

# Lekérdezések SQL-ben (Access)

Készítette: Kuli Ferenc  
[oktatas.tanarurkerem.com](http://oktatas.tanarurkerem.com)

# SQL az egyszerű lekérdezésekben

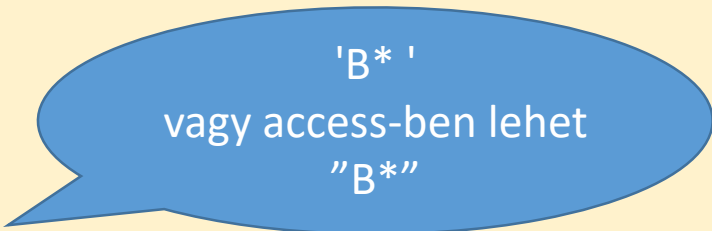
- SQL (Structured Query Language) strukturált lekérdezőnyelv
- Az egyszerű lekérdezés általános alakja:  
SELECT <mezőnévlista>  
FROM <táblanév>  
WHERE <feltétel>
- Az **europa** adatbázist használjuk  
Kapcsolódó videó (csak access):  
<https://youtu.be/wQVYM5XURdU>

# Egyszerű lekérdezések (LIKE ? \*)

Jelenítsük meg azon országok nevét és fővárosát, melyek városneve B-vel kezdődik! (*B-sek*)

(A lekérdezést az access-ben megszokott módon hozzuk létre és átváltunk SQL nézetre.)

```
SELECT fovaros  
FROM Europa  
WHERE fovaros LIKE 'B*';
```



'B\* '  
vagy access-ben lehet  
"B\*"

Jelenítsük meg azon országok összes adatát, melyek városneve B-vel kezdődik!

```
SELECT *  
FROM Europa  
WHERE fovaros LIKE 'B*';
```

# Egyszerű lekérdezések (rendezés)

Jelenítsük meg azon országok nevét és fővárosát, városnév szerint abc sorrendben, melyek fővárosának második betűje „u”! (*Második\_u*)

```
SELECT orszag, fovaros  
FROM Europa  
WHERE fovaros LIKE '?u*'  
ORDER BY fovaros ASC;
```

Növekvő sorrend:

```
ORDER BY fovaros ASC
```

vagy

```
ORDER BY fovaros
```

Csökkenő sorrend:

```
ORDER BY fovaros DESC
```

# Egyszerű lekérdezések (IN)

Jelenítsük meg azokat az országokat, melynek fővárosa Budapest vagy London! (*Budapest\_London*)

```
SELECT orszag, fovaros
```

```
FROM Europa
```

```
WHERE fovaros IN ('Budapest','London');
```

Másik megoldás:

```
SELECT orszag, fovaros
```

```
FROM Europa
```

```
WHERE fovaros LIKE 'Budapest' OR fovaros LIKE 'London';
```

Harmadik megoldás (csak a WHERE sor):

```
WHERE fovaros='Budapest' OR fovaros='London';
```

# Egyszerű lekérdezések (NOT IN)

Jelenítsük meg azokat az országokat, melynek fővárosa nem Budapest vagy London! (*Nem\_Budapest\_London*)

```
SELECT orszag, fovaros
```

```
FROM Europa
```

```
WHERE fovaros NOT IN ('Budapest','London');
```

Másik megoldás:

```
SELECT orszag, fovaros
```

```
FROM Europa
```

```
WHERE fovaros NOT LIKE 'Budapest' AND fovaros NOT LIKE 'London';
```

Harmadik megoldás (csak a WHERE sor):

```
WHERE fovaros<>'Budapest' AND fovaros<>'London';
```

# Egyszerű lekérdezések (BETWEEN)

Jelenítsük meg azokat a közép-európai országokat, melyek lakossága 10 és 20 millió közötti! (*10\_20\_közép\_Eu*)

```
SELECT ország, lakosság
```

```
FROM Europa
```

```
WHERE lakosság BETWEEN 10 AND 20 AND egtaj='közép';
```

