

# Proiect EABD

Kotai Melinda-Antonela  
IE, an 3, grupa 1  
An universitar 2020-2021

## 1. Descriere generala si implementarea bazei de date

Trebuie sa se proiecteze o baza de date pentru o agentie de turism. Crescandu-i semnificativ popularitatea si numarul de clienti, agentia de turism s-a hotarat ca toate datele legate de activitatea pe care o desfasoara sa fie stocate in cadrul unei baze de date pentru o mai buna gestiune a acestora. Baza de date va fi folosita in cadrul unei aplicatii utilizata de angajatii firmei din toate departamentele. De exemplu angajatii din departamentul de vanzari vor putea adauga clienti noi si vanzari noi efectuate date care vor putea fi folosite pentru generarea unor rapoarte (ex. cele mai vandute pachete, angajatul care a efectuat cele mai multe vanzari), angajatii din departamentul de marketing vor putea adauga pachete noi si vor folosi datele inregistrate in baza de date pentru a putea face rapoarte legate de pachetele preferate de clienti informatie care ii va ajuta in desfasurarea optima a activitatii lor si in luarea deciziilor legate de pachetele viitoare comercializate, angajatii din departamentul de resurse umane vor putea adauga angajati noi in baza de date si vor putea actualize date legate de acestia iar angajatii din departamentul financiar-contabil vor putea utiliza datele legate de vanzarile efectuate pentru a-si facilita munca.

Agentia trebuie sa tina evidenta tuturor clientilor care au apelat la serviciile acesteia, astfel pe langa datele lor personale, de identificare, trebuie stocate si date de contact pentru ca firma sa poata lua legatura cu ei cu scopul de a le comunica noi oferte, detalii legate de pachete achizitionate sau eventuale modificari legate de acestea care pot aparea in timp, a pachetelor de vacant pe care le vand (toate detaliile necesare legate de acestea) si a angajatilor (datele personale si de contact, salariul, data angajarii, departamentul de care apartin, superiorul direct al fiecarui angajat, o poza pentru fiecare angajat).

In urma analizei cerintelor s-a creat o baza de date care a fost normalizata.

### CREAREA SI POPULAREA TABELELOR:

```
CREATE TABLE tari_p ( id_tara NUMBER(2) CONSTRAINT tari_p_pk  
PRIMARY KEY,  
denumire VARCHAR2(20) CONSTRAINT tari_p_den_uk UNIQUE  
CONSTRAINT  
tari_denumire_nn NOT NULL);
```

```
INSERT INTO tari_p VALUES(11,'Hawaii');
INSERT INTO tari_p VALUES(12,'Italia');
INSERT INTO tari_p VALUES(13,'Grecia');
INSERT INTO tari_p VALUES(14,'Frant');
INSERT INTO tari_p VALUES(15,'Egipt');
INSERT INTO tari_p VALUES(16,'Romania');
INSERT INTO tari_p VALUES(17,'Japonia');
```

```
CREATE TABLE destinatii_p (id_destinatie NUMBER(2) CONSTRAINT
destinatii_p_pk PRIMARY KEY, denumire VARCHAR2(20) CONSTRAINT
destinatii_p_denumire_nn NOT NULL, id_tra NUMBER(2) CONSTRAINT
destinatii_fk REFERENCES tari_p(id_tara) CONSTRAINT destinatii_fk_nn
NOT
NULL);
```

```
INSERT INTO destinatii_p VALUES(21, 'Honolulu', 11);
INSERT INTO destinatii_p VALUES(22, 'Maui', 11);
INSERT INTO destinatii_p VALUES(23, 'Roma', 12);
INSERT INTO destinatii_p VALUES(24, 'Atena', 13);
INSERT INTO destinatii_p VALUES(25, 'Creta', 13);
INSERT INTO destinatii_p VALUES(26, 'Giza', 15);
INSERT INTO destinatii_p VALUES(27, 'Cair', 15);
INSERT INTO destinatii_p VALUES(28, 'Florenta', 12);
INSERT INTO destinatii_p VALUES(29, 'Paris', 14);
INSERT INTO destinatii_p VALUES(30, 'Nisa', 14);
INSERT INTO destinatii_p VALUES(31, 'Constanta', 16);
INSERT INTO destinatii_p VALUES(32, 'Mamaia', 16);
INSERT INTO destinatii_p VALUES(33, 'Tokyo', 17);
INSERT INTO destinatii_p VALUES(34, 'Osaka', 17);
```

```
CREATE TABLE departamente_p(id_departament NUMBER(2)
CONSTRAINT departamente_p_pk PRIMARY KEY, denumire
VARCHAR2(20) CONSTRAINT departamente_p_denumire_nn NOT
NULL);
```

```
INSERT INTO departamente_p VALUES(31, 'Vanzari');
INSERT INTO departamente_p VALUES(32, 'Marketing');
INSERT INTO departamente_p VALUES(33, 'Financiar-Contabil');
INSERT INTO departamente_p VALUES(34, 'Resurse Umane');
```

```
CREATE TABLE pachete_p(id_pachet NUMBER(4) CONSTRAINT
pachete_p_pk PRIMARY KEY, nr_zile NUMBER(2) CONSTRAINT
pachete_p_nr_zile_nn NOT NULL, servicii VARCHAR2(20) CONSTRAINT
pachete_p_servicii_nn NOT NULL, hotel VARCHAR2(25) CONSTRAINT
pachete_p_hotel_nn NOT NULL, pret NUMBER(4) CONSTRAINT
pachete_p_pret_nn NOT NULL , tip_transport VARCHAR2(10)
CONSTRAINT pachete_p_tip_transport_nn NOT NULL, aeroport_plecare
VARCHAR2(45), aeroport_destinatie VARCHAR2(45), clasa_a
NUMBER(1), bagaj_cala NUMBER(2), nr_escale NUMBER(1),
gara_plecare VARCHAR2(45), gara_destinatie VARCHAR2(45), clasa_t
NUMBER(1), autogara_plecare VARCHAR2(45), autogara_destinatie
VARCHAR2(45), id_dte NUMBER(2) CONSTRAINT pachete_p_id_dte_fk
REFERENCES destinatii_p(id_destinatie) CONSTRAINT
pachete_p_id_dte_nn NOT NULL, CHECK(nr_zile>0 AND pret>0 AND
clasa_a>0 AND clasa_a<=3 AND clasa_t>0 AND clasa_t<=3),
CHECK(((tip_transport='aerian' AND aeroport_plecare is not null and
aeroport_destinatie is not null and clasa_a is not null and gara_plecare is
null and gara_destinatie is null and clasa_t is null and autogara_plecare is
null and autogara_destinatie is null)OR(tip_transport='feroviar' and
aeroport_plecare is null and aeroport_destinatie is null and clasa_a is null
and bagaj_cala is null and nr_escale is null and gara_plecare is not null
and gara_destinatie is not null and clasa_t is not null and autogara_plecare
is null and autogara_destinatie is null)OR(tip_transport='auto' and
aeroport_plecare is null and aeroport_destinatie is null and clasa_a is null
and bagaj_cala is null and nr_escale is null and gara_plecare is null and
gara_destinatie is null and clasa_t is null and autogara_plecare is not null
and autogara_destinatie is not null))));
```

```

INSERT INTO pachete_p VALUES(1121, 5, 'all inclusive', 'Waikiki Grand
Hotel',
2500, 'aerian', 'Aeroportul Internațional Avram Iancu Cluj', 'Aeroport
International
Honolulu',2, 25, 2,"","","",21);
INSERT INTO pachete_p VALUES(1122, 5, 'mic de jun',' Wailuku', 1700,
'aerian',
'Aeroportul Internațional Avram Iancu Cluj', 'Kahului Airport',2,"","","","",
22);
INSERT INTO pachete_p VALUES(1123, 7, 'demipensiune','San
Anselmo',1300,
'auto',"","","","","", 'Autogara Cluj-Napoca', 'Autogara Tiburtina', 23);
INSERT INTO pachete_p VALUES(1124, 4, 'all inclusive' , 'Luxor', 850,
'aerian',
'Aeroportul Internațional Avram Iancu Cluj', 'Aeroportul International Atena',
1,"","","","","", 24);
INSERT INTO pachete_p VALUES(1125, 5, 'demipensiune', 'Paradis', 800,
'aerian',
'Aeroportul Internațional Avram Iancu Cluj', 'Aeroportul international
Heraklion',
3, 25,"","","","","", 25);
INSERT INTO pachete_p VALUES(1126, 8, 'mic de jun', 'Amarante', 2200,
'aerian',
'Aeroportul Internațional Avram Iancu Cluj', 'Sphinx Airport' ,2, 20, 1,"","","","",
26);
INSERT INTO pachete_p VALUES(1127, 3, 'all inclusive', 'Atlantis',1700,
'aerian','Aeroportul Internațional Avram Iancu Cluj', 'Aeroportul International
Cairo', 1, "","","","","", 27);
INSERT INTO pachete_p VALUES(1128, 3, 'demipensiune', 'Villa Agape',
1100,
'auto',"","","","","", 'Autogara Cluj-Napoca','Valencia Central Bus Station',
28);
INSERT INTO pachete_p VALUES(1129, 7, 'all inclusive', 'Jardin de
Villiers', 2000,
'auto', "","","","","", 'Autogara Cluj-Napoca', 'Autogara Porte de Montreuil',
29);
INSERT INTO pachete_p VALUES(1130, 7, 'demipensiune', 'Charmant' ,
2000,
'aerian', 'Aeroportul Internațional Avram Iancu Cluj', 'Aeroportul Coasta de
Azur

```

```

Nisa', 2,30,"","","", 30);
INSERT INTO pachete_p VALUES(1131, 4, 'mic de jun', 'Orcagna',
1000,'aerian',
'Aeroportul Internațional Avram Iancu Cluj', 'Aeroportul Amerigo Vespucci',
2,"","","","", 28);
INSERT INTO pachete_p VALUES(1132, 7, 'demipensiune', 'Splendid' ,
1000,
'feroviar', "","","",'Gara Cluj-Napoca', 'Gara Constanta', 1,"", 31);

```

```

CREATE TABLE clienti_p (id_client NUMBER(4) PRIMARY KEY, CNP
CHAR(13) NOT NULL UNIQUE, nume VARCHAR2(20) NOT NULL,
prenume VARCHAR2(20) NOT NULL,telefon CHAR(10) NOT NULL
UNIQUE,email VARCHAR2(30) NOT NULL UNIQUE, adresa
VARCHAR2(60) NOT NULL);

