

# Sistem de Gestionare a Tichetelor

Student: Fodor Marius Ioan



Baza de date are ca scop: **achiziționarea** unor cantități de tichete (în funcție de numărul de locuri disponibile), **tipul** de bilete ales de client, **validarea** ticketului, **modificarea/editarea** unor date la opțiunea clientului(nr. telefon, adresa email), precum și **ștergerea** clientului din baza de date.

În urma achiziției, clientul este înscris automat în tragerea la sorți. Aplicația/programul selectează un număr aleatoriu, utilizând modulul „random”, iar pe baza acestuia, clientul poate câștiga un premiu, în limita stocului disponibil.

#### Entități:

- *Stoc Bilete Cumpărate* (Tabelul principal): stochează informații referitoare la numărul de bilete achiziționate pe o **singură** tranzacție efectuată de client, tranzacția fiind identificată (**unic**) în serie\_ticket, totodată conținând informații referitoare la validarea biletului, data achiziționării, nr. bilete achiziționate și suma platită de client.
- *Evidență Clienți*: conține informații referitoare la nume, prenume, email, telefon (opțional), **NR. EXTRAS** (selectat aleatoriu de program pentru a participa la concurs), **CNP** (Cod Numeric Personal, fiecare client este înregistrat/identificat printr-un cnp unic).
- *Stoc Bilete*: stochează informații referitoare la prețul unui singur bilet și numărul de locuri disponibile pentru fiecare tip de bilet/ticket. Este populată de către organizator în funcție de capacitatea totală pusă la dispoziție pentru organizarea evenimentului. Astfel pentru tipul de bilete **premium**: preț: 200 RON/bilet, nr. de locuri 10, tip ticket **mid**: nr locuri 20, preț 100 RON/bilet, tip ticket **low**: nr. locuri 30 preț 50 RON/bilet.
- *Evidență câștigători*: Ține evidența clienților care au câștigat un premiu.
- *Stoc Premii*: este populată de organizator și conține informații despre fiecare premiu disponibil. Fiecare premiu este identificat printr-o cantitate disponibilă și o valoare numerică precisă, care trebuie să corespundă exact cu numărul ales din program pentru a putea fi validat și atribuit clientului. Pentru premiul un autovehicul electric marca “Tesla” este atribuit valoarea numerică 1 pentru a fi validat câștigător, cantitate disponibilă 1 buc. Pentru premiul telefon mobil marca “Samsung S25” valorile numerice sunt 10 sau 20 cantitate 10 buc.



### Relațiile între entități:

- 1:1 între Evidență clienți și Stoc bilete cumpărate - un client poate face o singură tranzacție (indiferent de numărul de bilete cumpărate), astfel un client înregistrat în Evidență Clienți, pe bază de cnp unic este înregistrat pe baza tranzacției în Stoc Bilete Cumpărate, unde cheia străină FK\_IDEVIDENTA\_CLIENTI moștenește cheia primară PK\_IDEVIDENTA\_CLIENTI autoincrementată din Evidență Clienți.
- 1:N între Stoc bilete și Stoc bilete cumpărate – un tip de bilet poate fi cumpărat de mai mulți clienți, dar fiecare bilet cumpărat provine din numărul de locuri disponibile (cantitate tickete disponibile). Cheia străină FK\_IDSTOC\_BILETE din Stoc bilete cumpărate moștenește cheia primară PK\_IDSTOC\_BILETE tipului de bilet cumpărat.
- 1:1 între Evidență Câștigători și Stoc bilete cumpărate – un client pe baza tranzacționării are șansa de a participa o singură dată la concurs. Dacă numărul selectat de program corespunde, premiul este validat clientului, unde înregistrarea se va face astfel: cheia străină FK\_IDSTOC\_BILETE\_CUMPARATE din Evidență Câștigători va moșteni cheia străină FK\_IDEVIDENTA\_CLIENTI din Stoc bilete cumpărate. Cheia străină FK\_IDSTOC\_PREMII va moșteni cheia primară PK\_IDSTOC\_PREMII din Stoc premii pentru validarea premiului.
- 1:N între Stoc premii și Evidență Câștigători – un premiu poate fi câștigat de mai mulți clienți dar fiecare premiu provine dintr-o anumită cantitate disponibilă de premii.



