

## Laborator 2

### Comanda CREATE TABLE cu constrangeri. Modificarea structurii unei tabele

#### Problema rezolvata:

Se considera baza de date XE, ca fiind baza de date a unei biblioteci.

1. Sa se creeze o tabela, numita *Evidenta\_carti*, avand urmatoarea structura:

- *Nr\_inventar* (numarul de inventar al cartilor) – integer,
- *Titlu* ( titlul cartilor) - varchar (30),
- *Autor* (numele complet al autorului) - varchar (30),
- *Pagini* (numarul de pagini) - integer,
- *Editura* (numele complet al editurii) - varchar(30),
- *Datap* (data publicarii) – date,
- *Pret* (pretul cartii) - number (5,2),
- *Imprumutat* (starea cartii –DA/NU) – char(2) .

#### Observatii:

- Se cere aplicarea constrangerii Primary Key pe campul *Nr\_inventar*;
- Valoarea care poate fi introdusa in campul *Nr\_inventar* trebuie sa fie cuprinsa in intervalul (0, 100000). Se cere folosirea constrangerii CHECK;
- Titlul cartii trebuie sa fie unic si diferit de NULL. Ca urmare, se cere aplicarea constrangerilor UNIQUE si NOT NULL;
- Campurile *Autor* si *Pagini* trebuie sa fie diferite de NULL. Ca urmare, se cere aplicarea constrangerii NOT NULL;
- Implicit, cartile din tabela se considera neimprumutate. Ca urmare, valoarea implicita din campul *Imprumutat* este NU.

```
CREATE TABLE Evidenta_carti(  
    Nr_inventar integer PRIMARY KEY CHECK (Nr_inventar between 0 and 100000),  
    Titlu varchar(30) UNIQUE NOT NULL,  
    Autor varchar (30) NOT NULL,  
    Pagini integer NOT NULL,  
    Editura varchar (30),  
    Datap date,  
    Pret number (5,2),  
    Imprumutat char(2) DEFAULT 'NU');
```

2. Se cere popularea tablei *Evidenta\_carti* cu urmatoarele articole:

Nr_inventar	Titlu	Autor	Pagini	Editura	Datap	Pret	Imprumutat
1	Povesti	Ion Creanga	200	Teora	11.02.2005	10	NU
2	Basme	Petre Ispirescu	100	Teora	12.03.2001	20.50	NU
3	Poezii	Mihai Eminescu	50	Caro	22.02.2012	15	NU
4	Mara	Ion Slavici	70	Mirton	15.04.2010	40	NU
5	Ion	Liviu Rebreanu	100		10.03.2007		NU
6	Poeme	Ion Barbu	200	BPT		55.03	NU

```

SELECT sysdate FROM dual;
alter session set nls_date_format='dd-mm-yyyy';

set autocommit on;

INSERT INTO Evidenta_carti
(Nr_inventar, Titlu, Autor, Pagini, Editura, Datap, Pret)
VALUES(1, 'Povesti', 'Ion Creanga', 200, 'Teora', '11.02.2005', 10);

INSERT INTO Evidenta_carti
(Nr_inventar, Titlu, Autor, Pagini, Editura, Datap, Pret)
VALUES(2, 'Basme', 'Petre Ispirescu', 100, 'Teora', '12.03.2001',
20.50);

INSERT INTO Evidenta_carti
(Nr_inventar, Titlu, Autor, Pagini, Editura, Datap, Pret)
VALUES(3, 'Poezii', 'Mihai Eminescu', 50, 'Caro', '22.02.2012', 15);

INSERT INTO Evidenta_carti
(Nr_inventar, Titlu, Autor, Pagini, Editura, Datap, Pret)
VALUES(4, 'Mara', 'Ion Slavici', 70, 'Mirton', '15.04.2010', 40);

INSERT INTO Evidenta_carti
(Nr_inventar, Titlu, Autor, Pagini, Editura, Datap, Pret)
VALUES(5, 'Ion', 'Liviu Rebreanu', 100, NULL, '10.03.2007', NULL);

INSERT INTO Evidenta_carti
(Nr_inventar, Titlu, Autor, Pagini, Editura, Datap, Pret)
VALUES(6, 'Poeme', 'Ion Barbu', 200, 'BPT', NULL, 55.03);