```

```

INSERT INTO clienti_p
VALUES(10,'2960508149828','Pop','Alexandra','074515413',
'pop_alexandra96@yahoo.com', 'Covasna Str.Unirii nr.21');
INSERT INTO clienti_p
VALUES(11,'1950428106476','Sima','Mihai','0761923074',
'sima_m95@yahoo.com','Buzau Str. Garoafelor nr. 33 bloc 42 ap.41');
INSERT INTO clienti_p
VALUES(12,'2970705426060','Anton','Maria','0747342153',
'anton.maria@yahoo.com','Bucuresti Str. Florarilor nr. 4');
INSERT INTO clienti_p VALUES(13,'1910115191256','Ionescu','Adrian',
'0762154320','ionescu_adr91@yahoo.com', 'Harghita Str. Ferdinand
nr.15');
INSERT INTO clienti_p VALUES(14,'2861019302387', 'Anghel','Laura',
'0746342020','laura_anghel123@yahoo.com','Satu Mare Str. Bosfor nr.
11');
INSERT INTO clienti_p
VALUES(15,'1940906082791','Pancu','Ioan','0754432915',
'pancu.ioan4@yahoo.com','Brasov Str. Viitorului nr. 25');
INSERT INTO clienti_p VALUES(16,'1980207121475','Popescu','Bogdan'
,'0752126982','bogdan_popescu1@yahoo.com' , 'Cluj-Napoca Str.
Bucuresti nr. 74
bloc 2 ap. 22');

```

```

CREATE TABLE angajati_p(id_angajat NUMBER(3) CONSTRAINT
angajati_p_pk PRIMARY KEY,nume VARCHAR(20) CONSTRAINT
angajati_p_nume_nn NOT NULL,prenume VARCHAR(20) CONSTRAINT
angajati_p_prenume_nn NOT NULL,telefon CHAR(10) CONSTRAINT
angajati_p_telefon_nn NOT NULL CONSTRAINT angajati_p_tel_uk
UNIQUE,adresa VARCHAR(50) CONSTRAINT angajati_p_adresa_nn
NOT NULL,data_angajarii DATE CONSTRAINT
angajati_p_data_angajarii_nn NOT NULL,salariu NUMBER(4)
CONSTRAINT angajati_p_salariu_nn NOT NULL,id_dpt NUMBER(2)
CONSTRAINT angajati_p_fk1 REFERENCES
departamente_p(id_departament) CONSTRAINT angajati_p_fk1_nn NOT
NULL,id_mnr NUMBER(3) CONSTRAINT angajati_p_fk2 REFERENCES
angajati_p(id_angajat));

```

```

INSERT INTO angajati_p VALUES(111,'Sima','Sorin','0744103530','Cluj-
Napoca Str.
Muresului nr. 21','22-Oct-2010', 4500, 31, 112);
INSERT INTO angajati_p VALUES(112, 'Pancu', 'Alexandru',
'0734333511', 'Cluj-
Napoca Str. Oltului nr. 12 bl. 2 ap. 40', '17-May-2009', 6000, 31, 118);
INSERT INTO angajati_p VALUES(113,'Brandusa', 'Ionela','0722195400',
'Cluj-Napoca
Str. Bucuresti nr. 14 bl. 2 ap. 11', '19-Mar-2014', 3500, 31,112);
INSERT INTO angajati_p
VALUES(114,'Constantin','Marian','0761351102','Cluj-
Napoca Str. Muresului nr.12','13-Sep-2013', 3000, 32,119);
INSERT INTO angajati_p
VALUES(115,'Pop','Anamaria','0712345678','Cluj-Napoca Str.
Scortariilor nr. 4', '4-Feb-2018', 3200, 31, 112);
INSERT INTO angajati_p VALUES(116,'Ionescu', 'Vasile', '0722233300',
'Cluj-Napoca
Str. Horea nr. 7 bl. 3 ap. 11', '5-Dec-2009', 4100, 33, 118);
INSERT INTO angajati_p VALUES(117, 'Popa','Andrada', '0755522200',
'Cluj-Napoca
Str. Sesului nr. 5','1-Jan-2010', 4200, 34,118);
INSERT INTO angajati_p VALUES(118 ,'Bostan'
,'Emanuela','0723455555', 'Cluj-

```

```

Napoca Str. Horea nr. 17 bl. 1 ap. 12', '2-Feb-2009', 6300, 31,");
INSERT INTO angajati_p VALUES(119
,'Bostan','Cosmin','0742333333','Cluj-Napoca
Str. Lacul Rosu nr. 11 bl. 2 ap. 23','12-Mar-2009', 5000, 32, 118);
INSERT INTO angajati_p VALUES(120, 'Pop', 'Cristian', '0723835460',
'Cluj-Napoca Str.
Horea nr. 32 bl. 2 ap. 23', '3-Jul-2009', 4200, 32, 119);
INSERT INTO angajati_p VALUES(121, 'Mare', 'David', '0766554433',
'Cluj-Napoca Str.
Bucuresti nr.2 bl. 2 ap. 2', '6-Dec-2010', 4000, 34, 117);
INSERT INTO angajati_p VALUES(122, 'Paul', 'Maria', '0766557788', 'Cluj-
Napoca Str.
Fabricii nr. 50', '19-Nov-2010', 4000, 33, 116);

```

```

CREATE TABLE vanzari_p(nr_factura NUMBER(4), data_facturare DATE
CONSTRAINT vanzari_p_data_facturare_nn NOT NULL, data_plecare
DATE CONSTRAINT vanzari_p_data_plecare_nn NOT NULL,nr_persoane
NUMBER(2) CONSTRAINT vanzari_p_nr_persoane_nn NOT NULL
CONSTRAINT vanzari_p_nr_persoane_poz
CHECK(nr_persoane>0),id_pct NUMBER(4) CONSTRAINT vanzari_p_fk1
REFERENCES pachete_p(id_pachet),id_cet NUMBER(2) CONSTRAINT
vanzari_p_fk2 REFERENCES clienti_p(id_client) CONSTRAINT
vanzari_p_fk2_nn NOT NULL,id_agt NUMBER(3) CONSTRAINT
vanzari_p_fk3 REFERENCES angajati_p(id_angajat) CONSTRAINT
vanzari_p_fk3_nn NOT NULL, CONSTRAINT vanzari_p_pk PRIMARY
KEY(nr_factura,id_pct));

```

```

INSERT INTO vanzari_p VALUES(1111,'22-Jun-2019','5-Jul-2019', 2,
1131, 10,118);
INSERT INTO vanzari_p VALUES(1112,'1-May-2019','1-Jun-2019',2
,1121,12,112);
INSERT INTO vanzari_p VALUES(1113,'1-Mar-2019', '11-May-
2019',2,1128,14,112);
INSERT INTO vanzari_p VALUES(1114,'10-Jun-2019', '5-Aug-2019',3,
1131, 11,118);
INSERT INTO vanzari_p VALUES(1114,'10-Jun-2019' ,'20-Aug-2019',3
,1121,11,118);

```



```
INSERT INTO vanzari_p VALUES(1115,'3-Sep-2019','10-Oct-2019',2,
1129,13, 113);
INSERT INTO vanzari_p VALUES(1116,'1-May-2019','1-Jun-2019',3 ,
1122, 16,111);
INSERT INTO vanzari_p VALUES(1117,'10-Mar-2019','11-May-2019',2,
1124,12, 118);
INSERT INTO vanzari_p VALUES(1118,'1-Jul-2019', '20-Jul-2019',1,
1132,11, 113);
INSERT INTO vanzari_p VALUES(1119, '1-May-2019','1-Jun-2019' ,2,
1125,15, 112);
INSERT INTO vanzari_p VALUES(1119, '1-May-2019', '1-Aug-2019',2
,1129,15,112);
INSERT INTO vanzari_p VALUES(1120,'10-Sep-2019', '1-Oct-2019',1,
1123,13, 118);
INSERT INTO vanzari_p VALUES(1121,'1-Jul-2019','1-Aug-2019',4 , 1125,
12,118);
INSERT INTO vanzari_p VALUES(1122, '3-Jun-2019','25-Jun-2019',2
,1127,10,111);
INSERT INTO vanzari_p VALUES(1123, '10-Jul-2019','10-Sep-
2019',2,1131,10,112);
INSERT INTO vanzari_p VALUES(1124,'10-May-2019','15-Jul-
2019',2,1127,11,115);
INSERT INTO vanzari_p VALUES(1125,'3-Aug-2019','5-Sep-2019',2,
1130,12,111);
```

# Schema bazei de date normalizata

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
PACHETE_P	ID_PACHET	NUMBER	-	4	0	1	-	-	-
	NR_ZILE	NUMBER	-	2	0	-	-	-	-
	SERVICII	VARCHAR2	20	-	-	-	-	-	-
	HOTEL	VARCHAR2	25	-	-	-	-	-	-
	PRET	NUMBER	-	4	0	-	-	-	-
	TIP_TRANSPORT	VARCHAR2	10	-	-	-	-	-	-
	AEROPORT_PLECARE	VARCHAR2	45	-	-	-	✓	-	-
	AEROPORT_DESTINATIE	VARCHAR2	45	-	-	-	✓	-	-
	CLASA_A	NUMBER	-	1	0	-	✓	-	-
	BAGAJ_CALA	NUMBER	-	2	0	-	✓	-	-
	NR_ESCALE	NUMBER	-	1	0	-	✓	-	-
	GARA_PLECARE	VARCHAR2	45	-	-	-	✓	-	-
	GARA_DESTINATIE	VARCHAR2	45	-	-	-	✓	-	-
	CLASA_T	NUMBER	-	1	0	-	✓	-	-
	AUTOGARA_PLECARE	VARCHAR2	45	-	-	-	✓	-	-
	AUTOGARA_DESTINATIE	VARCHAR2	45	-	-	-	✓	-	-
	ID_OTE	NUMBER	-	2	0	-	✓	-	-

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
CLIENTI_P	ID_CLIENT	NUMBER	-	2	0	1	-	-	-
	CNP	CHAR	13	-	-	-	-	-	-
	NUME	VARCHAR2	20	-	-	-	-	-	-
	PRENUME	VARCHAR2	20	-	-	-	-	-	-
	TELEFON	CHAR	10	-	-	-	-	-	-
	EMAIL	VARCHAR2	30	-	-	-	-	-	-
	ADRESA	VARCHAR2	50	-	-	-	-	-	-

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
VANZARI_P	NR_FACTURA	NUMBER	-	4	0	1	-	-	-
	DATA_FACTURARE	DATE	7	-	-	-	-	-	-
	DATA_PLECARE	DATE	7	-	-	-	-	-	-
	NR_PERSOANE	NUMBER	-	2	0	-	-	-	-
	ID_PCT	NUMBER	-	4	0	2	-	-	-
	ID_CET	NUMBER	-	2	0	-	-	-	-
	ID_AGT	NUMBER	-	3	0	-	-	-	-

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
DESTINATII_P	ID_DESTINATIE	NUMBER	-	2	0	1	-	-	-
	DENUMIRE	VARCHAR2	20	-	-	-	-	-	-
	ID_TRA	NUMBER	-	2	0	-	-	-	-

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
TARI_P	ID_TARA	NUMBER	-	2	0	1	-	-	-
	DENUMIRE	VARCHAR2	20	-	-	-	-	-	-

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
ANGAJATI_P	ID_ANGAJAT	NUMBER	-	3	0	1	-	-	-
	NUME	VARCHAR2	20	-	-	-	-	-	-
	PRENUME	VARCHAR2	20	-	-	-	-	-	-
	TELEFON	CHAR	10	-	-	-	-	-	-
	ADRESA	VARCHAR2	50	-	-	-	-	-	-
	DATA_ANGAJARI	DATE	7	-	-	-	-	-	-
	SALARIU	NUMBER	-	4	0	-	-	-	-
	ID_DPT	NUMBER	-	2	0	-	-	-	-
	ID_MNR	NUMBER	-	3	0	-	✓	-	-

Object Type	TABLE ⓘ	Object	DEPARTAMENTE_P ⓘ						
Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
DEPARTAMENTE_P	ID_DEPARTAMENT	NUMBER	-	2	0	1	-	-	-
	DENUMIRE	VARCHAR2	20	-	-	-	-	-	-

In urma implementarii bazei de date au aparut noi cerinta si anume: pastrarea unei poze pentru fiecare angajat, a unei copii de buletin pentru fiecare client si a reducerilor valabile pentru clienti daca acestea exista. Deci este necesara adaugarea a trei coloane noi, in tabela angajati se va adauga coloana "poza" de tipul BLOB iar in tabela clienti se va adauga coloana "copie\_buletin" de tipul BLOB si coloana "discount" de tipul NUMBER.