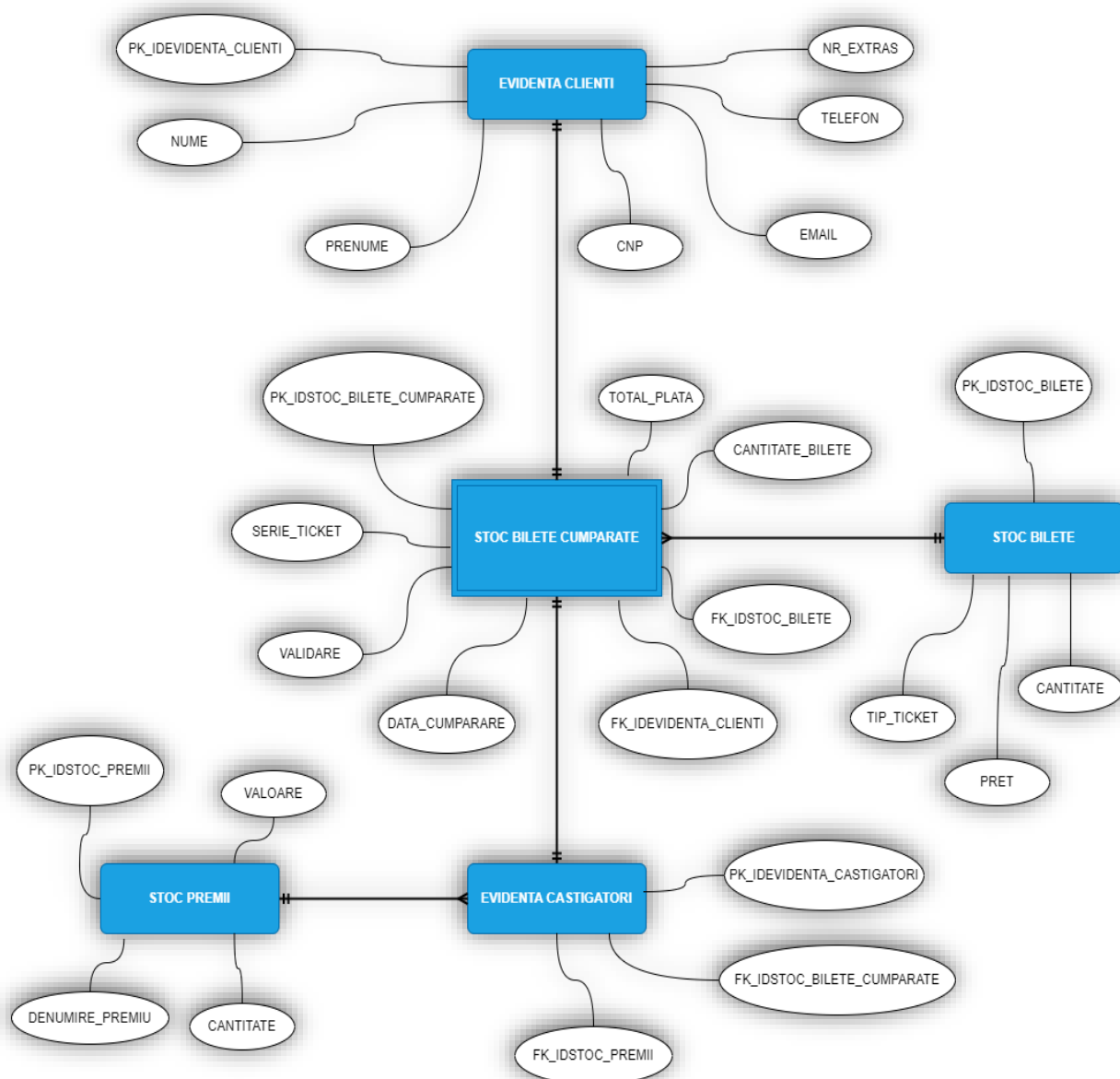


Figura 1 Diagrama Bachman



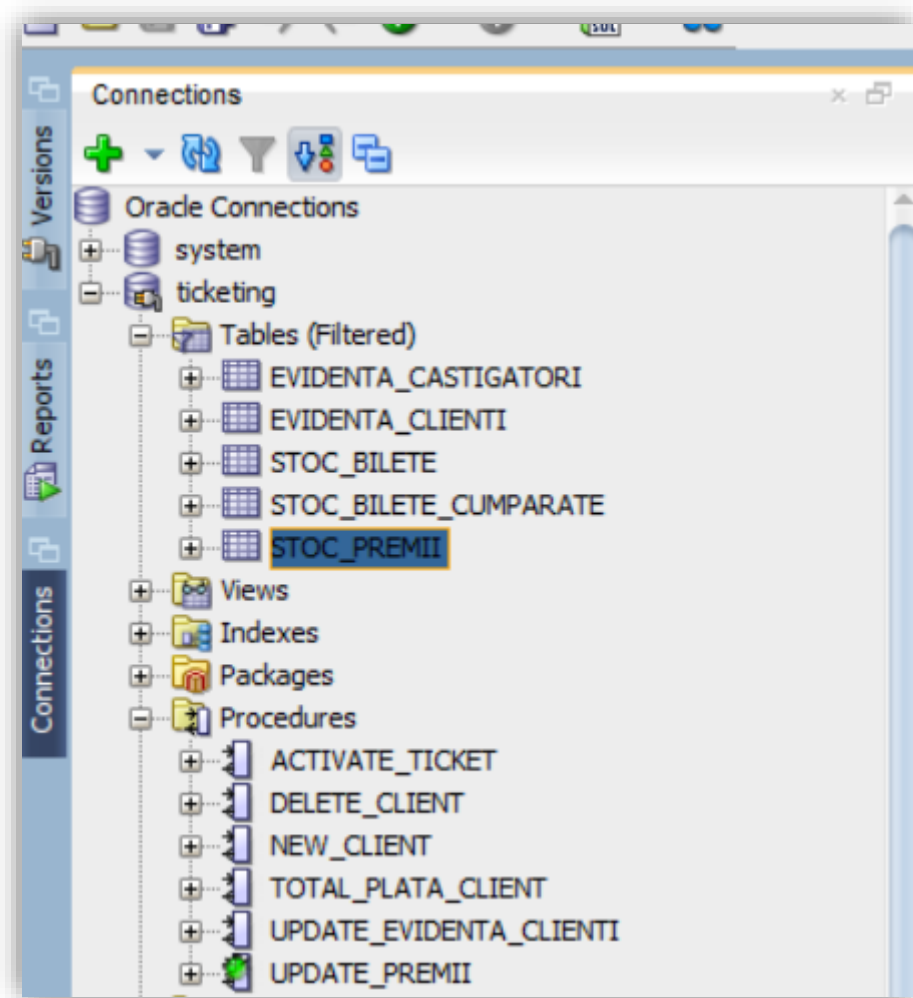


Figura 2 Entitățile bazei de date.

EVIDENTA_CLIENTI							
Columns   Data   Model   Constraints   Grants   Statistics   Triggers   Flashback   Dependencies   Details   Partitions   Indexes   SQL							
	PK_IDEVIDENTA_CLIENTI	NUME	PRENUME	CNP	EMAIL	TELEFON	NR_EXTRAS
1	145	testnume	testprenume	1234	test1@test1	(null)	4
2	144	fodor	marius	123	test@test	(null)	15

În Figura 3 sunt prezentate două inserări în Evidență Clienți



PK_IDSTOC_BILETE_CUMPARATE	SERIE_TICKET	VALIDARE	DATA_CUMPARARE	FK_IDEVIDENTA_CLIENTI	FK_IDSTOC_BILETE	CANTITATE_BILETE	TOTAL_PL...
1	145 K6P-E2F-F2F	0	02-FEB-25	145	2	2	200
2	144 E7X-V8V-02Z	0	02-FEB-25	144	1	1	200

Figura 4 cu cele două intrări și cheile moștenite.

astigator	samsung	1234567	numar20	(null)	20
odor	marius	123	test@test	(null)	15
esla	castigator	12345	numar1	(null)	1
samsung	castigator	123456	numar10	(null)	10