```

3. Sa se afiseze tot continutul tabelii *Evidenta\_carti*;

```
SELECT * FROM Evidenta_carti;
```

4. Sa se majoreze cu 10% pretul tuturor cartilor din tabela;

```
UPDATE Evidenta_carti SET pret = 1.1 * pret;
```

5. Sa se afiseze o lista cu toate editurile, fara dubluri si fara a afisa valoarea NULL;

```
SELECT DISTINCT editura FROM Evidenta_carti
WHERE editura IS NOT NULL;
```

6. Sa se incerce introducerea unei carti cu urmatoarele caracteristici, omitand precizarea *Nr\_inventar*:

Titlu	Autor	Pagini	Editura	Datap	Pret	Imprumutat
HTML	Jhon Doe	150	Mirton	15.02.2000	20	NU

```

INSERT INTO evidenta_carti (Titlu,Autor,Pagini,Editura,Datap,Pret)
VALUES ('HTML','Jhon Doe', 150, 'Mirton', '15.02.2000', 20);

```

Se va semnala eroare:

EROARE la linia 1:

ORA-01400: nu poate fi inserat NULL în ("SYS"."EVIDENTA\_CARTI"."NR\_INVENTAR")

7. Sa se incerce introducerea unei carti cu urmatoarele caracteristici:

Nr_inventar	Titlu	Autor	Pagini	Editura	Datap	Pret	Imprumutat
4	HTML	Jhon Doe	150	Mirton	15.02.2000	20	NU

```
INSERT INTO evidenta_carti
(Nr_inventar, Titlu, Autor, Pagini, Editura, Datap, Pret)
VALUES (4, 'HTML', 'Jhon Doe', 150, 'Mirton', '15.02.2000', 20);
```

Se va semnala eroare:

EROARE la linia 1:

ORA-00001: regula constrângerii unice (SYS.NR\_INV) nu este respectata

8. Sa se adauge constrangerea CHECK pe campul *Imprumutat*, astfel incat in acest camp sa nu fie acceptate decat valorile 'DA' sau 'NU';

```
ALTER TABLE Evidenta_carti ADD CONSTRAINT imprumut
CHECK (Imprumutat IN ('DA', 'NU'));
```

9. Sa se incerce introducerea unei carti cu urmatoarele caracteristici:

Nr_inventar	Titlu	Autor	Pagini	Editura	Datap	Pret	Imprumutat
7	PHP	Ioan Filip	200	Teora	15.02.2007	30	X

```
INSERT INTO evidenta_carti
(Nr_inventar, Titlu, Autor, Pagini, Editura, Datap, Pret, Imprumutat)
VALUES (7, 'PHP', 'Ioan Filip', 200, 'Teora', '15.02.2007', 30, 'X');
```

Se va semnala eroare:

EROARE la linia 1:

ORA-12899: regula constrângerii CHECK nu este respectata

10. Sa se modifice tipul de data al campului *Titlu*, din varchar(30) in varchar(40);

```
ALTER TABLE Evidenta_carti MODIFY Titlu varchar(40);
```

11. Sa se adauge o coloana numita *Gen* in cadrul tabelii, si sa se completeze cu valoarea 'beletristica' pentru toate cele 6 articole existente. Tipul de data al coloanei *Gen* este varchar(30);

```
ALTER TABLE Evidenta_carti ADD Gen varchar(30);
UPDATE Evidenta_carti SET Gen='beletristica';
```

sau:

```
ALTER TABLE Evidenta_carti ADD Gen varchar(30) DEFAULT 'beletristica';
```

12. Sa se introduca doua carti cu urmatoarele caracteristici:

Nr_inventar	Titlu	Autor	Pagini	Editura	Datap	Pret	Imprumutat	Gen
7	HTML	Jhon Doe	150	Mirton	15.02.2000	20	NU	tehnica
8	PHP	Ioan Filip	200	Teora	15.02.2007	30	NU	tehnica

```

INSERT INTO evidenta_carti
(Nr_inventar, Titlu, Autor, Pagini, Editura, Datap, Pret, Gen)
VALUES (7, 'HTML', 'Jhon
Doe', 150, 'Mirton', '15.02.2000', 20, 'tehnica');

INSERT INTO evidenta_carti
(Nr_inventar, Titlu, Autor, Pagini, Editura, Datap, Pret, Gen)
VALUES (8, 'PHP', 'Ioan
Filip', 200, 'Teora', '15.02.2007', 30, 'tehnica');