Cod:

```
ALTER TABLE angajati_p ADD poza BLOB DEFAULT empty_blob();
```

```
ALTER TABLE clienti_p ADD copie_buletin BLOB DEFAULT  
empty_blob();
```

```
ALTER TABLE clienti_p ADD discount NUMBER(3,2) DEFAULT 0.0;
```

## 2. Creare package, declansatori, secvente

Initial vom crea 3 secvente care ne vor ajuta in cadrul operatiilor de inserare din viitoarele proceduri. Vom crea secvente pentru tabela clienti, angajati si departamente:

```
CREATE SEQUENCE angajati_angid_seq  
    INCREMENT BY 1  
    START WITH 125  
    MAXVALUE 999  
    NOCACHE  
    NOCYCLE;
```

```
CREATE SEQUENCE clienti_cliid_seq  
    INCREMENT BY 1  
    START WITH 17  
    MAXVALUE 9999  
    NOCACHE  
    NOCYCLE;
```

```
CREATE SEQUENCE departamente_deptid_seq  
    INCREMENT BY 1  
    START WITH 35  
    MAXVALUE 99  
    NOCACHE  
    NOCYCLE;
```

Pentru satisfacerea nevoilor firmei este necesara crearea unor functii si proceduri care ulterior vor fi incluse intr-un package pentru o mai buna organizare.

1. Este nevoie de o procedura careia, transmitandu-i o valoare numerica (valoarea cu care sa scada pretul) si un numar de vanzari minim, aceasta sa scada pretul pachetelor care au un numar de vanzari mai mic

decat numarul minim transmis, cu valoarea specificata ca parametru. Aceasta procedura va fi folosita de firma pentru a afla pachetele neprofitabile si a le scadea pretul cu o anumita valoare pentru a fi mai atractive pentru clienti.

Vom afla pachetele care au un numar de vanzari mai mic sau egal cu 1:

```
SELECT id_pct, nr_vanzari from (SELECT id_pct, count(id_pct) as  
nr_vanzari from vanzari_p group by id_pct) WHERE nr_vanzari<=1;
```

ID_PCT	NR_VANZARI
1130	1
1123	1
1122	1
1132	1
1124	1
1128	1

Vom crea o procedura care, transmitandu-i id-ul unui pachet si o valoare, scade pretul pachetului cu valoarea transmisa.

```
CREATE OR REPLACE PROCEDURE scade_pret_pachet(id IN  
pachete_p.id_pachet%TYPE, valoare IN pachete_p.pret%TYPE) IS  
pret_nou pachete_p.pret%TYPE;  
pret_vechi pachete_p.pret%TYPE;  
BEGIN  
SELECT pret INTO pret_vechi FROM pachete_p WHERE id_pachet=id;  
pret_nou:=pret_vechi-valoare;  
UPDATE pachete_p SET pret=pret_nou WHERE id_pachet=id;  
END;
```

Cu ajutorul procedurii create anterior si cu ajutorul **unui cursor cu un parametru** vom crea procedura necesara firmei:

```
CREATE OR REPLACE PROCEDURE  
scad_pret_pachet_nepopular(nr_vanzari IN NUMBER, valoare IN  
pachete_p.pret%TYPE) IS  
CURSOR pachete_nepopulare(nr_v NUMBER) IS  
SELECT id_pct, nr_vanzari from (SELECT id_pct, count(id_pct) as  
nr_vanzari from vanzari_p group by id_pct) WHERE nr_vanzari<=nr_v;  
BEGIN
```

```

FOR pachet IN pachete_nepopulare(nr_vanzari) LOOP
scade_pret_pachet(pachet.id_pct,valoare);
END LOOP;
END;

```

Vom testa procedura prin selectarea pachetelor inainte si dupa rulara  
codului care apeleaza procedura cu nr minim de vanzari 1 si valoarea care  
sa se scada 10:

```
select * from pachete_p;
```

```

BEGIN
scad_pret_pachet_nepopular(1,10);
END;

```

```
select * from pachete_p;
```

1121	5	all inclusive	Waikiki Grand Hotel	2500	aerian	Aeroportul International Avram Iancu Cluj	Aeroport International Honolulu	2	25	2	-	-	-	-	-
1122	5	mic de jun	Wailuku	1700	aerian	Aeroportul International Avram Iancu Cluj	Kahului Airport	2	-	-	-	-	-	-	-
1123	7	demipensiune	San Anselmo	1300	auto	-	-	-	-	-	-	-	-	Autogara Cluj-Napoca	Autogara Tiburtina
1126	8	mic de jun	Amarante	2200	aerian	Aeroportul International Avram Iancu Cluj	Sphinx Airport	2	20	1	-	-	-	-	-
1127	3	all inclusive	Atlantis	1700	aerian	Aeroportul International Avram Iancu Cluj	Aeroportul International Cairo	1	-	-	-	-	-	-	-
1128	3	demipensiune	Villa Agape	1100	auto	-	-	-	-	-	-	-	-	Autogara Cluj-Napoca	Valencia Central Bus Station
1129	7	all inclusive	Jardin de Villiers	2000	auto	-	-	-	-	-	-	-	-	Autogara Cluj-Napoca	Autogara Porte de Montreuil
1130	7	demipensiune	Charmant	2000	aerian	Aeroportul International Avram Iancu Cluj	Aeroportul Coasta de Azur Nisa	2	30	-	-	-	-	-	-
1131	4	mic de jun	Orcagna	1000	aerian	Aeroportul International Avram Iancu Cluj	Aeroportul Amerigo Vespucci	2	-	-	-	-	-	-	-
1132	7	demipensiune	Splendid	1000	feroviar	-	-	-	-	-	Gara Cluj-Napoca	Gara Constanta	1	-	-

ID_PACHET	NR_ZILE	SERVICIU	HOTEL	PRET	TIPI_TRANSPORT	AEROPORTUL_FLECARE	AEROPORTUL_DESTINATIE	CLASA_A	BAZOU_CLASA	NR_ESCALE	ORAR_FLECARE	ORAR_DESTINATIE	CLASA_T	AUTOGARA_FLECARE	AUTOGARA_DESTINATIE	ID_
1121	5	all inclusive	Waikiki Grand Hotel	2500	aerian	Aeroportul International Avram Iancu Cluj	Aeroport International Honolulu	2	25	2	-	-	-	-	-	21
1122	5	mic de jun	Wailluku	1690	aerian	Aeroportul International Avram Iancu Cluj	Kahului Airport	2	-	-	-	-	-	-	-	22
1123	7	demipensiune	San Anselmo	1290	auto	-	-	-	-	-	-	-	-	Autogara Cluj-Napoca	Autogara Tiburtina	23
1126	8	mic de jun	Amarante	2200	aerian	Aeroportul International Avram Iancu Cluj	Sphinx Airport	2	20	1	-	-	-	-	-	26
1127	3	all inclusive	Atlantis	1700	aerian	Aeroportul International Avram Iancu Cluj	Aeroportul International Cairo	1	-	-	-	-	-	-	-	27
1128	3	demipensiune	Villa Agape	1090	auto	-	-	-	-	-	-	-	-	Autogara Cluj-Napoca	Valencia Central Bus Station	28
1129	7	all inclusive	Jardin de Villiers	2000	auto	-	-	-	-	-	-	-	-	Autogara Cluj-Napoca	Autogara Porte de Montreuil	29
1130	7	demipensiune	Charmant	1990	aerian	Aeroportul International Avram Iancu Cluj	Aeroportul Coasta de Azur Nisa	2	30	-	-	-	-	-	-	30
1131	4	mic de jun	Orcagna	1000	aerian	Aeroportul International Avram Iancu Cluj	Aeroportul Amerigo Vespucci	2	-	-	-	-	-	-	-	28
1132	7	demipensiune	Splendid	990	feroviar	-	-	-	-	-	Gara Cluj-Napoca	Gara Constanta	1	-	-	31

Se observa scaderea pretului la pachetele obtinute la prima selectare: 1130, 1122, 1123, 1132 etc.

2. In fiecare luna, firma acorda on reducere de 10% la urmatoarea achizitie clientului cel mai fidel, adica clientului care are cele mai multe cumparaturi efectuate. Pentru a facilita acest proces se va crea o procedura, care folosindu-se de o functie care afla care este clientul cel mai fidel, va introduce in baza de date reducerea pentru clientul aflat.

Initial nici un client nu are reducere valabila in coloana discount;

select \* from clienti\_p;

ID_CLIENT	CNP	NUME	PRENUME	TELEFON	EMAIL	ADRESA	COPIE_BULETIN	DISCOUNT
11	1950428106476	Sima	Mihai	0761923074	sima_m95@yahoo.com	Buzau Str. Garoafelor nr. 33 bloc 42 ap.41	[unsupported data type]	0
12	2970705426060	Anton	Maria	0747342153	anton.maria@yahoo.com	Bucuresti Str. Floranilor nr. 4	[unsupported data type]	0
13	1910115191256	Ionescu	Adrian	0762154320	ionescu_adr91@yahoo.com	Harghita Str. Ferdinand nr.15	[unsupported data type]	0
14	2861019302387	Anghel	Laura	0746342020	laura_angel123@yahoo.com	Satu Mare Str. Bosfor nr. 11	[unsupported data type]	0
15	1940906082791	Pancu	Ioan	0754432915	pancu.ioan4@yahoo.com	Brasov Str. Viitorului nr. 25	[unsupported data type]	0
16	1980207121475	Popescu	Bogdan	0752126982	bogdan_popescu1@yahoo.com	Cluj-Napoca Str. Bucuresti nr. 74 bloc 2 ap. 22	[unsupported data type]	0
10	2960508149828	Pop	Alexandra	074515413	pop_alexandra96@yahoo.com	Covasna Str.Unirii nr.21	[unsupported data type]	0

Se creaza functia care afla clientul cel mai fidel:

```
CREATE OR REPLACE FUNCTION client_fidel RETURN
clienti_p.id_client%TYPE IS
v_id clienti_p.id_client%TYPE;
BEGIN
SELECT id_client INTO v_id FROM (SELECT
id_cet ,SUM(nr_persoane*pret*1.19) AS valoare_achizitii FROM pachete_p
JOIN vanzari_p ON id_pachet=id_pct GROUP BY id_cet ORDER BY
valoare_achizitii DESC) JOIN CLIENTI_P ON ID_CET=ID_CLIENT
WHERE ROWNUM=1;
RETURN v_id;
END client_fidel;
```