Buy Ticket

Activate

Edit

Nume \* sadasd

Prenume \* dsaasd

CNP \* 123

Email \* dsdd

Telefon

Acest CNP 123 exista in baza de date

Figura 5 Exemplu interogare Evidenta Client atribut CNP

BILETE	TIP_TICKET	PRET	CANTITATE
1	premium	200	7
2	mid	100	17
3	low	50	29

Buy Ticket

Activate

Edit

Nume \* sadasd

Prenume \* dsaasd

CNP \* aaaa

Email \* dsdd

Telefon

Selecteaza Tip Ticket

premium

Pret: 200 RON

Introdu nr. bilete 7

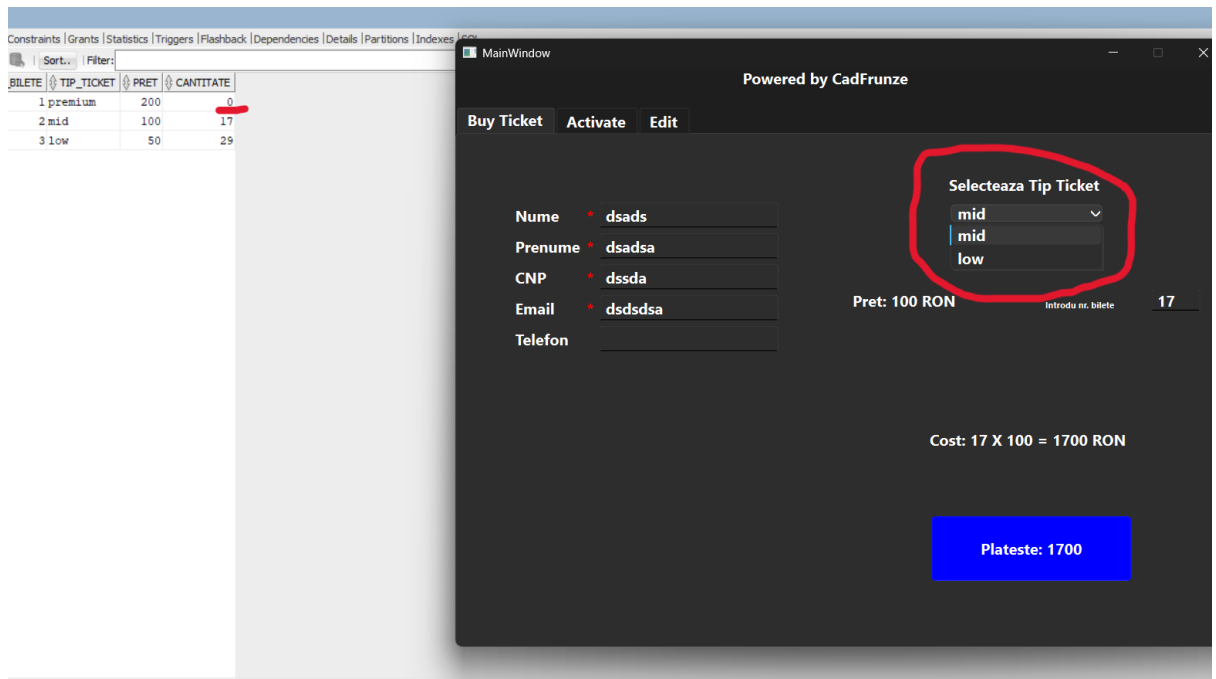
Cost: 7 X 200 = 1400 RON

Plateste: 1400

Figura 6 Exemplu Interogare Stoc Bilete (tip ticket, preț și cantitate bilete/nr locuri rămase)

Student: Fodor Marius Ioan





În Figura 7 interogare după cantitate bilete (nr. locuri disponibile)

### Exemplu de validare ticket

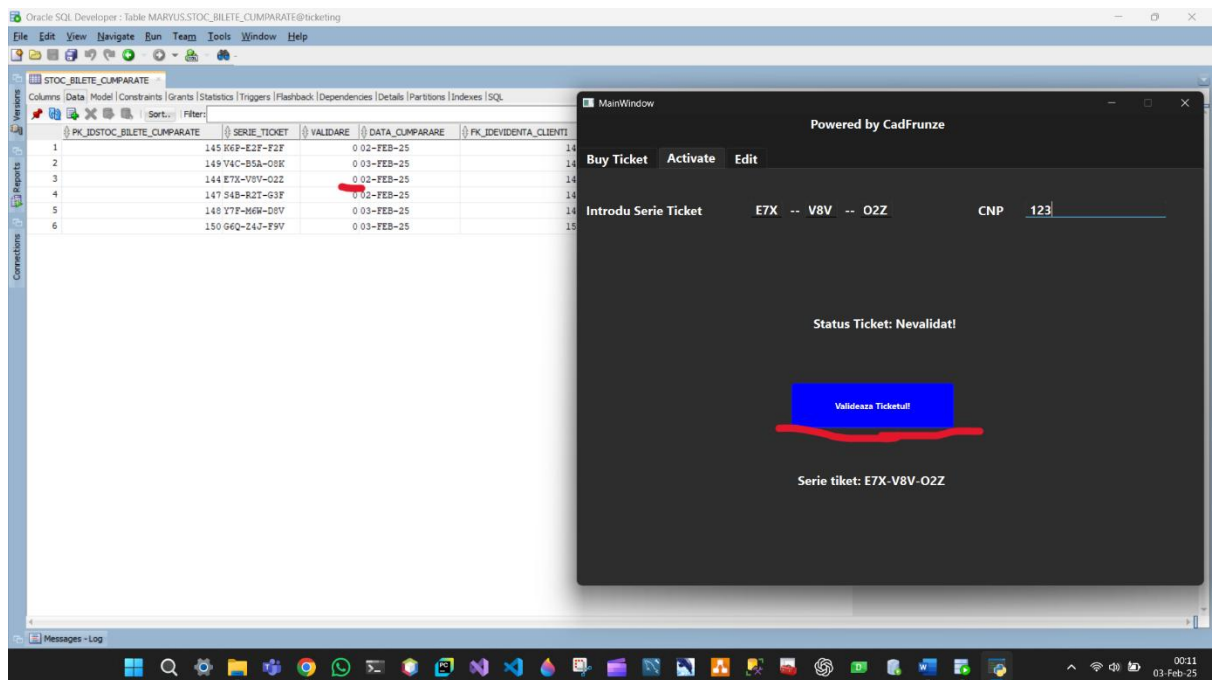


Figura 8 Înainte de validare.