Másik megoldás:

```
SELECT ország, lakosság
```

```
FROM Europa
```

```
WHERE lakosság >=10 and lakosság<= 20 and egtaj='közép';
```

A BETWEEN, az OR, az END írható kisbetűvel is

# Egyszerű lekérdezések (TOP, AS)

Jelenítsük meg a legnagyobb területű ország nevét!  
(*Legnagyobb\_ország*)

```
SELECT TOP 1 orszag AS Ország  
FROM Europa  
ORDER BY terület DESC;
```

AS adja meg,  
hogy milyen  
néven jelenjen  
meg az adott  
mező  
(másodlagos név)

TOP 1 csak az elsőt  
jeleníti meg.

Pl.: TOP 5 az első ötöt



# Egyszerű lekérdezések (számított mező)

Jelenítsük meg az országok nevét és népsűrűségét!  
(*Népsűrűség*)

```
SELECT ország, lakosság/terulet*1000000 AS népsűrűség  
FROM Europa;
```

A lekérdezésben számítja ki:  
lakosság/terulet\*1000000  
és népsűrűség névvel jelenik meg.

# Egyszerű lekérdezések (paraméteres)

Jelenítsük meg egy adott égtájhoz tartozó országok nevét, úgy hogy az égtájat a lekérdezés futásakor kell megadni (paraméter)! (*Paraméteres*)

```
SELECT ország  
FROM Europa  
WHERE Egtaj=Melyik_égtáj;
```

Melyik\_égtáj mezőnéven bekéri az aktuális értéket

Access-ben lehet így is:

```
SELECT ország  
FROM Europa  
WHERE Egtaj=[Melyik égtáj?];
```

# Összesítés a lekérdezésben

- Aggregáló (összegyűjtő) függvények segítségével a teljes táblára, vagy csoportonként végezhetünk számításokat a lekérdezésben.

COUNT( ...) - az értéket tartalmazó sorok száma

SUM(...) - az adott oszlop sorainak összege

AVG(...) - az adott oszlop sorainak átlaga

MIN(...) - az adott oszlop sorainak minimuma

MAX(...) - az adott oszlop sorainak maximuma

- SELECT parancsban ALL és DISTINCT opció is lehet
  - ALL mindent megjelenít (elhagyható)
  - DISTINCT minden azonosból csak egyet jelenít meg

Kapcsolódó videó (csak access):

<https://youtu.be/WDW1ahWqnjA>

# Összesítés (COUNT, SUM, AVG)

Adjuk meg a lekérdezésben az országok számát, Európa összterületét, a lakosság átlagát! (*Eu\_összesítés*)

```
SELECT Count(ország) AS országok_száma, Sum(terulet) AS  
összterület, Avg(lakosság) AS lakosság_átlaga  
FROM Europa;
```

Adjuk meg Közép-Európa országainak számát!  
(*Közép\_EU\_országyszáma*)

```
SELECT Count(ország) AS országok_száma  
FROM Europa  
WHERE egtaj='közép';
```

Adjuk meg az égtájakat! (*Égtájak*)

```
SELECT DISTINCT Egtaj  
FROM Europa;
```

# Csoportosítás a lekérdezésben (GROUP BY, HAVING)

Adjuk meg égtájanként az országok számát, összterületét!  
(*Csoportosítás*)

```
SELECT egtaj AS Égtáj, Count(ország) AS Országok_száma,  
Sum(terulet) AS összterület
```

```
FROM Europa
```

```
GROUP BY egtaj;
```

GROUP BY a megadott mező  
szerint csoportosítva adja meg  
az adatokat

Csoportosítással adjuk meg Közép-Európa országainak  
számát! (*Közép\_EU\_országyszáma2*)

```
SELECT count(ország) AS országok_száma
```

```
FROM Europa
```

```
GROUP BY egtaj
```

```
HAVING egtaj='közép';
```

A csoportosítás utáni  
feltételként a HAVING  
záradékot kell használni!  
(nem WHERE)

# Csoportosítás a lekérdezésben (számított mező)

Adjuk meg égtájanként a népsűrűséget, népsűrűség szerinti csökkenő sorrendben! (*népsűrűség\_égtájanként*)

```
SELECT egtaj AS Égtáj, Sum(lakosság)*1000000/Sum(terulet) AS  
Népsűrűség  
FROM Europa  
GROUP BY Egtaj  
ORDER BY Sum(lakosság)*1000000/Sum(terulet) DESC;
```

A népsűrűség: összes lakos/összes terület!