Se creaza procedura care aplica reducerea pt clientul aflat:

```
CREATE OR REPLACE PROCEDURE ofera_reducere IS
v_id clienti_p.id_client%TYPE;
BEGIN
v_id := client_fidel;
UPDATE clienti_p SET discount=0.1 WHERE id_client=v_id;
END;
```

Se verifica functionarea procedurii:

```
begin
ofera_reducere;
end;
```

ID_CLIENT	CNP	NUME	PRENUME	TELEFON	EMAIL	ADRESA	COPIE_BULETIN	DISCOUNT
11	1950428106476	Sima	Mihai	0761923074	sima_m95@yahoo.com	Buzau Str. Garoafelor nr. 33 bloc 42 ap.41	[unsupported data type]	.1
12	2970705426060	Anton	Maria	0747342153	anton.maria@yahoo.com	Bucuresti Str. Florarilor nr. 4	[unsupported data type]	0
13	1910115191256	Ionescu	Adrian	0762154320	ionescu_adr91@yahoo.com	Harghita Str. Ferdinand nr.15	[unsupported data type]	0
14	2861019302387	Anghel	Laura	0746342020	laura_angel123@yahoo.com	Satu Mare Str. Bosfor nr. 11	[unsupported data type]	0
15	1940906082791	Pancu	Ioan	0754432915	pancu.ioan4@yahoo.com	Brasov Str. Viitorului nr. 25	[unsupported data type]	0
16	1980207121475	Popescu	Bogdan	0752126982	bogdan_popescu1@yahoo.com	Cluj-Napoca Str. Bucuresti nr. 74 bloc 2 ap. 22	[unsupported data type]	0
10	2960508149828	Pop	Alexandra	074515413	pop_alexandra96@yahoo.com	Covasna Str.Unirii nr.21	[unsupported data type]	0

Se poate observa modificarea valorii coloanei discount pentru clientul cu id 11, Sima Mihai.



3. Odata la 6 luni firma ofera o marire de salariu de 10% angajatului care a realizat cele mai multe vanzari. Pentru a facilita acest process se va crea o procedura care utilizeaza o functie care afla cine este cel mai productiv angajat si il returneaza si care ii creste salariul acestui angajat.

Pentru a observa diferenta vom selecta inregistrarile din tabela angajati inainte de a rula procedura:  
 select \* from ANGAJATI\_p;

ID_ANGAJAT	NUME	PRENUME	TELEFON	ADRESA	DATA_ANGAJARII	SALARIU	ID_DPT	ID_MNR	POZA
118	Bostan	Emanuela	0723455555	Cluj- Napoca Str. Horea nr. 17 bl. 1 ap. 12	02-Feb-2009	6300	31	-	[unsupported data type]
112	Pancu	Alexandru	073433511	Cluj- Napoca Str. Oltului nr. 12 bl. 2 ap. 40	17-May-2009	6000	31	118	[unsupported data type]
113	Brandusa	Ionela	0722195400	Cluj-Napoca Str. Bucuresti nr. 14 bl. 2 ap. 11	19-Mar-2014	3500	31	112	[unsupported data type]
119	Bostan	Cosmin	0742333333	Cluj-Napoca Str. Lacul Rosu nr. 11 bl. 2 ap. 23	12-Mar-2009	5000	32	118	[unsupported data type]
120	Pop	Cristian	0723835460	Cluj-Napoca Str. Horea nr. 32 bl. 2 ap. 23	03-Jul-2009	4200	32	119	[unsupported data type]
121	Mare	David	0766554433	Cluj-Napoca Str. Bucuresti nr.2 bl. 2 ap. 2	06-Dec-2010	4000	34	117	[unsupported data type]
111	Sima	Sorin	0744103530	Cluj-Napoca Str. Muresului nr. 21	22-Oct-2010	4500	31	112	[unsupported data type]
115	Pop	Anamaria	0712345678	Cluj-Napoca Str. Scortariilor nr.4	04-Feb-2018	3200	31	112	[unsupported data type]
116	Ionescu	Vasile	0722233300	Cluj-Napoca Str. Horea nr. 7 bl. 3 ap. 11	05-Dec-2009	4100	33	118	[unsupported data type]
117	Popa	Andrada	0755522200	Cluj-Napoca Str. Sesului nr. 5	01-Jan-2010	4200	34	118	[unsupported data type]

Vom crea functia care afla cine este cel mai productiv angajat si ii returneaza id-ul:

```
CREATE OR REPLACE FUNCTION angajat_productiv RETURN
angajati_p.id_angajat%TYPE IS
v_id angajati_p.id_angajat%TYPE;
BEGIN
SELECT id_angajat INTO v_id FROM (select count(distinct nr_factura) as
nr_vanzari, id_agt from vanzari_p group by
id_agt order by nr_vanzari desc) JOIN angajati_p ON id_agt= id_angajat
WHERE ROWNUM=1;
RETURN v_id;
END angajat_productiv;
```

Vom crea procedura care mareste salariul acestui angajat:

```
CREATE OR REPLACE PROCEDURE mareste_salariul IS
v_id angajati_p.id_angajat%TYPE;
salariu_vechi angajati_p.salariu%TYPE;
salariu_nou angajati_p.salariu%TYPE;
BEGIN
v_id := angajat_productiv;
```

```

SELECT salariu INTO salariu_vechi FROM angajati_p WHERE
id_angajat=v_id;
salariu_nou:= salariu_vechi*(1+0.1);
UPDATE angajati_p SET salariu=salariu_nou WHERE id_angajat=v_id;
END;

```

Se verifica functionarea procedurii:

```

begin
mareste_salaiul;
end;

```

ID_ANGAJAT	NUME	PRENUME	TELEFON	ADRESA	DATA_ANGAJARII	SALARIU	ID_DPT	ID_MNR	POZA
118	Bostan	Emanuela	0723455555	Cluj- Napoca Str. Horea nr. 17 bl. 1 ap. 12	02-Feb-2009	6930	31	-	[unsupported data type]
112	Pancu	Alexandru	0734333511	Cluj- Napoca Str. Oltului nr. 12 bl. 2 ap. 40	17-May-2009	6000	31	118	[unsupported data type]
113	Brandusa	Ionela	0722195400	Cluj-Napoca Str. Bucuresti nr. 14 bl. 2 ap. 11	19-Mar-2014	3500	31	112	[unsupported data type]
119	Bostan	Cosmin	0742333333	Cluj-Napoca Str. Lacul Rosu nr. 11 bl. 2 ap. 23	12-Mar-2009	5000	32	118	[unsupported data type]
120	Pop	Cristian	0723835460	Cluj-Napoca Str. Horea nr. 32 bl. 2 ap. 23	03-Jul-2009	4200	32	119	[unsupported data type]
121	Mare	David	0766554433	Cluj-Napoca Str. Bucuresti nr.2 bl. 2 ap. 2	06-Dec-2010	4000	34	117	[unsupported data type]
111	Sima	Sorin	0744103530	Cluj-Napoca Str. Muresului nr. 21	22-Oct-2010	4500	31	112	[unsupported data type]
115	Don	Anamaria	0713245678	Cluj-Napoca Str. Scortolnicei nr. 4	04-Feb-2010	3300	31	113	[unsupported data type]

Se observa marirea salariului angajatului cu id-ul 118, Bostan Emanuela.

4. Sa se afle angajatii din departamentul x, transmis ca parametru, care au salariu mai mic decat salariul mediu din departamentul x si sa li se creasca salariul pana la salariul mediu, apoi sa se returneze valoarea salariilor lor dupa impozitare si numele lor. Pana la valoarea de 3500 de euro salariile vor fi impozitate 20%, valorile cuprinse intre 3500 si 5000 de euro vor fi impozitate 30% iar valorile mai mari de 6000 vor fi impozitate 40%.

Vom crea o procedura care transmitandu-l id-ul unui angajat si o valoare, va modifica salariul angajatului la valoarea transmisa:

```

CREATE OR REPLACE PROCEDURE modifica_salariu(valoare IN
angajati_p.salariu%TYPE,id_agt IN angajati_p.id_angajat%TYPE) IS
BEGIN
UPDATE angajati_p SET salariu=valoare WHERE id_angajat=id_agt;
END;

```

Vom crea o procedura care utilizeaza **un cursor care primeste 2 parametrii**, departamentul si salariul mediu din acel department, pentru a afla angajatii cu salariul sub medie. Apoi, trecand prin cursorul respective, se va modifica salariul fiecarui angajat aflat in cursor cu salariul mediu din department si se va afla salariul impozitat prin intermediul unei functii care primeste ca si parametru un salariu si returneaza salariul impozitat. In final se afiseaza numele angajatului si noul salariu a acestuia dupa impozitare

```
CREATE OR REPLACE PROCEDURE creste_salarii_sub_medie(dpt IN
angajati_p.id_dpt%TYPE) IS
CURSOR angajati_sub_medie(dpt angajati_p.id_dpt%TYPE, salariu_m
angajati_p.salariu%TYPE) IS
SELECT * FROM angajati_p WHERE id_dpt=dpt AND salariu<salariu_m;
salariu_mediu angajati_p.salariu%TYPE;
salariu_impozitat angajati_p.salariu%TYPE;
BEGIN
SELECT AVG(salariu) INTO salariu_mediu FROM angajati_p WHERE
id_dpt=dpt;
FOR angajati IN angajati_sub_medie(dpt,salariu_mediu) LOOP
modifica_salariu(salariu_mediu, angajati.id_angajat);
salariu_impozitat:=calcul_impozit(salariu_mediu);
DBMS_OUTPUT.PUT_LINE('Salariul angajatului '||angajati.numel||
' a fost actualizat.Salariul actual impozitat este:
'||salariu_impozitat);
END LOOP;
END;
```

Vom crea o functie care transmitandu-l un salariu va returna valoarea acestui salariu impozitat:

```
CREATE OR REPLACE FUNCTION calcul_impozit(sal IN
angajati_p.salariu%TYPE)
RETURN angajati_p.salariu%TYPE IS
v_impozit angajati_p.salariu%TYPE;
v_salariu angajati_p.salariu%TYPE;
BEGIN
IF sal<=3500 THEN
v_impozit:=sal*0.2;
ELSIF sal>3500 AND sal<=5000 THEN
v_impozit:=3500*0.2+(sal-3500)*0.3;
```

```

ELSIF sal>5000 THEN
v_impozit:=3500*0.2+1500*0.3+(sal-5000)*0.4;
END IF;
v_salariu:=sal-v_impozit;
RETURN v_salariu;
END calcul_impozit;

```

Vom testa functionalitatea functiei intr-un select:

```

SELECT nume, prenume, calcul_impozit(salariu) as salariu_impozitat from
angajati_p;

```

NUME	PRENUME	SALARIU_IMPOZITAT
Bostan	Emanuela	5008
Pancu	Alexandru	4450
Brandusa	Ionela	2800
Bostan	Cosmin	3850
Pop	Cristian	3290
Mare	David	3150
Sima	Sorin	3500
Pop	Anamaria	2560
Ionescu	Vasile	3220
Popa	Andrada	3290

Deoarece aflarea salariului impozitat poate fi necesara in multe actiuni realizate de firma, aceasta **functie va fi publica** in cadrul pachetului care il vom realiza pentru a putea fi apelat de catre angajatii firmei in orice moment prin intermediul pachetului.