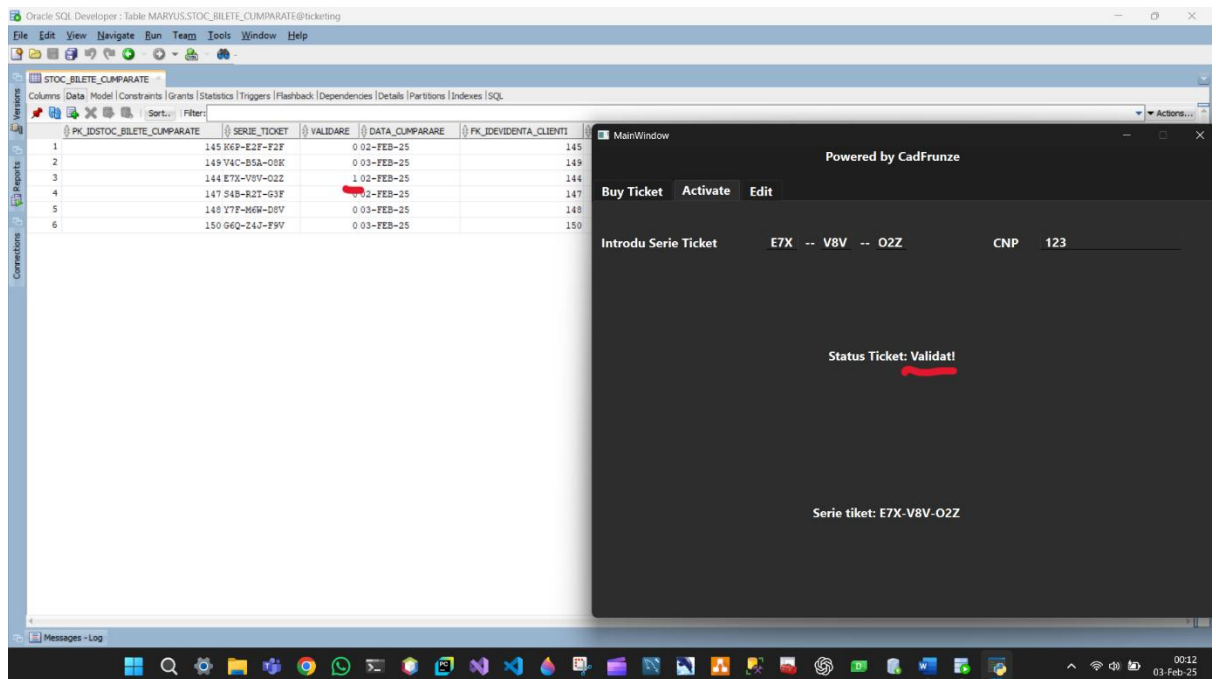


Figura 9 După validare.