A rendezésnél is újra le kell írni a számított mezőt, ami szerint kell majd rendeznie!

# Allekérdezés

Adjuk meg azokat az országok nevét, amelyek ugyanabban az égtájban vannak mint Magyarország! (Magyarország neve ne jelenjen meg!) (*Mint\_Mo*)

```
SELECT ország  
FROM Europa  
WHERE egtaj=(SELECT egtaj FROM Europa WHERE  
ország='Magyarország') AND ország<>'Magyarország';
```

Egy allekérdezéssel meg kell adni Magyarország égtáját:  
**SELECT egtaj FROM Europa WHERE ország='Magyarország'**  
majd ez kerül be a feltételbe!

**ország<>'Magyarország'**  
biztosítja hogy Magyarország neve  
ne jelenjen meg

# Allekérdezés

Másik megoldás: (*Mint\_Mo2*)

```
SELECT ország  
FROM Europa  
WHERE egtaj IN (SELECT egtaj FROM Europa WHERE  
ország='Magyarország') AND ország<>'Magyarország';
```

Adjuk meg azoknak az országoknak nevét, amelyek nem azon az égtájon találhatóak mint Belgium! (*Nem\_Belgium*)

```
SELECT ország  
FROM Europa  
WHERE egtaj NOT IN (SELECT egtaj FROM Europa WHERE  
ország='Belgium');
```

Másik megoldás (csak a WHERE sor):

```
WHERE egtaj<>(SELECT egtaj FROM Europa WHERE  
ország='Belgium');
```



# Allekérdezés

Adjuk meg azoknak az országoknak nevét, amelyek területe Európa átlagterületénél kisebb!

*(Terület\_EU\_átlag\_alatt)*

```
SELECT ország  
FROM Europa  
WHERE terület<(SELECT Avg(terület) FROM Europa);
```

terület<Avg(terület) nem jó!  
Allekérdezés szükséges!

Hány ország sűrűbben lakott mint Magyarország?

*(Sűrűbb\_mint\_Mo)*

```
SELECT Count(ország) AS országszám  
FROM Europa  
WHERE lakosság*1000000/terület>(SELECT  
lakosság*1000000/terület FROM Europa WHERE  
ország='Magyarország');
```

# Több tábla összekapcsolása

A *teljesitmeny* adatbázist használjuk.

A feladat leírása és forrása:

*oktatas.tanarurkerem.com – Adatbáziskezelő feladatok – Adatbáziskezelő feladatok Access -- 22. Területi képviselők c. feladat*

Az **Ugynokok** tábla mezőinek jelentése a következő:

<i>UID</i>	A területi képviselő azonosítója (szám vagy számláló, elsődleges kulcs)
<i>Ugynok</i>	A területi képviselő neve (szöveg)

A **Forgalom** tábla mezőinek jelentése a következő:

<i>FID</i>	A bejegyzés azonosítója (szám vagy számláló, elsődleges kulcs)
<i>UID</i>	A területi képviselő azonosítója, akihez a bejegyzés tartozik
<i>KB</i>	A bejegyzés jellege, megadja, hogy a bejegyzés kifizetésről (K) vagy bevételről (B) szól (szöveg)
<i>Osszeg</i>	Az a pénzösszeg, amiről a bejegyzés szól (szám vagy pénznem)
<i>Datum</i>	A bejegyzés keletkezésének dátuma (dátum)