La rularea procedurii, dandu-l departamentul 31 unde salariul mediu este 4826 obtinem rezultatul dorit:

```

Salariul angajatului Brandusa Ionela a fost actualizat.Salariul actual impozitat este: 3728
Salariul angajatului Sima Sorin a fost actualizat.Salariul actual impozitat este: 3728
Salariul angajatului Pop Anamaria a fost actualizat.Salariul actual impozitat este: 3728

Statement processed.

```

Se poate observa si la selectarea inregistrarilor din tabela angajati ca salariul unor angajati din departamentul 31 a fost modificat la valoarea 4826:

ID_ANGAJAT	NUME	PRENUME	TELEFON	ADRESA	DATA_ANGAJARII	SALARIU	ID_DPT	ID_MNR	POZA
118	Bostan	Emanuela	0723455555	Cluj- Napoca Str. Horea nr. 17 bl. 1 ap. 12	02-Feb-2009	6930	31	-	[unsupported data type]
112	Pancu	Alexandru	0734333511	Cluj- Napoca Str. Oltului nr. 12 bl. 2 ap. 40	17-May-2009	6000	31	118	[unsupported data type]
113	Brandusa	Ionela	0722195400	Cluj-Napoca Str. Bucuresti nr. 14 bl. 2 ap. 11	19-Mar-2014	4826	31	112	[unsupported data type]
119	Bostan	Cosmin	0742333333	Cluj-Napoca Str. Lacul Rosu nr. 11 bl. 2 ap. 23	12-Mar-2009	5000	32	118	[unsupported data type]
120	Pop	Cristian	0723835460	Cluj-Napoca Str. Horea nr. 32 bl. 2 ap. 23	03-Jul-2009	4200	32	119	[unsupported data type]
121	Mare	David	0766554433	Cluj-Napoca Str. Bucuresti nr.2 bl. 2 ap. 2	06-Dec-2010	4000	34	117	[unsupported data type]
111	Sima	Sorin	0744103530	Cluj-Napoca Str. Muresului nr. 21	22-Oct-2010	4826	31	112	[unsupported data type]
115	Pop	Anamaria	0712345678	Cluj-Napoca Str. Scortariilor nr. 4	04-Feb-2018	4826	31	112	[unsupported data type]
116	Ionescu	Vasile	0722233300	Cluj-Napoca Str. Horea nr. 7 bl. 3 ap. 11	05-Dec-2009	4100	33	118	[unsupported data type]
117	Popa	Andrada	0755522200	Cluj-Napoca Str. Sesului nr. 5	01-Jan-2010	4200	34	118	[unsupported data type]

More than 10 rows available. Increase rows selector to view more rows.

5. Se doreste sa se afle angajatii dintr-un department transmis care au salariul mai mare decat salariul mediu din acel department. Se doreste sa se afiseze numele si salariul impozitat a acestor angajati. Va trebui sa utilizam functia create la punctual anterior in cadrul unei procedure. Procedura va utiliza un **cursor implicit** care va trece prin angajatii cu salariul mai mare decat salariul mediu din departamentul transmis.

```

CREATE OR REPLACE PROCEDURE angajati_bine_platiti(dpt IN
angajati_p.id_dpt%TYPE) IS
salariu_mediu angajati_p.salariu%TYPE;
salariu_impozitat angajati_p.salariu%TYPE;
BEGIN
SELECT AVG(salariu) INTO salariu_mediu FROM angajati_p WHERE
id_dpt=dpt;

DBMS_OUTPUT.PUT_LINE('Angajatii cu salariul peste medie: ');

FOR angajati IN (SELECT * FROM angajati_p WHERE id_dpt=dpt AND
salariu>salariu_mediu) LOOP

salariu_impozitat:=calcul_impozit(angajati.salariu);
DBMS_OUTPUT.PUT_LINE('Nume: '||angajati.nume||' Prenume:
'||angajati.prenume||' salariu impozitat: '||salariu_impozitat);

END LOOP;
END;
```

Vom testa procedura:

```
BEGIN  
angajati_bine_platiti(31);  
END;
```

Angajatii cu salariul peste medie:

```
Nume: Bostan Prenume: Emanuela salariu impozitat: 5008  
Nume: Pancu Prenume: Alexandru salariu impozitat: 4450
```

Statement processed.

6. Vom utiliza secventa creata pentru tabela angajati si vom angaja prin intermediul unei proceduri o persoana noua in departamentul de vanzari.Vom folosi si **variabila globala cur\_date** care are valoarea sysdate.

```
CREATE or replace PROCEDURE angajez(ume IN  
angajati_p.ume%TYPE, prenume IN angajati_p.prenume%TYPE, telefon  
IN angajati_p.telefon%TYPE, adresa IN angajati_p.adresa%TYPE , salariu  
IN angajati_p.salariu%TYPE, id_dpt IN angajati_p.id_dpt%TYPE, id_mnr  
IN angajati_p.id_mnr%TYPE) IS  
BEGIN  
INSERT INTO  
angajati_p(id_angajat,ume,prenume,telefon,adresa,data_angajarii,salariu,  
id_dpt,id_mnr)  
VALUES(angajati_angid_seq.NEXTVAL,ume,prenume,telefon,adresa,cur  
_date,salariu,id_dpt,id_mnr);  
END;
```

La rularea procedurii obtinem rezultatul dorit:

```
BEGIN  
angajez('Kotai','Melinda','0746342020','STRADA  
VIITORULUI',3000,31,118);  
END;
```

ID_ANGAJAT	NUME	PRENUME	TELEFON	ADRESA	DATA_ANGAJARII	SALARIU	ID_DPT	ID_MNR
118	Bostan	Emanuela	0723455555	Cluj- Napoca Str. Horea nr. 17 bl. 1 ap. 12	02-Feb-2009	6930	31	-
112	Pancu	Alexandru	0734333511	Cluj- Napoca Str. Oltului nr. 12 bl. 2 ap. 40	17-May-2009	6000	31	118
113	Brandusa	Ionela	0722195400	Cluj- Napoca Str. Bucuresti nr. 14 bl. 2 ap. 11	19-Mar-2014	4826	31	112
119	Bostan	Cosmin	0742333333	Cluj- Napoca Str. Lacul Rosu nr. 11 bl. 2 ap. 23	12-Mar-2009	5000	32	118
120	Pop	Cristian	0723835460	Cluj- Napoca Str. Horea nr. 32 bl. 2 ap. 23	03-Jul-2009	4200	32	119
121	Mare	David	0766554433	Cluj- Napoca Str. Bucuresti nr.2 bl. 2 ap. 2	06-Dec-2010	4000	34	117
125	Kotai	Melinda	0746342020	STRADA VIITORULUI	23-Dec-2020	3000	31	118
111	Sima	Sorin	0744103530	Cluj- Napoca Str. Muresului nr. 21	22-Oct-2010	4826	31	112
115	Pop	Anamaria	0712345678	Cluj- Napoca Str. Scortariilor nr. 4	04-Feb-2018	4826	31	112
116	Ionescu	Vasile	0722233300	Cluj- Napoca Str. Horea nr. 7 bl. 3 ap. 11	05-Dec-2009	4100	33	118

7. Vom utiliza secventa creata pentru tabelul clienti pentru a crea o pocedura care introduce un client nou in baza de date.

```
CREATE OR REPLACE PROCEDURE adaug_client(cnp IN
clienti_p.cnp%TYPE, nume IN clienti_p.numa%TYPE, prenume IN
clienti_p.prenume%TYPE, telefon IN clienti_p.telefon%TYPE,email IN
clienti_p.email%TYPE, adresa IN clienti_p.adresa%TYPE) IS
BEGIN
INSERT INTO clienti_p(id_client,cnp,nume,prenume,telefon,email,adresa)
VALUES(clienti_cliid_seq.NEXTVAL,cnp,nume,prenume,telefon,email,adre
sa);
END;
```

La rularea procedurii obtinem rezultatul dorit:

```
BEGIN
adaug_client('1234567899999','Kotai','Melinda','0746342120','kotai.melinda
@yahoo.com','STRADA VIITORULUI');
END;
```

ID_CLIENT	CNP	NUME	PRENUME	TELEFON	EMAIL	ADRESA	COPIE_BULETIN	DISCOUNT
11	1950428106476	Sima	Mihai	0761923074	sima_m95@yahoo.com	Buzau Str. Garoafelor nr. 33 bloc 42 ap.41	[unsupported data type]	.1
12	2970705426060	Anton	Maria	0747342153	anton.maria@yahoo.com	Bucuresti Str. Florarilor nr. 4	[unsupported data type]	0
13	1910115191256	Ionescu	Adrian	0762154320	ionescu_adr91@yahoo.com	Harghita Str. Ferdinand nr.15	[unsupported data type]	0
14	2861019302387	Anghel	Laura	0746342020	laura_anghel123@yahoo.com	Satu Mare Str. Bosfor nr. 11	[unsupported data type]	0
15	1940906082791	Pancu	Ioan	0754432915	pancu.ioan4@yahoo.com	Brasov Str. Viitorului nr. 25	[unsupported data type]	0
16	1980207121475	Popescu	Bogdan	0752126982	bogdan_popescu1@yahoo.com	Cluj-Napoca Str. Bucuresti nr. 74 bloc 2 ap. 22	[unsupported data type]	0
10	2960508149828	Pop	Alexandra	074515413	pop_alexandra96@yahoo.com	Covasna Str.Unirii nr.21	[unsupported data type]	0
18	1234567899999	Kotai	Melinda	0746342120	kotai.melinda@yahoo.com	STRADA VIITORULUI	[unsupported data type]	0

8. Vom utiliza secventa creata pentru tabelul departamente pentru a crea o procedura care introduce un departament nou in baza de date.

```
CREATE OR REPLACE PROCEDURE adaug_departament(denumire IN
departamente_p.denumire%TYPE) IS
BEGIN
INSERT INTO departamente_p(id_departament,denumire)
VALUES(departamente_deptid_seq.NEXTVAL,denumire);
END;
```

La rularea procedurii obtinem rezultatul dorit:

```
BEGIN
adaug_departament('Management');
END;
```

ID_DEPARTAMENT	DENUMIRE
31	Vanzari
32	Marketing
33	Financiar-Contabil
34	Resurse Umane
35	Management