## Proceduri

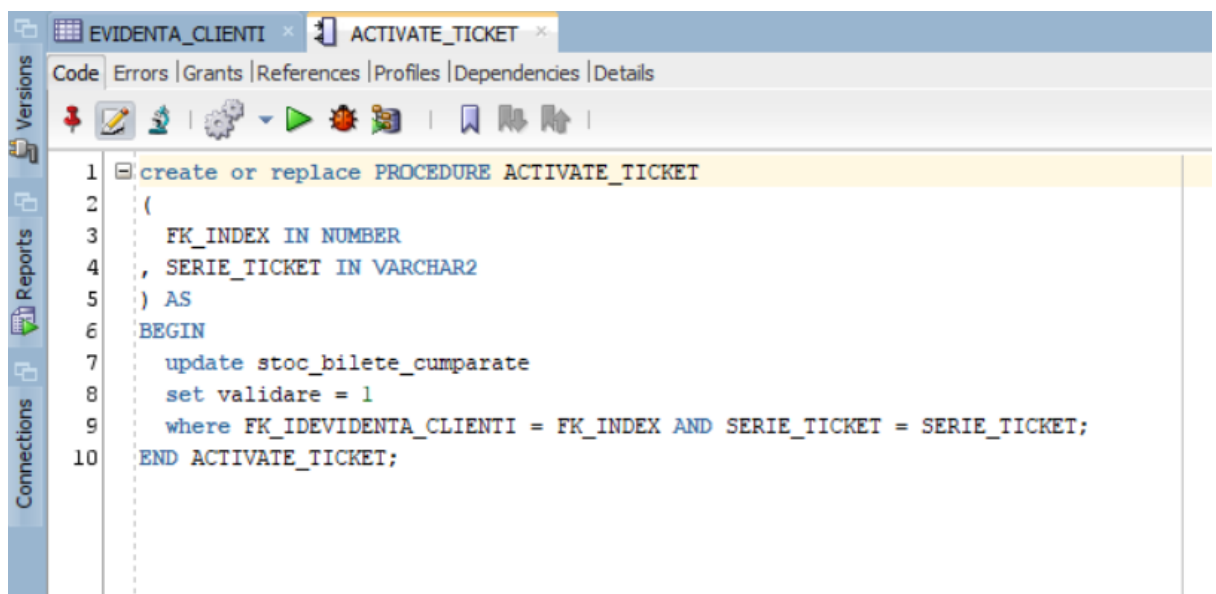


Figura 10 Validare ticket.





```

1 create or replace PROCEDURE DELETE_CLIENT
2 (
3     INDEX_KEY IN NUMBER
4 ) AS
5 BEGIN
6
7     DELETE EVIDENTA_CASTIGATORI
8     WHERE FK_IDSTOC_BILETE_CUMPARATE = INDEX_KEY;
9
10    DELETE STOC_BILETE_CUMPARATE
11    WHERE PK_IDSTOC_BILETE_CUMPARATE = INDEX_KEY;
12
13    DELETE EVIDENTA_CLIENTI
14    WHERE PK_IDEVIDENTA_CLIENTI = INDEX_KEY;
15
16 END DELETE_CLIENT;

```

Figura 11 Ștergere client din baza de date.

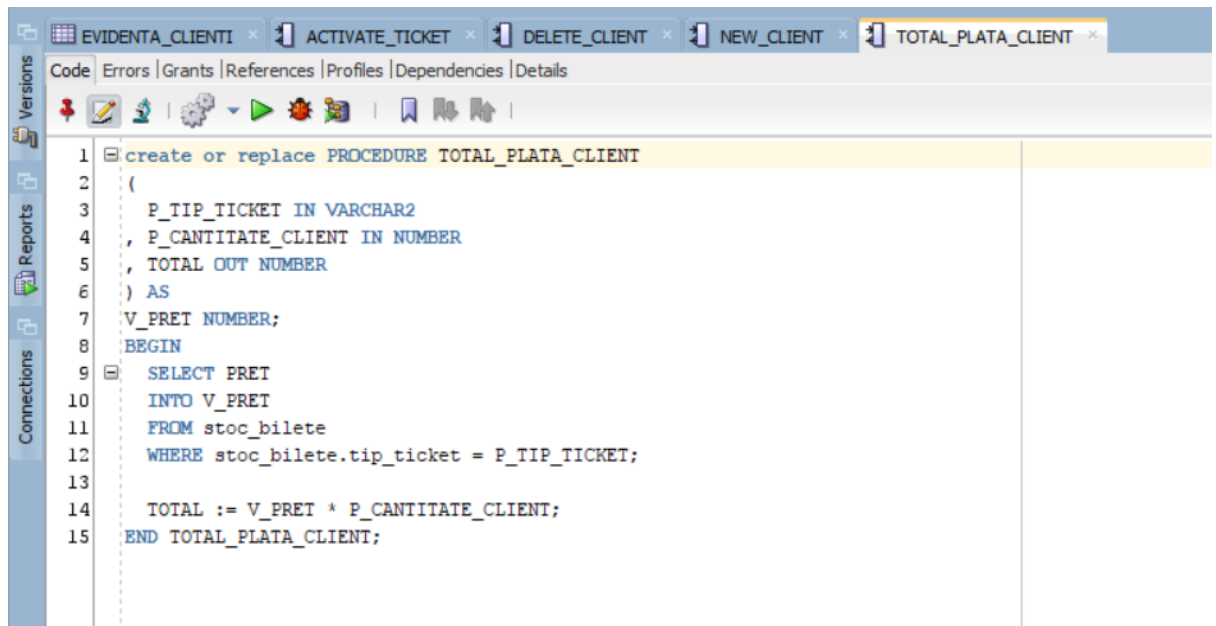
```

1 create or replace PROCEDURE NEW_CLIENT
2 (
3     --EVIDENTA_CLIENTI--
4     P_NUME IN VARCHAR2
5     , P_PRENUME IN VARCHAR2
6     , P_CNP IN VARCHAR2
7     , P_EMAIL IN VARCHAR2
8     , P_TELEFON IN VARCHAR2
9     , P_NR_EXTRAS IN NUMBER
10    , P_TIP_TICKET IN VARCHAR2
11    , P_SERIE_TICKET IN VARCHAR2
12    , P_VALIDARE IN VARCHAR2
13    , P_CANTITATE_BILETE IN NUMBER
14 ) AS
15
16     V_IDCLIENT EVIDENTA_CLIENTI.PK_IDEVIDENTA_CLIENTI%TYPE; --ORIN ID-UL DIN EVIDENTA_CLIENTI
17     V_IDSTOC_BILETE STOC_BILETE.pk_idstoc_bilete%type; -- STOCHEZ IDUL DIN STOC_BILETE
18     V_TOTAL NUMBER; --CALCUL NR_BILETE_CLIENT * PRETUL DIN STOC_BILETE
19
20 BEGIN
21     INSERT INTO EVIDENTA_CLIENTI (nume, prenume, cnp, email, telefon, nr_extras)
22     VALUES (P_NUME, P_PRENUME, P_CNP, P_EMAIL, P_TELEFON, P_NR_EXTRAS);
23     RETURN PK_IDEVIDENTA_CLIENTI INTO V_IDCLIENT;
24     COMMIT;
25
26     total_plata_client(P_TIP_TICKET, P_CANTITATE_BILETE, V_TOTAL); --PROCEDURA DE CALCUL
27
28     UPDATE stoc_bilete
29     SET cantitate = cantitate - P_CANTITATE_BILETE
30     WHERE tip_ticket = P_TIP_TICKET;
31
32     SELECT pk_idstoc_bilete INTO V_IDSTOC_BILETE
33     FROM stoc_bilete
34     WHERE tip_ticket = P_TIP_TICKET;

