# Subselecturile:

1. Afisati numele (last\_name) si ziua de angajare ai angajatilor ce sunt colegi de departament cu Zlotkey. Zlotkey va fi exculs din lista.
2. Scrieti un query care sa afiseje employee\_id si last\_name pentru toti angajatii ce lucreaza in acelasi departament cu angajatii al caror nume contine litera u.
3. Afisati last\_name, job\_title si salary pentru angajatii cu joburi ce incep cu P si au salariul diferit de valorile 2500, 3500 si 7000. Ordonati lista dupa job ascendent si nume descendent.
4. Afisati last\_name, job\_id, department\_id, salary din tabela employees pentru toti angajatii care castiga mai mult decat orice salariat din departamentul 30.

# Alte obiecte din baza de date:

View-uri

1. Creati un view cu numele V\_SALARIATI, care sa aiba urmatoarele informatii:

Employee\_id, first\_name, last\_name, job\_id,job\_title, department\_id, department\_name, salary, commission\_pct.

Viewul trebuie sa contina toti salariatii, chiar daca printere ei exista unii care nu apartin nici unui department.

2 . Creati un view cu numele V\_SALARIATI\_FINANCE care sa contina toate informatiile din tabela employees si care sunt in departamentul 100.

View-ul sa fie creat cu clauza de WITH CHECK OPTION.

Inserati in view o inregistrare cu orice date doriti, dar care sa fie in departamentul 200.

Ce se intampla?

Secvente:

1.Creati o secventa cu numele emp\_sequence care sa inceapa cu valoarea 10 , sa se incrementeze cu 10,valoarea maxima pe care o va genera secventa sa fie 1000, iar valoarea pe care o va pastra Oracle in cache sa fie de 100.

2. Inserati o inregistrare in tabela emp\_test folosind secventa emp\_sequence pentru completarea campului employee\_id.

3. Afisati printr-un select valoarea curenta si valoarea urmatoare a secventei.

# Manipularea datelor (DML)

1. Introduceti in tabela employees o inregistrare ce contine un nou employee\_id unic (fomat din 1000 + nr lista prezenta) , numele si prenumele vostru, data de anagajare data sistemului, email TNI ,jobul Programmer, salariul 10000 si comisionul 0.2.
2. Faceti un update alocandu-va la departamentul IT si managerul acestuia.
3. Mariti-va salariul aplicand comisionul, si aflati media salariilor anuale in departamentul dvs inainte si dupa marire.

# Controlul tranzactiilor

1. Creati o tabela cu structura Employees\_test(EMPLOYEE\_ID NUMBER,

FIRST\_NAME VARCHAR2(20 BYTE),

LAST\_NAME VARCHAR2(25 BYTE) )

1. Inserati un rand cu numele vostru si employee\_id = 1
2. Creati un savepoint savepoint a.
3. Mai inserati un rand cu ce nume doriti si employee\_id = 2
4. Creati un savepoint savepoint b.
5. Mai inserati un rand cu ce nume doriti si employee\_id = 3
6. Creati un savepoint savepoint c.
7. Rulati comanda rollback to savepoint b;
8. Ce observati?
9. Rulati comanda commit;
10. Apoi faceti select din tabela. Cate inregistrari vor exista in tabela?

# Operatii cu seturi de date

1. Folosind MINUS rulati un select care sa aduca toti din tabela employees , mai putin pe cei al caror firs\_name incepe cu litera S.
2. Creati o tabela employees1 cu aceeasi structura ca si tabela employees dar doar cu cei din departamentul 30.
3. Rulati un select care sa aduca first\_name, salary, hire\_date,department\_id din employees, dar si din employees1 care au inregistrari identice.
4. Rulati un select care sa aduca first\_name, salary, hire\_date,department\_id din ambele tabele, chiar daca inregistrarile sunt duplicate.
5. Rulati un select care sa aduca first\_name, salary, hire\_date,department\_id din ambele tabele, dar fara a duplica inregistrari.