

## 1. rész – DDL és DML (csak áttekintés)

- a) **Hozzon létre egy táblát, amiben van változó hosszúságú karakterlánc (ami ne vehessen fel *null* értéket, valamint legyen *primary key*), dátum és szám attribútum!**

```
create table tabla (  
    m1 varchar2(10) primary key not null,  
    m2 date,  
    m3 integer);
```

- b) **Módosítsa az előző táblát: törölje az egyik előzőleg létrehozott attribútumot, majd vegyen fel egy új attribútumot!**

```
alter table tabla drop column m2;
```

```
alter table tabla add m2 date;
```

- c) **Nevezze át az előző táblát!**

```
alter table tabla rename to tabla1;
```

- d) **Szúrjon be néhány rekordot a táblába!**

```
insert into tabla1 values ('ertek', 20, to_date('20110911','yyyymmdd'));
```

- e) **Egészítse ki a táblát egy *CHECK* típusú megszorítással! Próbáljon beszúrni olyan sorokat, amik megsértenék ezt a megszorítást!**

```
alter table tabla1 add check (m3 > 10)
```

- f) **Töröljön sorokat a táblában egy adott attribútum értéke alapján!**

```
delete from tabla1 where feltétel;
```

- g) **Módosítsa valamely attribútum értékét valamilyen feltétel teljesülése esetén!**

```
update tabla1 set m1='mas' where feltétel;
```

- h) **Hozzon létre egy indexet a tábla egy attribútumára, majd szüntesse meg!**

```
create index idx on tabla1(m2);
```

```
drop index idx;
```

- i) **Szüntesse meg az előző táblát, de előtte másolja le (csináljon egy ugyanolyat)!**

```
CREATE TABLE tabla4 AS SELECT * FROM tabla1;
```

```
drop table tabla1;
```

## 2. rész – halmazműveletek megvalósítása SELECT-tel

### KEDVENC táblával:

#### a) Képezze az összes lehetséges (nev, gyumolcs) párt!

```
select distinct nev, gyumolcs
from (select distinct nev from kedvenc) nevek,
     (select distinct gyumolcs from kedvenc) gym;
```

#### b) Melyek azok a gyumolcsök, amelyeket Péter nem kedvel?

```
select distinct gyumolcs from kedvenc
minus
select distinct gyumolcs from kedvenc where nev = 'Péter';
```

#### c) Melyek azok a gyumolcsök, amelyeket Róberten kívül más is kedvel?

```
select distinct gyumolcs from kedvenc where nev='Róbert'
intersect
select distinct gyumolcs from kedvenc where nev <>'Róbert';
```

#### d) Kik kedvencnek legalább kétféle gyumolcsöt?

```
select nev, count(gyumolcs) from kedvenc
group by nev
having count(gyumolcs) >= 2;
```

#### e) Kik kedvelik Ádám összes kedvenc gyumolcsét?

```
SELECT DISTINCT sz1.nev
FROM kedvenc sz1
WHERE sz1.nev <>'Ádám' AND NOT EXISTS
( SELECT *
  FROM kedvenc sz2
  WHERE sz2.nev='Ádám' AND sz2.gyumolcs NOT IN
    ( SELECT gyumolcs
      FROM kedvenc
      WHERE nev = sz1.nev)
)
```

```
select nev, gyumolcs
from kedvenc
where nev<>'Ádám' and gyumolcs
in (
  select gyumolcs
  from kedvenc
  where nev='Ádám'
);
```

#### f)\* Kik azok, akiknek minden kedvenc gyumolcsét Ádám is kedveli?

```
SELECT DISTINCT sz1.nev
FROM kedvenc sz1
WHERE sz1.nev <>'Ádám' AND NOT EXISTS
( SELECT *
  FROM kedvenc sz2
  WHERE sz2.nev=sz1.nev AND sz2.gyumolcs NOT IN
    ( SELECT gyumolcs FROM kedvenc WHERE nev='Ádám'))
```

### 3. rész – Az ORACLE demo adatbázisa

Állítsa be a dátumok formátumát magyarra az `alter session set nls_date_format = 'YYYY.MM.DD';` paranccsal!