```

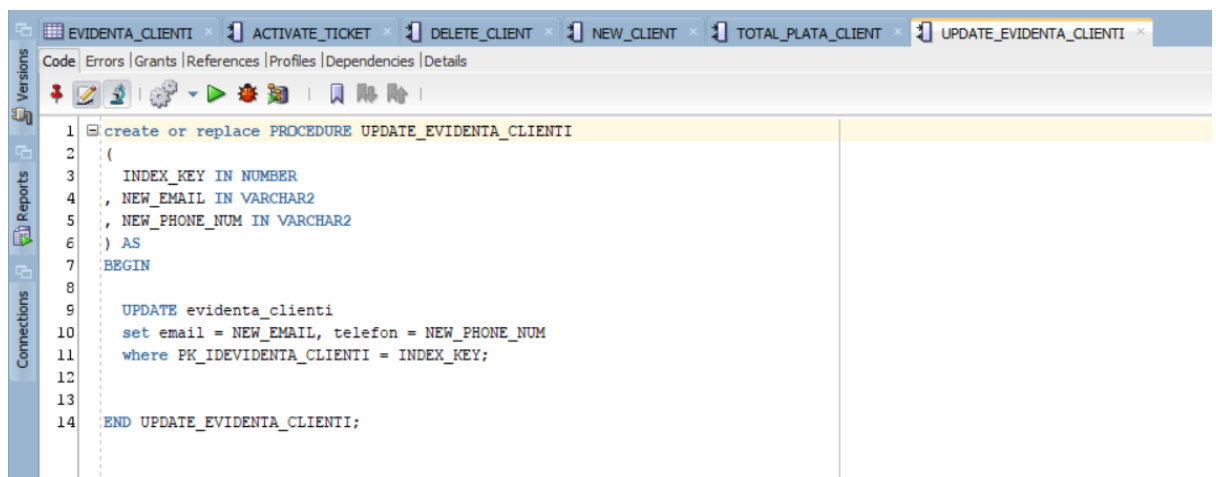
Figura 12 Înregistrare client nou.





```
1 create or replace PROCEDURE TOTAL_PLATA_CLIENT
2 (
3     P_TIP_TICKET IN VARCHAR2
4     , P_CANTITATE_CLIENT IN NUMBER
5     , TOTAL OUT NUMBER
6 ) AS
7 V_PRET NUMBER;
8 BEGIN
9     SELECT PRET
10    INTO V_PRET
11    FROM stoc_bilete
12    WHERE stoc_bilete.tip_ticket = P_TIP_TICKET;
13
14    TOTAL := V_PRET * P_CANTITATE_CLIENT;
15 END TOTAL_PLATA_CLIENT;
```