9. Sa se returneze pretul cu tva a pachetelor cu destinatia in tara x transmisa ca parametru. Se va crea o procedura care utilizand un cursor care prin intermediul parametrului transmis "tara" va afla toate pachetele oferite de firma din tara respective. Prin intermediul unei bucle for se va trece prin toate pachetele aflate si se va afisa id-u lor si pretul lor cu tva aflat prin inmultirea pretului din baza de date cu **variabila globala din pachet tva** .

```
CREATE OR REPLACE PROCEDURE pret_pachete_tara(tara IN
tari_p.denumire%TYPE) IS
CURSOR pachete_tara(tara tari_p.denumire%TYPE) IS
SELECT pachete_p.id_pachet , pachete_p.pret FROM pachete_p,
destinatii_p, tari_p WHERE id_dte=id_destinatie AND id_tra=id_tara AND
tari_p.denumire=tara ;
BEGIN
FOR pachet IN pachete_tara(tara) LOOP
```



```

DBMS_OUTPUT.PUT_LINE('Id pachet: '||pachet.id_pachet||' pret cu tva:
'||pachet.pret*tva);
END LOOP;
EXCEPTION
    WHEN NO_DATA_FOUND THEN
        DBMS_OUTPUT.PUT_LINE('NU EXISTA PACHETE IN TARA
INTRODUSA!');
END;

BEGIN
    pret_pachete_tara('Italia');
END;

```

```

Id pachet: 1123 pret cu tva: 1290
Id pachet: 1128 pret cu tva: 1090
Id pachet: 1131 pret cu tva: 1000

```

10. **Cursor multiplu:** Se doreste sa se afiseze destinatiile si pachetele valabile la acele destinatii.

```

CREATE OR REPLACE PROCEDURE destinatii_pachete IS
    CURSOR destinatii IS
        SELECT * FROM destinatii_p;
    CURSOR pachete(destinatie destinatii_p.id_destinatie%TYPE) IS
        SELECT * FROM pachete_p WHERE id_dte=destinatie;
    BEGIN
        FOR destinatie IN destinatii LOOP
            DBMS_OUTPUT.PUT_LINE('La destinatia '||destinatie.denumire||'
            exista pachetele cu id-urile: ');
            FOR pachet IN pachete(destinatie.id_destinatie) LOOP
                DBMS_OUTPUT.PUT_LINE(pachet.id_pachet);
            END LOOP;
        END LOOP;
    END;

begin
    destinatii_pachete;
end;

```

```

La destinatia Nisa exista pachetele cu id-urile:
1130
La destinatia Constanta exista pachetele cu id-urile:
1132
La destinatia Tokyo exista pachetele cu id-urile:
La destinatia Honolulu exista pachetele cu id-urile:
1121
La destinatia Maui exista pachetele cu id-urile:
1122
La destinatia Roma exista pachetele cu id-urile:
1123
La destinatia Atena exista pachetele cu id-urile:
1124
La destinatia Creta exista pachetele cu id-urile:
1125
La destinatia Giza exista pachetele cu id-urile:
1126
La destinatia Cair exista pachetele cu id-urile:
1127
La destinatia Florenta exista pachetele cu id-urile:
1128
1131
La destinatia Paris exista pachetele cu id-urile:
1129
La destinatia Mamaia exista pachetele cu id-urile:
La destinatia Osaka exista pachetele cu id-urile:

Statement processed.

```

11. Cursor multiplu: Firma se decide sa acorde o marire de salariu de 10% tuturor angajatilor care au salariul mai mic decat salariul mediu din departamentul in care se afla.

Pentru aceasta se creaza o procedura care va utiliza un cursor multiplu. Se va trece prin toate departamentele, pentru fiecare departament se va afla salariul mediu dupa care se va trece prin toti angajatii din departament si se va verifica daca acestia au salariul mai mic decat salariul mediu. Daca au, atunci se va apela procedura marire salariu.

```

CREATE OR REPLACE PROCEDURE mareste_salarii_sub_medie IS
CURSOR departamente IS
SELECT * FROM departamente_p;

```

```

CURSOR angajati(dpt angajati_p.id_dpt%TYPE) IS
SELECT * FROM angajati_p WHERE id_dpt=dpt;

```

```

salariu_mediu angajati_p.salariu%TYPE;

```

```

BEGIN
FOR departament IN departamente LOOP
    SELECT AVG(salariu) INTO salariu_mediu FROM angajati_p
WHERE id_dpt=departament.id_departament;

```

```

FOR angajat IN angajati(departament.id_departament) LOOP
    IF angajat.salariu<salariu_mediu THEN
        marire_salariu(0.1, angajat.id_angajat);
    END IF;
END LOOP;
END LOOP;
END;

```

```

CREATE OR REPLACE PROCEDURE marire_salariu(valoare IN
NUMBER, id IN angajati_p.id_angajat%TYPE) IS
    salariu_vechi angajati_p.salariu%TYPE;
    salariu_nou angajati_p.salariu%TYPE;
BEGIN
    SELECT salariu INTO salariu_vechi FROM angajati_p WHERE
    id_angajat=id;
    salariu_nou:=salariu_vechi*(1+valoare);
    UPDATE angajati_p SET salariu=salariu_nou WHERE id_angajat=id;
END;

```

```

BEGIN
    Mareste_salarii_sub_medie;
END;

```

Inregistrările din tabela angajati inainte si dupa rularea procedurii:

ID_ANGAJAT	NUME	PRENUME	TELEFON	ADRESA	DATA_ANGAJARII	SALARIU	ID_DPT	ID_MNR	POZA
118	Bostan	Emanuela	0723455555	Cluj- Napoca Str. Horea nr. 17 bl. 1 ap. 12	02-Feb-2009	6930	31	-	[unsupported data type]
112	Pancu	Alexandru	0734333511	Cluj- Napoca Str. Oltului nr. 12 bl. 2 ap. 40	17-May-2009	6000	31	118	[unsupported data type]
113	Brandusa	Ionela	0722195400	Cluj-Napoca Str. Bucuresti nr. 14 bl. 2 ap. 11	19-Mar-2014	4826	31	112	[unsupported data type]
119	Bostan	Cosmin	0742333333	Cluj-Napoca Str. Lacul Rosu nr. 11 bl. 2 ap. 23	12-Mar-2009	5000	32	118	[unsupported data type]
120	Pop	Cristian	0723835460	Cluj-Napoca Str. Horea nr. 32 bl. 2 ap. 23	03-Jul-2009	4200	32	119	[unsupported data type]
121	Mare	David	0766554433	Cluj-Napoca Str. Bucuresti nr.2 bl. 2 ap. 2	06-Dec-2010	4000	34	117	[unsupported data type]
125	Kotai	Melinda	0746342020	STRADA VIITORULUI	23-Dec-2020	3000	31	118	[unsupported data type]
111	Sima	Sorin	0744103530	Cluj-Napoca Str. Muresului nr. 21	22-Oct-2010	4826	31	112	[unsupported data type]
115	Pop	Anamaria	0712345678	Cluj-Napoca Str. Scortanilor nr. 4	04-Feb-2018	4826	31	112	[unsupported data type]
116	Ionescu	Vasile	0722233300	Cluj-Napoca Str. Horea nr. 7 bl. 3 ap. 11	05-Dec-2009	4100	33	118	[unsupported data type]
117	Popa	Andrada	0755522200	Cluj-Napoca Str. Sesului nr. 5	01-Jan-2010	4200	34	118	[unsupported data type]
114	Constantin	Marian	0761351102	Cluj- Napoca Str. Muresului nr.12	13-Sep-2013	3000	32	119	[unsupported data type]
122	Paul	Maria	0766557788	Cluj-Napoca Str. Fabricii nr. 50	19-Nov-2010	4000	33	116	[unsupported data type]

ID_ANGAJAT	NUME	PRENUME	TELEFON	ADRESA	DATA_ANGAJARII	SALARIU	ID_DPT	ID_MNR	POZA
118	Bostan	Emanuela	0723455555	Cluj- Napoca Str. Horea nr. 17 bl. 1 ap. 12	02-Feb-2009	6930	31	-	[unsupported data type]
112	Pancu	Alexandru	0734333511	Cluj- Napoca Str. Oltului nr. 12 bl. 2 ap. 40	17-May-2009	6000	31	118	[unsupported data type]
113	Brandusa	Ionela	0722195400	Cluj-Napoca Str. Bucuresti nr. 14 bl. 2 ap. 11	19-Mar-2014	5309	31	112	[unsupported data type]
119	Bostan	Cosmin	0742333333	Cluj-Napoca Str. Lacul Rosu nr. 11 bl. 2 ap. 23	12-Mar-2009	5000	32	118	[unsupported data type]
120	Pop	Cristian	0723835460	Cluj-Napoca Str. Horea nr. 32 bl. 2 ap. 23	03-Jul-2009	4200	32	119	[unsupported data type]
121	Mare	David	0766554433	Cluj-Napoca Str. Bucuresti nr.2 bl. 2 ap. 2	06-Dec-2010	4400	34	117	[unsupported data type]
125	Kotai	Melinda	0746342020	STRADA VIITORULUI	23-Dec-2020	3300	31	118	[unsupported data type]
111	Sima	Sorin	0744103530	Cluj-Napoca Str. Muresului nr. 21	22-Oct-2010	5309	31	112	[unsupported data type]
115	Pop	Anamaria	0712345678	Cluj-Napoca Str. Scortanilor nr. 4	04-Feb-2018	5309	31	112	[unsupported data type]
116	Ionescu	Vasile	0722233300	Cluj-Napoca Str. Horea nr. 7 bl. 3 ap. 11	05-Dec-2009	4100	33	118	[unsupported data type]
117	Popa	Andrada	0755522200	Cluj-Napoca Str. Sesului nr. 5	01-Jan-2010	4200	34	118	[unsupported data type]
114	Constantin	Marian	0761351102	Cluj- Napoca Str. Muresului nr.12	13-Sep-2013	3300	32	119	[unsupported data type]
122	Paul	Maria	0766557788	Cluj-Napoca Str. Fabricii nr. 50	19-Nov-2010	4400	33	116	[unsupported data type]

12. Firma se decide sa concedieze angajatul cel mai putin productiv. Vom crea o functie care returneaza id-ul angajatului cu cele mai putine vanzari care va fi folosita intr-o procedura care il sterge pe acesta din baza de date.

```

Functia care returneaza id-ul angajatului neproductiv din baza de date.
CREATE OR REPLACE FUNCTION angajat_neproductiv RETURN
angajati_p.id_angajat%TYPE IS
v_id angajati_p.id_angajat%TYPE;
BEGIN
SELECT id_angajat INTO v_id FROM (select count(distinct nr_factura) as
nr_vanzari, id_agt from vanzari_p group by
id_agt order by nr_vanzari) JOIN angajati_p ON id_agt= id_angajat
WHERE ROWNUM=1;
RETURN v_id;
END;
```

Procedura care sterge acest angjat din baza de date:

```

CREATE OR REPLACE PROCEDURE concediez IS
v_id angajati.employee_id%TYPE;
PROCEDURE AFISEZ_CONFIRMARE IS
BEGIN
DBMS_OUTPUT.PUT_LINE('Angajatul a fost sters');
```

```
END AFISEZ_CONFIRMARE;  
BEGIN  
v_id:=angajat_neproductiv;  
DELETE FROM angajati_p WHERE id_angajat=v_id;  
AFISEZ_CONFIRMARE;  
END;
```

```
BEGIN  
concediez;  
END;
```

---

Angajatul a fost sters

Statement processed.