- a) Listázza ki fizetés szerint csökkenő sorrendben az összes olyan dolgozó nevét, belépési dátumát és fizetését, aki *Alexander Khoo*-val azonos osztályon dolgozik!

```
select first_name, last_name, hire_date, salary
from employees
where department_id = (
    select department_id
    from employees
    where first_name='Alexander' and last_name='Khoo'
)
order by salary desc;
```

- b) Készítse el azoknak a dolgozóknak a listáját, akiknek a vezetékneve *K*-val kezdődik! A listában a dolgozó vezetékneve és a beosztása szerepeljen, továbbá a jutalék százzal beszorzott nagysága, illetve a „nem jár” szöveg, ha a dolgozónak nincs beállítva jutalék!

```
select last_name AS utónév,
       job_title AS beosztás,
       case
           when commission_pct is null then 'Nem jár'
           else to_char(commission_pct * 100)
       end AS jutalék
from employees inner join jobs on jobs.job_id = employees.job_id
where last_name like 'K%';
```

*Az NVL függvény nem jó, mert csak a commission\_pct-vel azonos típust írhat ki!*

- c) Listázza ki csökkenő fizetés szerint az összes alkalmazott vezetéknevét és a fizetésének nagyságát egy mezőben úgy, hogy a vezetéknevek 10 karakterhelyet foglaljanak, utána kettőspont álljon, majd annyi # karakter, ahány 1000 USD-t keres a dolgozó!

```
select RPAD(last_name, 10) ||
       RPAD(':', ROUND((salary/1000),0)+1, '#') AS fizetésnagyságrend
from employees
order by salary desc;
```

- d) Készítse el azoknak a részlegeknek a listáját, akik 2000 dollár feletti átlagos jövedelmet fizetnek a dolgozóiknak! A listában a részleg neve és az egészre kerekített átlagos jövedelem szerepeljen, az utóbbi szerinti sorrendben!

```
select department_name AS részleg,
       ROUND(AVG(salary)) AS átlagjövedelem
from employees inner join departments using(department_id)
group by department_name
having ROUND(AVG(salary)) > 2000
order by átlagjövedelem;
```

*Ha a két táblában megegyezik a kapcsolómezők neve, akkor a USING kulcsszóval elvégezhetjük az összekapcsolást.*

*A having záradékban nem használhatjuk az álnevet!*

- e) Módosítsa az előző lekérdezést úgy, hogy a jutalékkal megnövelt fizetésekből számítsa az átlagot! A NULL értékeket az NVL függvény segítségével alakítsa 0 értékűvé!**

```
select department_name AS részleg,  
       ROUND(AVG(salary * (1 + nvl(commission_pct, 0)))) AS átlagjövedelem  
from employees inner join departments using(department_id)  
group by department_name  
having ROUND(AVG(salary)) > 2000  
order by átlagjövedelem;
```

- f) Listázza ki a locations táblában szereplő helyszínek országnevét, címét és városnevét! Azok az országok is jelenjenek meg, amelyekhez egyetlen helyszín sem tartozik! Rendezze a listát az ország, azon belül a város neve szerint!**

```
select country_name, street_address, city  
from countries left join locations using(country_id)  
order by 1, 3;
```

*A USING féloldalas join-oknál is használható. A rendezés szempontjait sorszámokkal is megadhatjuk (a mezők select-beli sorrendjének megfelelően).*

- g) Listázza ki az 50-nél kisebb részlegazonosítójú részlegen dolgozók nevét, részlegazonosítóját és főnökének nevét! A mezőknek adjon magyar megnevezést, majd rendezze a listát a főnök neve, azon belül a dolgozó neve szerint!**

```
select beosztott.last_name AS "beosztott neve",  
       beosztott.department_id AS "részlegazonosító",  
       főnök.last_name AS "főnök neve"  
from employees főnök, employees beosztott  
where főnök.employee_id = beosztott.manager_id  
and beosztott.department_id < 50  
order by főnök.last_name, beosztott.last_name;
```