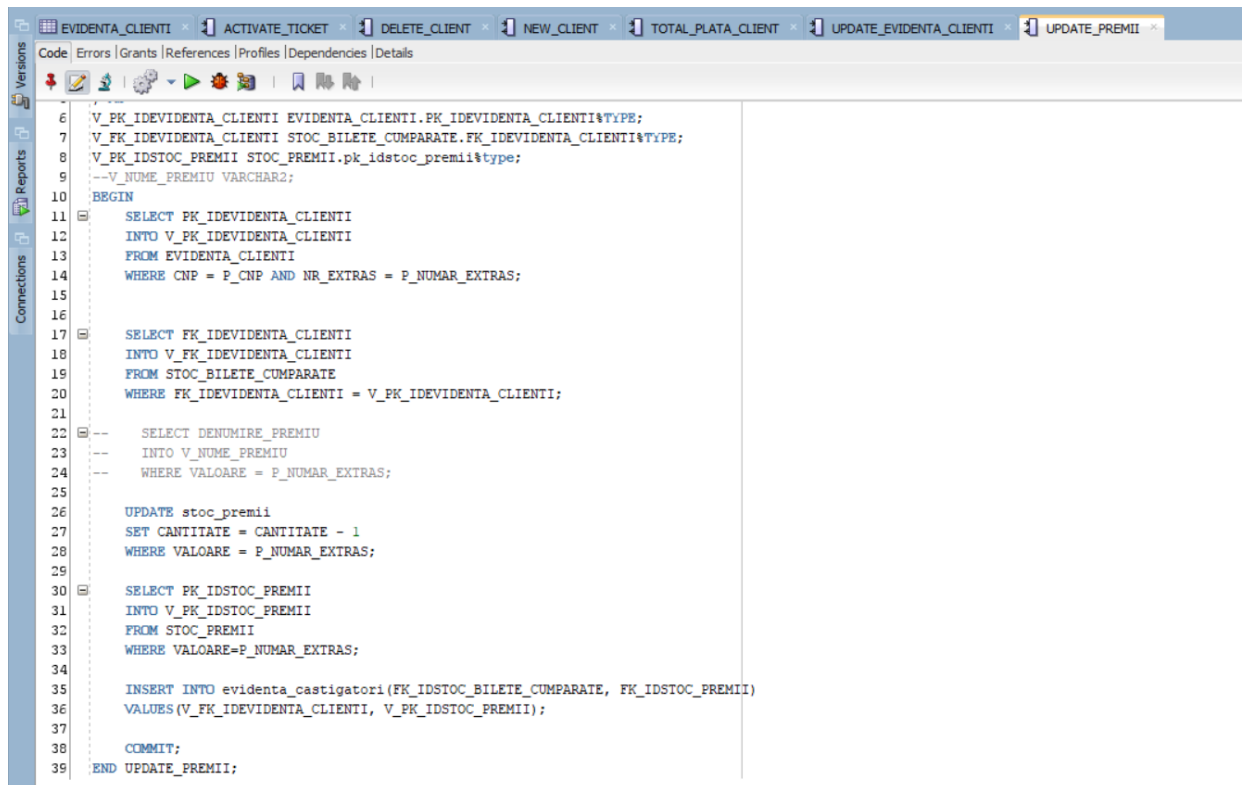
Figura 13 Total plată client = preț ticket x nr. tickete comandate



```
1 create or replace PROCEDURE UPDATE_EVIDENTA_CLIENTI
2 (
3     INDEX_KEY IN NUMBER
4     , NEW_EMAIL IN VARCHAR2
5     , NEW_PHONE_NUM IN VARCHAR2
6 ) AS
7 BEGIN
8
9     UPDATE evidenta_clienti
10    set email = NEW_EMAIL, telefon = NEW_PHONE_NUM
11    where PK_IDEVIDENTA_CLIENTI = INDEX_KEY;
12
13
14 END UPDATE_EVIDENTA_CLIENTI;
```

Figura 14 Modificare date ale clientului.





```
6 V_PK_IDEVIDENTA_CLIENTI EVIDENTA_CLIENTI.FK_IDEVIDENTA_CLIENTI%TYPE;  
7 V_FK_IDEVIDENTA_CLIENTI STOC_BILETE_CUMPARATE.FK_IDEVIDENTA_CLIENTI%TYPE;  
8 V_PK_IDSTOC_PREMII STOC_PREMII.pk_idstoc_premii%type;  
9 --V_NUME_PREMIU VARCHAR2;  
10 BEGIN  
11     SELECT PK_IDEVIDENTA_CLIENTI  
12     INTO V_PK_IDEVIDENTA_CLIENTI  
13     FROM EVIDENTA_CLIENTI  
14     WHERE CNP = P_CNP AND NR_EXTRAS = P_NUMAR_EXTRAS;  
15  
16  
17     SELECT FK_IDEVIDENTA_CLIENTI  
18     INTO V_FK_IDEVIDENTA_CLIENTI  
19     FROM STOC_BILETE_CUMPARATE  
20     WHERE FK_IDEVIDENTA_CLIENTI = V_PK_IDEVIDENTA_CLIENTI;  
21  
22     -- SELECT DENUMIRE_PREMIU  
23     -- INTO V_NUME_PREMIU  
24     -- WHERE VALOARE = P_NUMAR_EXTRAS;  
25  
26     UPDATE stoc_premii  
27     SET CANTITATE = CANTITATE - 1  
28     WHERE VALOARE = P_NUMAR_EXTRAS;  
29  
30     SELECT PK_IDSTOC_PREMII  
31     INTO V_PK_IDSTOC_PREMII  
32     FROM STOC_PREMII  
33     WHERE VALOARE=P_NUMAR_EXTRAS;  
34  
35     INSERT INTO evidenta_castigatori(FK_IDSTOC_BILETE_CUMPARATE, FK_IDSTOC_PREMII)  
36     VALUES(V_FK_IDEVIDENTA_CLIENTI, V_PK_IDSTOC_PREMII);  
37  
38     COMMIT;  
39 END UPDATE_PREMII;
```

Figura 15 Modificare cantitate premiu și înregistrarea în Evidență Câștigători.

Pentru mai multe exemple vizitați :

<https://github.com/cadfrunze/ticketing/tree/master/documentatie/capturi>