```

13. Sa se afiseze toate cartile care au fost publicate inainte de anul 2008;

```
SELECT * FROM Evidenta_carti WHERE datap < '01.01.2008';
```

14. Sa se afiseze titlul si autorul cartilor care au fost publicate dupa anul 2000, din genul 'beletristica';

```
SELECT Titlu, Autor FROM Evidenta_carti
WHERE Datap > '01.01.2000' AND Gen = 'beletristica';
```

15. Sa se afiseze titlul si autorul cartilor care au fost publicate dupa anul 2000, din genul 'beletristica', ordonate dupa Titlu;

```
SELECT Titlu, Autor FROM Evidenta_carti
WHERE Datap > '01.01.2000' AND Gen = 'beletristica'
ORDER BY Titlu;
```

16. Sa se creeze tabela *Beletristica*, cu exact aceeasi structura cu a tabelii *Evidenta\_carti*, de dinaintea adaugarii coloanei *Gen*, dar fara a contine nicio linie (deci fara copierea datelor din tabela originala).

```

CREATE TABLE Beletristica(
  Nr_inventar integer PRIMARY KEY CHECK (Nr_inventar between 0 and 100000),
  Titlu varchar(30) UNIQUE NOT NULL,
  Autor varchar (30) NOT NULL,
  Pagini integer NOT NULL,
  Editura varchar (30),
  Datap date,
  Pret number (5,2),
  Imprumutat char(2) DEFAULT 'NU');

```

Sau

```

CREATE TABLE Beletristica AS
SELECT Nr_inventar, Titlu, Autor, Pagini, Editura, Datap, Pret, Imprumutat
FROM Evidenta_carti
WHERE 1=0;

```

17. Sa se afiseze structura tabelii Beletristica;

```
DESCRIBE Beletristica;
```

18. Sa se mute toate cartile de genul 'beletristica' din tabela *Evidenta\_carti* in tabela *Beletristica*;

```

INSERT INTO Beletristica
SELECT Nr_inventar, Titlu, Autor, Pagini, Editura, Datap, Pret, Imprumutat

```

```
FROM Evidenta_carti  
WHERE Gen='beletristica';  
DELETE FROM Evidenta_carti  
WHERE Gen='beletristica';
```

19. Sa se creeze o noua tabela *Tehnica*, care sa aiba aceeasi structura ca si tabela *Evidenta\_carti*, de dinaintea adaugarii coloanei *Gen* si sa contina toate cartile din tabela *Evidenta\_carti* care au genul 'tehnica';

```
CREATE TABLE Tehnica AS  
SELECT Nr_inventar, Titlu, Autor, Pagini, Editura, Datap, Pret, Imprumutat  
FROM Evidenta_carti  
WHERE Gen='tehnica';
```

20. Sa se afiseze structura tabelii *Tehnica*;

```
DESCRIBE Tehnica;
```

21. Sa se afiseze continutul tabelii *Tehnica*;

```
SELECT * FROM Tehnica;
```

22. Sa se stearga coloana *Gen* din tabela *Evidenta\_carti*;

```
ALTER TABLE Evidenta_carti DROP COLUMN Gen;
```

23. Sa se stearga cele 3 tabele create.

```
DROP TABLE Evidenta_carti;  
DROP TABLE Beletristica;  
DROP TABLE Tehnica;
```

## Problema propusa:

Se considera baza de date XE, ca fiind baza de date a unui magazin de electrocasnice.

Sa se creeze o tabela, numita *Evidenta\_produce*, avand urmatoarea structura:

- *Cod\_produ*s – varchar(10),
- *Pro*du*s* - varchar (30),
- *Mar*ca – varchar (20),
- *Stoc* – integer,
- *Pret* - number (7,2),
- *Discount* – number (3,2).

