

## 12. Gyakorlat

### Összetett és belső lekérdezések, virtuális táblák

#### 1. Összetett lekérdezések:

Összetett lekérdezésről beszélünk, ha több tábla alapján történik a lekérdezés. Ilyenkor a táblák Descartes-szorzatából szűrjük ki azokat a sorokat, amelyek számunkra fontosak. Az összetett lekérdezéseknél általában természetes összekapcsolást használunk.

##### Példák:

Válogassuk ki azokat a hallgatókat, akik szegediek!

HALLGATO( eha, nev, anyja\_neve, irsz, lakcim )  
HELYSEG( irsz, varos )

```
SELECT * FROM hallgato, helyseg
      WHERE hallgato.irsz = helyseg.irsz AND varos='SZEGED';
```

Listázzuk ki azoknak a budapesti ügyfelek nevét és születési évét, akik már 3 éve ügyfelek!

UGYFEL(ukod, nev, lakcim, szulev, irsz)  
HELYSEG( irsz, varos )  
BELEPETT(ukod, belep, kilep)

```
SELECT nev, szulev FROM ugyfel, helyseg, belepett
      WHERE ugyfel.ukod = belepett.ukod AND
            ugyfel.irsz = helyseg.irsz AND
            varos='Budapest' AND belep < 2008-3 AND kilep=0;
```

#### 2. Belső lekérdezések (alkérdések):

Alkérdezéseket általában akkor alkalmazunk, ha az adattáblákból valamilyen „globális” információt szeretnénk kinyerni.

##### Példák:

Listázzuk ki azokat a dolgozókat, akiknek az átlagnál kevesebb a fizetése.

DOLGOZO( id, nev, szulev, fizetes )

```
SELECT nev, fizetes FROM dolgozo
      WHERE fizetes < (SELECT AVG(fizetes) FROM dolgozo);
```

Listázzuk ki a budapesti ügyfelek nevét és születési évét! (Egyszerűbben is lehet.)

UGYFEL(ukod, nev, lakcim, szulev, irsz)  
HELYSEG( irsz, varos )  
BELEPETT(ukod, belep, kilep)

```
SELECT nev, szulev FROM ugyfel AS u1
      WHERE u1.ukod IN (SELECT ukod FROM ugyfel AS u2, helyseg
                       WHERE u2.irsz = helyseg.irsz AND
                             varos = 'Budapest');
```

Listázzuk ki azokat a dolgozókat, akiknek az osztályon belül legmagasabb a fizetése!

DOLGOZO(nev, cim, osztalykod, fizetes)

```
SELECT osztalykod, nev, fizetes FROM dolgozo AS d1
WHERE fizetes = ( SELECT MAX(fizetes) FROM dolgozo AS d2
WHERE d1.osztalykod=d2.osztalykod );
```

### 3. Virtuális táblák (nézettáblák)

A virtuális táblák nem valódi táblák az adatbázisban, csak származtatott adatokat tartalmaznak.

A virtuális tábla nem tárol adatokat. Ez egy transzformációs formula, amelyet úgy képzelhetünk el, mintha a tárolt táblák adatait látnánk egy speciális „szűrőn” keresztül.

Nézettáblák alkalmazási lehetőségei:

- Származtatott táblák létrehozása, amelyek a törzsadatok módosításakor automatikusan módosulnak (pl. összegzőtáblák)
- Bizonyos adatok elrejtése a felhasználók előtt (adatbiztonság vagy egyszerűsítés céljából).

Nézettáblák létrehozása:

```
CREATE VIEW táblanév [(oszloplista)] AS alkérdés;
```

Az alkérdés utasítás eredménytáblája alkotja a nézettáblát. Az oszloplista megadásával új nevet adhatunk az oszlopoknak.

A nézettáblákon is végrehajthatók azok a műveletek, amelyek a tárolt táblákon. Ha a nézettáblában változik egy adatsor, akkor az változni fog a megfelelő tárolt táblában is (ha egyértelműen módosítható).

Példa:

RAKTÁR( cikkszám, név, egységár, mennyiség )

```
CREATE VIEW Keszlet(aru, ertek) AS
SELECT nev, egysegar*mennyiseg FROM raktar;
```

Nem lehet módosítani a nézettáblát, ha a definíciója

- DISTINCT opciót,
  - FROM után egynél több táblanevet
  - GROUP BY alparancsot
- tartalmaz.

#### **Feladatok:**

1. Számítsuk ki a 35 és 45 év közötti dolgozók átlagfizetését!
2. Ki az a legidősebb dolgozó, akinek a fizetése az átlagfizetés alatt van?