare_reparatii;  
/
```

```
set serveroutput on;  
DECLARE  
    reparatie_row REPARATIE%ROWTYPE;  
    cantitate NUMBER;  
BEGIN  
    reparatie_row := gestionare_reparatii.afisare_reparatie(307);
```

Isvoranu Alexandru-Florian

```
cantitate:=gestionare_reparatii.afisare_cantitate_piesa(912);

DBMS_OUTPUT.PUT_LINE('Reparatia '|| reparatie_row.cod_r || ':' || reparatie_row.data_constatare || ' -> ' ||
reparatie_row.data_finalizare || ',' || reparatie_row.pret_total || ' lei');

DBMS_OUTPUT.PUT_LINE('Cantitatea piesei este '| |cantitate);

END;

/

select * from reparatie;

select * from client;

EXECUTE gestionare_reparatii.scadere_pret_reparatie(302);

EXECUTE gestionare_reparatii.creste_nr_reparatii_client(106);
```

Package Body GESTIONARE_REPARATII compiled

Reparatia 307: 19-DEC-21 -> 20-DEC-21, 510 lei
Cantitatea piesei este 7

PL/SQL procedure successfully completed.

Pretul total a scazut cu 10%.

PL/SQL procedure successfully completed.

Numarul de reparatii a crescut cu 1.