### Observatii pentru crearea tabeli:

- Se cere aplicarea constrangerii Primary Key pe campul *Cod\_produ*s;
- Campurile *Pro*du*s*, *Mar*ca, *Stoc*, *Pret* si *Discount* trebuie sa fie diferite de NULL. Ca urmare, se cere aplicarea constrangerii NOT NULL;
- Valoarea care poate fi introdusa in campul *Pret* trebuie sa fie cuprinsa in intervalul (0, 10000). Se cere folosirea constrangerii CHECK;
- Valoarea implicita pentru campul *Discount* este 0. Se va folosi atributul DEFAULT.

1. Se cere popularea tabeli *Evidenta\_produce* cu urmatoarele articole:

Cod_produ	Pro	Mar	Stoc	Pret	Discount
DX11	Frigider	Bosch	40	1680.99	0
DX12	Frigider	Electrolux	30	1730.00	0
DX13	Frigider	Whirlpool	30	1500.99	0
PX11	Cuptor	Arctic	10	1000.00	0
PX12	Cuptor	Samsung	20	1567.99	0
MX11	Plita	Gorenje	50	870.99	0
MX12	Plita	Electrolux	80	980.00	0
NX11	Masina de spalat vase	Gorenje	30	1799.00	0
NX12	Masina de spalat vase	Electrolux	50	2020.99	0
NX13	Masina de spalat vase	AEG	70	2257.99	0
CX11	Cuptor cu microunde	Bosch	90	1200.99	0
CX12	Cuptor cu microunde	Electrolux	30	1500.99	0

2. Sa se afiseze tot continutul tabeli *Evidenta\_produce*;
3. Sa se afiseze toate produsele de tip Whirlpool din tabela *Evidenta\_produce*;
4. Sa se actualizeze stocul produsului avand codul *DX11*, in urma vanzarii a 30 de bucati;
5. Sa se afiseze o lista cu toate marcile din tabela *Evidenta\_produce*, fara dubluri;

6. Sa se aplice un discount de 30% la toate produsele de tip *Masina de spalat vase* (actualizarea campului *discount* pentru acele produse se va face cu valoarea 0.3) si sa se afiseze pe ecran produsele respective si noul pret obtinut in urma aplicarii discount-ului (noul pret va fi egal cu  $Pret-Pret*Discount$ ), sub forma:

Cod_produs	Produs	Marca	Pret	Discount	Pret nou
NX11	Masina de spalat vase	Gorenje	1799.00	0.3	1259.3
NX12	Masina de spalat vase	Electrolux	2020.99	0.3	1414.69
NX13	Masina de spalat vase	AEG	2257.99	0.3	1580.59

7. Sa se adauge constrangerea CHECK pe campul *Discount*, astfel incat in acest camp sa nu fie acceptate decat valori cuprinse in intervalul  $[0,1]$ ;
8. Sa se incerce actualizarea campului *Discount* pentru produsele de tip *Frigider* cu valoarea 1.2 (120%). Motivati eroarea care va apare.
9. Sa se modifice tipul de data al campului *Produs*, din varchar(30) in varchar(40);
10. Sa se adauge in tabela *Evidenta\_produce* o coloana noua, numita *Garantie*, de tipul integer si sa se introduca in aceasta coloana, pentru toate produsele existente, numarul de ani pentru care se ofera garantie.
11. Sa se creeze tabela *Reduceri* cu exact aceeasi structura cu a tabelii *Evidenta\_produce*, dar fara a contine nicio linie (deci fara copierea datelor din tabela originala).
12. Sa se afiseze structura tabelii *Reduceri*;
13. Sa se mute toate produsele care sunt la reducere (care au un anumit discount) din tabela *Evidenta\_produce*, in tabela *Reduceri*;
14. Sa se afiseze continutul tabelii *Reduceri*;
15. Sa se stearga coloana Discount din tabela *Evidenta\_produce*;
16. Sa se afiseze structura tabelii *Reduceri*;
17. Sa se creeze o noua tabela *Frigidere*, care sa aiba aceeasi structura ca si tabela *Evidenta\_produce* si sa contina toate produsele de tip *Frigider* din tabela *Evidenta\_produce*;
18. Sa se afiseze structura tabelii *Frigidere*;
19. Sa se afiseze continutul tabelii *Frigidere*;
20. Sa se stearga cele 3 tabele create.