# Több tábla összekapcsolása (WHERE)

Amennyiben a lekérdezésben több tábla szükséges, azok összekapcsolását a WHERE utasításban tesszük meg. Ha olyan mezőre hivatkozunk, amely más táblában is van ugyanazon névvel, akkor a táblák nevét is le kell írni pl. `ugynok.UID` vagy `forgalom.UID`

Adjuk meg, Bíró Béla bevételeinek dátumait és az összegeket! (A)

```
SELECT Forgalom.Datum, Forgalom.Osszeg  
FROM Forgalom, Ugynokok  
WHERE Ugynokok.UID=Forgalom.UID and  
Ugynokok.Ugynok='Bíró Béla' and Forgalom.KB='B';
```

A használt táblák felsorolva

Ugynokok.UID=Forgalom.UID  
kapcsolja össze a két táblát

# Több tábla összekapcsolása (csoportosítás)

Adjuk meg képviselőnként az összbevételt! (*B*)

```
SELECT Ugynokok.Ugynok, Sum(Forgalom.Osszeg)  
AS bevételek
```

```
FROM Forgalom, Ugynokok
```

A használt táblák felsorolva

```
WHERE Ugynokok.UID=Forgalom.UID
```

```
GROUP BY Ugynokok.Ugynok;
```

WHERE Ugynokok.UID=Forgalom.UID  
kapcsolja össze a két táblát

# Több tábla összekapcsolása (allekérdezés)

Kinek volt a képviselők között bevétele azokon a napokon, amikor Bíró Bélának is (Bíró Béla neve ne jelenjen meg!)? (C)

```
SELECT DISTINCT Ugynokok.Ugynok  
FROM Forgalom, Ugynokok
```

DISTINCT csak a  
különbözőket jeleníti meg

```
WHERE Ugynokok.UID=Forgalom.UID and Forgalom.KB='B'  
and Ugynokok.Ugynok<>'Bíró Béla' and Forgalom.Datum IN  
(SELECT Forgalom.Datum FROM Forgalom, Ugynokok WHERE  
Ugynokok.UID=Forgalom.UID and Ugynokok.Ugynok='Bíró  
Béla' and Forgalom.KB='B');
```

Egy allekérdezéssel meg kell adni Bíró Béla bevételt hozó napjainak dátumait, ez egyezik meg a többiek dátumával:

```
IN (SELECT Forgalom.Datum FROM Forgalom, Ugynokok WHERE Ugynokok.UID=Forgalom.UID  
and Ugynokok.Ugynok='Bíró Béla' and Forgalom.KB='B')
```

Az IN-t használjuk mert több ilyen nap is van

# Több tábla összekapcsolása (JOIN)

A táblák összekapcsolása a JOIN paranccsal is lehetséges a FROM részben pl.:

```
FROM Forgalom INNER JOIN Ugyenokok ON Forgalom.UID =  
Ugyenokok.UID
```

Adjuk meg, Bíró Béla bevételeinek dátumait és az összegeket! (*A2*)

```
SELECT Forgalom.Datum, Forgalom.Osszeg  
FROM Forgalom INNER JOIN Ugyenokok ON Forgalom.UID =  
Ugyenokok.UID  
WHERE Ugyenokok.Ugyenok='Bíró Béla' and Forgalom.KB='B';
```

# INNER JOIN, LEFT JOIN, RIGHT JOIN

- INNER JOIN a két táblában meglévőket létezőket kapcsolja össze.
- LEFT JOIN Ez annyiban különbözik az INNER JOIN-től, hogy amikor összekapcsolunk két táblát és ha nincsen megfelelő érték az első helyen lévő táblába a második helyen lévő táblához, akkor kiegészíti azokat NULL értékekkel. Ez akkor kellhet, amikor minden sornak szerepelni kell az eredményben, még akkor is, ha nem tartozik hozzá érték.
- RIGHT JOIN Ez annyiban különbözik az LEFT JOIN-től, hogy a másik irányba egészíti ki a táblát.