0.01 seconds

Angajatul cu id-ul 115 a fost sters.

Vom grupa procedurile si functiile create intr-un pachet astfel:

**Procedurile** :scad\_pret\_pachet\_nepopular, ofera\_reducere, mareste\_salariul, creste\_salarii\_sub\_medie, angajati\_bine\_platiti, angajezi, adaug\_client, adaug\_departament, pret\_pachete\_tara, destinatii\_pachete, mareste\_salarii\_sub\_medie, concediez vor fi **publice**.

**Procedurile**: scade\_pret\_pachet, modifica\_salariul, marire\_salariu vor fi **private**.

**Functia**: calcul\_impozit vs fi **publica**.

**Functiile**: client\_fidel, angajat\_productiv, angajat\_neproductiv vor fi **private**

**Si vom avea 2 variabile globale tva=1.19 si data\_cur=SYSDATE.**

Pachetul se va numi PAT (pachet agentie turism).

```

CREATE OR REPLACE PACKAGE pat IS
tva NUMBER:=1.19;
cur_date DATE:=SYSDATE;
FUNCTION calcul_impozit(sal IN angajati_p.salariu%TYPE)
RETURN angajati_p.salariu%TYPE;
PROCEDURE scad_pret_pachet_nepopular(nr_vanzari IN NUMBER,
valoare IN pachete_p.pret%TYPE);
PROCEDURE ofera_reducere;
PROCEDURE mareste_salariul;
PROCEDURE creste_salarii_sub_medie(dpt IN angajati_p.id_dpt%TYPE);
PROCEDURE angajati_bine_platiti(dpt IN angajati_p.id_dpt%TYPE);
PROCEDURE angajezi(ume IN angajati_p.ume%TYPE, prenume IN
angajati_p.prenume%TYPE, telefon IN angajati_p.telefon%TYPE, adresa
IN angajati_p.adresa%TYPE , salariu IN angajati_p.salariu%TYPE, id_dpt
IN angajati_p.id_dpt%TYPE, id_mnr IN angajati_p.id_mnr%TYPE);
PROCEDURE adaug_client(cnp IN clienti_p.cnp%TYPE, nume IN
clienti_p.nume%TYPE, prenume IN clienti_p.prenume%TYPE, telefon IN
clienti_p.telefon%TYPE,email IN clienti_p.email%TYPE, adresa IN
clienti_p.adresa%TYPE);
PROCEDURE pret_pachete_tara(tara IN tari_p.denumire%TYPE);
PROCEDURE destinatii_pachete;
PROCEDURE mareste_salarii_sub_medie;
PROCEDURE concediez;
END pat;

```

CREATE OR REPLACE PACKAGE BODY pat IS

```

PROCEDURE scade_pret_pachet(id IN pachete_p.id_pachet%TYPE,
valoare IN pachete_p.pret%TYPE) IS
pret_nou pachete_p.pret%TYPE;
pret_vechi pachete_p.pret%TYPE;
BEGIN
SELECT pret INTO pret_vechi FROM pachete_p WHERE id_pachet=id;
pret_nou:=pret_vechi-valoare;
UPDATE pachete_p SET pret=pret_nou WHERE id_pachet=id;

```

END;

```
PROCEDURE modifica_salariu(valoare IN angajati_p.salariu%TYPE,id_agt
IN angajati_p.id_angajat%TYPE) IS
BEGIN
UPDATE angajati_p SET salariu=valoare WHERE id_angajat=id_agt;
END;
```

```
PROCEDURE marire_salariu(valoare IN NUMBER, id IN
angajati_p.id_angajat%TYPE) IS
salariu_vechi angajati_p.salariu%TYPE;
salariu_nou angajati_p.salariu%TYPE;
BEGIN
SELECT salariu INTO salariu_vechi FROM angajati_p WHERE
id_angajat=id;
salariu_nou:=salariu_vechi*(1+valoare);
UPDATE angajati_p SET salariu=salariu_nou WHERE id_angajat=id;
END;
```

```
FUNCTION client_fidel RETURN clienti_p.id_client%TYPE IS
v_id clienti_p.id_client%TYPE;
BEGIN
SELECT id_client INTO v_id FROM (SELECT
id_cet ,SUM(nr_persoane*pret*tva) AS valoare_achizitii FROM pachete_p
JOIN vanzari_p ON id_pachet=id_pct GROUP BY id_cet ORDER BY
valoare_achizitii DESC) JOIN CLIENTI_P ON ID_CET=ID_CLIENT
WHERE ROWNUM=1;
RETURN v_id;
END client_fidel;
```

```
FUNCTION angajat_productiv RETURN angajati_p.id_angajat%TYPE IS
v_id angajati_p.id_angajat%TYPE;
BEGIN
SELECT id_angajat INTO v_id FROM (select count(distinct nr_factura) as
nr_vanzari, id_agt from vanzari_p group by
id_agt order by nr_vanzari desc) JOIN angajati_p ON id_agt= id_angajat
WHERE ROWNUM=1;
RETURN v_id;
```

```
END angajat_productiv;
```

```
FUNCTION angajat_neproductiv RETURN angajati_p.id_angajat%TYPE IS  
v_id angajati_p.id_angajat%TYPE;  
BEGIN  
SELECT id_angajat INTO v_id FROM (select count(distinct nr_factura) as  
nr_vanzari, id_agt from vanzari_p group by  
id_agt order by nr_vanzari) JOIN angajati_p ON id_agt= id_angajat  
WHERE ROWNUM=1;  
RETURN v_id;  
END;
```

```
FUNCTION calcul_impozit(sal IN angajati_p.salariu%TYPE)  
RETURN angajati_p.salariu%TYPE IS  
v_impozit angajati_p.salariu%TYPE;  
v_salariu angajati_p.salariu%TYPE;  
BEGIN  
IF sal<=3500 THEN  
v_impozit:=sal*0.2;  
ELSIF sal>3500 AND sal<=5000 THEN  
v_impozit:=3500*0.2+(sal-3500)*0.3;  
ELSIF sal>5000 THEN  
v_impozit:=3500*0.2+1500*0.3+(sal-5000)*0.4;  
END IF;  
v_salariu:=sal-v_impozit;  
RETURN v_salariu;  
END calcul_impozit;
```

```
PROCEDURE scad_pret_pachet_nepopular(nr_vanzari IN NUMBER,  
valoare IN pachete_p.pret%TYPE) IS  
CURSOR pachete_nepopulare(nr_v NUMBER) IS  
SELECT id_pct, nr_vanzari from (SELECT id_pct, count(id_pct) as  
nr_vanzari from vanzari_p group by id_pct) WHERE nr_vanzari<=nr_v;  
BEGIN  
FOR pachet IN pachete_nepopulare(nr_vanzari) LOOP  
scade_pret_pachet(pachet.id_pct,valoare);
```



```
END LOOP;  
END;
```

```
PROCEDURE ofera_reducere IS  
v_id clienti_p.id_client%TYPE;  
BEGIN  
v_id := client_fidel;  
UPDATE clienti_p SET discount=0.1 WHERE id_client=v_id;  
END;
```

```
PROCEDURE mareste_salariul IS  
v_id angajati_p.id_angajat%TYPE;  
salariu_vechi angajati_p.salariu%TYPE;  
salariu_nou angajati_p.salariu%TYPE;  
BEGIN  
v_id := angajat_productiv;  
SELECT salariu INTO salariu_vechi FROM angajati_p WHERE  
id_angajat=v_id;  
salariu_nou:= salariu_vechi*(1+0.1);  
UPDATE angajati_p SET salariu=salariu_nou WHERE id_angajat=v_id;  
END;
```

```
PROCEDURE creste_salarii_sub_medie(dpt IN angajati_p.id_dpt%TYPE)  
IS  
CURSOR angajati_sub_medie(dpt angajati_p.id_dpt%TYPE, salariu_m  
angajati_p.salariu%TYPE) IS  
SELECT * FROM angajati_p WHERE id_dpt=dpt AND salariu<salariu_m;  
salariu_mediu angajati_p.salariu%TYPE;  
salariu_impozitat angajati_p.salariu%TYPE;  
BEGIN  
SELECT AVG(salariu) INTO salariu_mediu FROM angajati_p WHERE  
id_dpt=dpt;  
FOR angajati IN angajati_sub_medie(dpt,salariu_mediu) LOOP  
modifica_salariu(salariu_mediu, angajati.id_angajat);  
salariu_impozitat:=calcul_impozit(salariu_mediu);  
DBMS_OUTPUT.PUT_LINE('Salariul angajatului '||angajati.numel||'  
'||angajati.prenume||' a fost actualizat.Salariul actual impozitat este:  
'||salariu_impozitat);
```

```
END LOOP;  
END;
```

```
PROCEDURE angajati_bine_platiti(dpt IN angajati_p.id_dpt%TYPE) IS  
salariu_mediu angajati_p.salariu%TYPE;  
salariu_impozitat angajati_p.salariu%TYPE;  
BEGIN  
SELECT AVG(salariu) INTO salariu_mediu FROM angajati_p WHERE  
id_dpt=dpt;
```

```
DBMS_OUTPUT.PUT_LINE('Angajatii cu salariul peste medie: ');
```

```
FOR angajati IN (SELECT * FROM angajati_p WHERE id_dpt=dpt AND  
salariu>salariu_mediu) LOOP
```

```
salariu_impozitat:=calcul_impozit(angajati.salariu);  
DBMS_OUTPUT.PUT_LINE('Nume: '||angajati.numel||' Prenume:  
'||angajati.prenume||' salariu impozitat: '||salariu_impozitat);
```

```
END LOOP;  
END;
```

```
PROCEDURE angajez(numel IN angajati_p.numel%TYPE, prenumel IN  
angajati_p.prenume%TYPE, telefon IN angajati_p.telefon%TYPE, adresa  
IN angajati_p.adresa%TYPE , salariu IN angajati_p.salariu%TYPE, id_dpt  
IN angajati_p.id_dpt%TYPE, id_mnr IN angajati_p.id_mnr%TYPE) IS  
BEGIN  
INSERT INTO  
angajati_p(id_angajat,numel,prenume,telefon,adresa,data_angajarii,salariu,  
id_dpt,id_mnr)  
VALUES(angajati_angid_seq.NEXTVAL,numel,prenume,telefon,adresa,cur  
_date,salariu,id_dpt,id_mnr);  
END;
```

```
PROCEDURE adaug_client(cnp IN clienti_p.cnp%TYPE, numel IN  
clienti_p.numel%TYPE, prenumel IN clienti_p.prenume%TYPE, telefon IN  
clienti_p.telefon%TYPE,email IN clienti_p.email%TYPE, adresa IN  
clienti_p.adresa%TYPE) IS  
BEGIN
```

```
INSERT INTO clienti_p(id_client,cnp,nume,prenume,telefon,email,adresa)
VALUES(clienti_cliid_seq.NEXTVAL,cnp,nume,prenume,telefon,email,adresa);
END;
```

```
PROCEDURE adaug_departament(denumire IN
departamente_p.denumire%TYPE) IS
BEGIN
INSERT INTO departamente_p(id_departament,denumire)
VALUES(departamente_deptid_seq.NEXTVAL,denumire);
END;
```

```
PROCEDURE pret_pachete_tara(tara IN tari_p.denumire%TYPE) IS
CURSOR pachete_tara(tara tari_p.denumire%TYPE) IS
SELECT pachete_p.id_pachet , pachete_p.pret FROM pachete_p,
destinatii_p, tari_p WHERE id_dte=id_destinatie AND id_tra=id_tara AND
tari_p.denumire=tara ;
BEGIN
FOR pachet IN pachete_tara(tara) LOOP
DBMS_OUTPUT.PUT_LINE('Id pachet: '||pachet.id_pachet||' pret cu tva:
' || pachet.pret*tva);
END LOOP;
EXCEPTION
    WHEN NO_DATA_FOUND THEN
        DBMS_OUTPUT.PUT_LINE('NU EXISTA PACHETE IN TARA
INTRODUSA!');
END;
```

```
PROCEDURE destinatii_pachete IS
CURSOR destinatii IS
SELECT * FROM destinatii_p;
CURSOR pachete(destinatie destinatii_p.id_destinatie%TYPE) IS
SELECT * FROM pachete_p WHERE id_dte=destinatie;
BEGIN
FOR destinatie IN destinatii LOOP
    DBMS_OUTPUT.PUT_LINE('La destinatia '||destinatie.denumire||'
exista pachetele cu id-urile: ');
    FOR pachet IN pachete(destinatie.id_destinatie) LOOP
        DBMS_OUTPUT.PUT_LINE(pachet.id_pachet);
    END LOOP;
```

```
END LOOP;  
END;
```

```
PROCEDURE maresta_salarii_sub_medie IS  
CURSOR departamente IS  
SELECT * FROM departamente_p;
```

```
CURSOR angajati(dpt angajati_p.id_dpt%TYPE) IS  
SELECT * FROM angajati_p WHERE id_dpt=dpt;
```

```
salariu_mediu angajati_p.salariu%TYPE;
```

```
BEGIN  
FOR departament IN departamente LOOP  
    SELECT AVG(salariu) INTO salariu_mediu FROM angajati_p  
WHERE id_dpt=departament.id_departament;  
    FOR angajat IN angajati(departament.id_departament) LOOP  
        IF angajat.salariu<salariu_mediu THEN  
            marire_salariu(0.1, angajat.id_angajat);  
        END IF;  
    END LOOP;  
END LOOP;  
END;
```

```
PROCEDURE concediez IS  
v_id angajati.employee_id%TYPE;  
PROCEDURE AFISEZ_CONFIRMARE IS  
BEGIN  
    DBMS_OUTPUT.PUT_LINE('Angajatul a fost sters');  
END AFISEZ_CONFIRMARE;  
BEGIN  
v_id:=angajat_neproductiv;  
DELETE FROM angajati_p WHERE id_angajat=v_id;  
AFISEZ_CONFIRMARE;  
END;
```

```
END pat;
```

