

Atestare Nr.2

1. Specificați și descrieți care sunt tipurile de date în Oracle?

- CHAR Datatype
- VARCHAR2 and VARCHAR Datatypes
- NCHAR and NVARCHAR2 Datatypes
- LOB Character Datatypes
- LONG Datatype

2. Cum modificăm structura unei baze de date în SQL? Dați un exemplu.

Cu ajutorul comenzii **ALTER TABLE** putem realiza modificări în tabele, dar să adăugăm înregistrări și să efectuăm ștergeri de atribute.

Exemplu :

ALTER TABLE <table_name> MODIFY <table_camp>;

ALTER TABLE angajat MODIFY stagiu_ani CHAR(25);

3. Descrieți ordonarea înregistrărilor într-o BD. Dați exemple.

Instrucțiunea **ORDER BY** ordonează înregistrările într-o interogare. Locul acestei instrucțiuni **ORDER BY** este mereu la finele interogării. By default, la utilizarea acestei comenzi fără a specifica ordinea, înregistrările vor fi ordonate crescător. Pentru a specifica ordinea dorită utilizăm sintaxele următoare:

Sintaxa ASC - ascending (crescător).

Sintaxa DESC - descending (descrescător)

SELECT * FROM angajat WHERE orasul='Balti' ORDER BYsalariu ASC

4. Definiți și descrieți cu un exemplu instrucțiunea LEFT OUTER JOIN.

Comanda **JOIN** este asociere pe baza căreia sunt selectate selectate înregistrările, care satisfac strict condiția sau condițiile specificate prin asociere de către user. Respectiv **LEFT OUTER JOIN** înseamnă asocierea înregistrărilor din exterior spre stânga.

SELECT flight.id, plane.data_livr, departure.city, arrival.city FROM flightschedule LEFT OUTER JOIN booked_tickets ON flight.id_cump=plane.id WHERE nume LIKE'V%'

5. Creați tabelul “shop” ce conține următoarele câmpuri: id_shop (integer not null primary key), denumire (varchar2(30) null, uppercase), orasul (varchar2(20) default ‘Chisinau’), adresa (varchar2(40) null), data_deschiderii (date null).

create table shop(id_apartment integer not null primary key, denumire varchar2(30) null, orasul varchar2(20) default ‘Chisinau’, adresa varchar2(40) null, data_deschiderii date null);

6. Afisați toți cumpărătorii născuți între luna august și luna noiembrie.
SELECT * FROM cumparator WHERE EXTRACT(MONTH FROM data_nasterii) BETWEEN 08 AND 11;

7. Calculați câte apartamente din regiunea, care se începe cu litera “B”, cu suprafata maximală aveți.
SELECT count(id_apartment) FROM apartament where regiunea like 'B%' and suptafata=(SELECT max(suprafata) from apartament);

8. Selectați toate apartamentele cu nr_de_camere egal cu 2 sau 3. Ordonăți această selecție crescător după câmpul cod.
SELECT * FROM apartament where nr_de_cam='2' OR nr_de_cam='3' order by id_apartment ASC;

9. Afisați numărul de cumpărători din fiecare oraș, care au procurat apartamente la pret maximal. (folosiți INNER JOIN).
SELECT MAX(pretul), count(*) FROM makler INNER JOIN apartament1 ON makler.id_apart=apartament1.id INNER JOIN cumparator ON makler.id_cump=cumparator.id;

10. Creați o procedură care va adauga la preț 1000 de lei, la apartamentele care sunt de tipul “143”.
***CREATE OR REPLACE PROCEDURE pretul_adaug
(p_id IN apartament1.tipul_apart%Type)
IS
BEGIN
UPDATE apartament1 SET pretul= pretul+1000
Where tipul_apart='143';
END pretul_adaug;***

11. Vorbiți despre baza de date personală: tabele, relații, interogări, etc.

Tabele:

- operatori
- client
- contracte

*fiecare cu 10 înregistrări

Am utilizat comenzi de sortare precum:

```
SELECT * FROM client WHERE EXTRACT(MONTH FROM data_nasterii)  
BETWEEN 04 AND 09;
```

```
SELECT * FROM contract where nr_de_servicii='1' OR nr_de_servicii='5'  
order by id_contract ASC;
```

De asemenea am aplicat comenzi de stergerere precum si a tabelelor **DROP TABLE** pentru a sterge tabelul contractinitial, dar si a inregistrarilor din table cu ajutorul comenzii **DELETE FROM**.

In cadrul comenzilor de selectie am utilizat operatori logici precum **and, or, not**

Am modificat date din cadrul acestora utilizand instructiunea **ALTER TABLE** impreuna cu comenzile **ADD, MODIFY, RENAME COLUMN, DROP COLUMN**

De asemenea am utilizat instructiuni de ordonare/ grupare (**ORDER BY** cu sintaxele **ASC – crescator & DESC -descrescator; GROUP BY + HAVING**)

La utilizarea comenzilor din familia de instructiuni **JOIN (INNER/OUTER JOIN)** am selectat/sortat datele din mai multe tabele prin asociere

La moment am efectuat doar o procedura, una de modificare a pretului contractului

```
CREATE OR REPLACE PROCEDURE pretul_mod  
(p_id IN contract.tip_contract%Type )  
IS  
BEGIN  
UPDATE contract SET pretul= pretul+239  
Where tipul_contract='3';  
END pretul_mod;
```