La rularea codului de mai sus obținem mesaje de confirmare ca pachetul a fost creat cu succes:

Results	Explain	Describe	Saved SQL	History
Package created.				
0.02 seconds				
Package Body created.				
0.07 seconds				

Vom rula codul urmator pentru a testa functionarea pachetului:

```
DECLARE
nr NUMBER;
BEGIN
nr:=pat.calcul_impozit(4000);
DBMS_OUTPUT.PUT_LINE(nr);
END;
```

3150  
Statement processed.

În acest fel pot fi apelate toate procedurile/functiile publice declarate în specificațiile pachetului, funcțiile/procedurile private (care nu sunt declarate în specificații și apar doar în body) vor fi vizibile doar în corpul pachetului, nu vor putea fi apelate ca și funcția apelată mai sus.

Vom crea 5 declansatori pentru a jurnaliza unele operatii din paza de date:

1. Vom crea un declansator care sa inregistreze userul care a efectuat update-ul la salariul unui angajat, data la care s-a realizat update-ul, salariu vechi, salariul nou intr-un table jurnalizare\_salarii si care sa nu permita schimbarea salariului intr-unul mai mic decat 90% din salariul vechi.

```
CREATE TABLE jurnalizare_salarii(  
    utilizator VARCHAR2(30),  
    data DATE,  
    id_angajat NUMBER(3),  
    salariu_vechi NUMBER(4),  
    salariu_nou NUMBER(4)  
);
```

```
CREATE OR REPLACE TRIGGER jurnalizare_salarii_trigg BEFORE  
UPDATE OF salariu ON angajati_p FOR EACH ROW  
BEGIN
```

```
    IF :NEW.salariu < (:OLD.salariu * 0.9) THEN  
        RAISE_APPLICATION_ERROR(-20500, 'Nu este permisa  
scaderea salariului mai mult de 10%! ');  
    END IF;  
    INSERT INTO jurnalizare_salarii VALUES (USER, SYSDATE,  
:OLD.id_angajat, :OLD.salariu, :NEW.salariu);  
  
END;
```

Vom testa declansatorul:

```
UPDATE angajati_p SET salariu=0 WHERE id_angajat=118;
```

```
ORA-20500: Nu este permisa scaderea salariului mai mult de 10%!  
ORA-06512: at "RO_A178_PLSQL_S17.JURNALIZARE_SALARII_TRIGG", line 4  
ORA-04088: error during execution of trigger  
"RO_A178_PLSQL_S17.JURNALIZARE_SALARII_TRIGG"
```

```
update angajati_p set salariu=7000 where id_angajat=118;
```

```
SELECT * FROM jurnalizare_salarii ;
```

UTILIZATOR	DATA	ID_ANGAJAT	SALARIU_VECHI	SALARIU_NOU
APEX_PUBLIC_USER	24-Dec-2020	118	6930	7000

1 rows returned in 0.00 seconds [Download](#)

2. Vom crea un declansator care va copia intr-un table de rezerva pachete\_sterse pachetele sterse.

```
CREATE TABLE pachete_sterse LIKE pachete_p;
```

```
CREATE OR REPLACE TRIGGER jurnalizare_pachete_sterse
AFTER DELETE ON pachete_p FOR EACH ROW BEGIN
    INSERT INTO pachete_sterse VALUES(:OLD.id_pachet,
:OLD.nr_zile, :OLD.servicii, :OLD.hotel, :OLD.pret,
:OLD.tip_transport, :OLD.aeroport_plecare,
:OLD.aeroport_destinatie, :OLD.clasa_a, :OLD.bagaj_cala,
:OLD.nr_escale, :OLD.gara_plecare, :OLD.gara_destinatie,
:OLD.clasa_t,:OLD.autogara_plecare, :OLD.autogara_destinatie,
:OLD.id_dte);
END;
```

Vom testa declansatorul:

```
DELETE FROM pachete_p WHERE id_pachet=1130;
```

```
SELECT * FROM pachete_sterse;
```

ID_PACHET	NR_ZILE	SERVICII	HOTEL	PRET	TIP_TRANSPORT	AEROPORT_PLECARE	AEROPORT_DESTINATIE	CLASA_A	BAGAJ_CALA	NR_ESCALE	GARA_PLECARE	GARA_DESTINATIE	CLASA_T	AUTOGARA_PLECARE	AUTOGARA_DESTINATIE	ID_DTE
1130	7	demipensiune	Charmant	1990	aerian	Aeroportul International Avram Iancu Cluj	Aeroportul Coasta de Azur Nisa	2	30	-	-	-	-	-	-	30

3. Se doreste jurnalizarea schimparii preturilor la pachete.  
Atunci cand se schimba coloana pret din tabela pachete sa se inregistreze cine a schimbat + valoare veche + valoare noua + data modificarii pretului.

```
CREATE TABLE jurnalizare_pret_pachete(
Utilizator VARCHAR2(30),
Data DATE,
Pret_vechi NUMBER(4),
Pret_nou NUMBER(4)
);
```

```
CREATE OR REPLACE TRIGGER jurnalizare_pret_pachete_tgrr
AFTER UPDATE OF pret ON pachete_p FOR EACH ROW
BEGIN
INSERT INTO jurnalizare_pret_pachete VALUES (USER, SYSDATE,
:OLD.pret , :NEW.pret );

END;
```

Se testeaza functionarea declansatorului:

```
UPDATE pachete_p SET pret=2000 WHERE id_pachet=1132;
SELECT * FROM jurnalizare_pret_pachete;
```

UTILIZATOR	DATA	PRET_VECHI	PRET_NOU
APEX_PUBLIC_USER	24-Dec-2020	990	2000

1 rows returned in 0.00 seconds [Download](#)

4. Cand se adauga un angajat nou se doreste sa se jurnalizeze in ce data s-a adaugat, id-ul angajatului adaugat si sa nu se permita adaugarea daca salariul angajatului nou adaugat este mai mare decat cel mai mare salariu din departamentul in care este adaugat.

```
CREATE TABLE jurnalizare_angajari (
utilizator VARCHAR2(30),
id_angajat NUMBER(3),
data DATE,
salariu NUMBER(4)
);
```

```
CREATE OR REPLACE TRIGGER jurnalizare_angajari_tgrr
BEFORE INSERT ON angajati_p FOR EACH ROW
DECLARE
```



```

salariu_maxim angajati_p.salariu%TYPE;
BEGIN
SELECT MAX(salariu) INTO salariu_maxim FROM angajati_p
WHERE id_dpt=:NEW.id_dpt;

IF :NEW.salariu>salariu_maxim THEN
    RAISE_APPLICATION_ERROR(-20203,'Nu este permisa
adaugarea unui nou angajat cu salariul mai mare decat
'||salariu_maxim);
END IF;

INSERT INTO jurnalizare_angajari VALUES(USER,
:NEW.id_angajat, SYSDATE, :NEW.salariu);

END;

```

Vom testa functionalitatea declansatorului:

```

BEGIN
pat.angajezi('Kotai','Melinda','0746342020','STRADA
VIITORULUI',9000,31,118);
END;

```

```

ORA-20203: Nu este permisa adaugarea unui nou angajat cu salariul mai mare decat
7000
ORA-06512: at "RO_A178_PLSQL_S17.JURNALIZARE_ANGAJARI_TGRR", line 7
ORA-04088: error during execution of trigger
"RO_A178_PLSQL_S17.JURNALIZARE_ANGAJARI_TGRR"

```

```

BEGIN
pat.angajezi('Kotai','Melinda','0746342120','STRADA
VIITORULUI',4000,31,118);
END;

```

```

select * from jurnalizare_angajari;

```

UTILIZATOR	ID_ANGAJAT	DATA	SALARIU
APEX_PUBLIC_USER	129	24-Dec-2020	4000

1 rows returned in 0.00 seconds [Download](#)

5. Se doreste sa se jurnalizeze cine se logheaza la baza de date si cand.

```
CREATE OR REPLACE TRIGGER jurnalizare_logari AFTER  
LOGON ON DATABASE  
BEGIN  
INSERT INTO logari VALUES (USER, SYSDATE)  
END;
```

```
CREATE TABLE logari(  
user_id VARCHAR2(30),  
logon_date DATE );
```

(pentru aceasta comanda este necesara instalarea aplicatiei desktop,  
in apex se va afisa `insufficient privileges`)