

# Adatbázis kezelés I.

## SQL bevezető

Rostagni Csaba

2024. szeptember 2.

# Ezen az órán... I

- 1 Bevezető
- 2 Egyszerű lekérdezések (SELECT)
- 3 Műveleti jelek
- 4 Feltételes lekérdezések (WHERE)
- 5 Rendezés (ORDER BY)

# Tartalom I

## 1 Bevezető



# Speciális karakterek

A diákon különböző speciális jelek találhatók meg.

## aposztróf

- A MySQL aposztóffal jelöli a szöveget.

pl.: 'szoveg'

-  + 

## backtick

- Így jelöli adatbázisokat, táblákat és a mezőneveket.

pl.: `tablanev`

-  + 

# Megjegyzés (egy soros)

- A # karaktertől a sor végéig
- A -- karaktersorozattól a sor végéig
  - Utána egy szóköz kell!  
pl.: --*Ez egy megjegyzés*
  - Kissé eltér a szabványtól, de **ezt használjuk!**

Linkek:

- MySQL dokumentáció: Megjegyzések
- MySQL dokumentáció: A szabványtól eltérő megjegyzés

# Megjegyzés (több soros)

- Nyitó karaktersorozat: `/*`
- Záró karaktersorozat: `*/`
- Szöveg közben használható  
pl.: `/* Ez szövegközi vagy több soros megjegyzes */`

Linkek:

- MySQL dokumentáció: Megjegyzések
- MySQL dokumentáció: A szabványtól eltérő megjegyzés

# Különböző típusú értékek jelölése

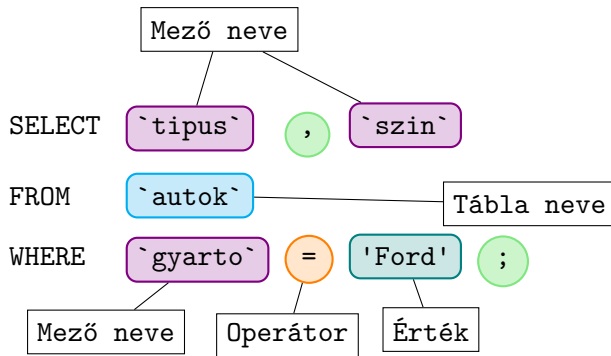
- A szöveges értékeket aposztróf közé kell elhelyezni.  
`'Nagy Lajos'`
- A dátumokat és az időket is aposztróf között fogjuk szabványosan megadni.  
`'2018-12-01'`  
`'15:55:30'`  
`'2018-12-01 12:55:30'`
- Az egész számokat számmal megadhatjuk a szokásos módon.  
`10`
- A valós számokat **tizedes ponttal** kell elválasztani!  
`10.5`

# Tartalom I

## 2 Egyszerű lekérdezések (SELECT)



# Egyszerű SELECT felépítése



# Első lekérdezésünk

Listázzunk ki minden adatot az **autok** táblából!

MySQL

```
SELECT *  
FROM `autok`;
```

rendszám	gyarto	tipus	kategoria	uzemanyag	szin
XXX-111	Opel	Adam	M1	benzin	piros
AAA-555	Honda	Jazz	M1	hibrid	kék
ABC-123	Ford	Focus	M1	diesel	kék
AA-AX-1234	Ford	Fiesta	M1	benzin	sárga

# Mezők kiválasztása

Listázzuk ki az összes autó **rendszámát** és **típusát**!

MySQL

```
SELECT `rendszam`,`tipus`  
FROM `autok`;
```

rendszam	tipus
XXX-111	Adam
AAA-555	Jazz
ABC-123	Focus
AA-AX-1234	Fiesta

# Tartalom I

- 3 Műveleti jelek
  - Összehasonlító operátorok
  - Logikai operátorok

# Tartalom

- 3 Műveleti jelek
  - Összehasonlító operátorok
  - Logikai operátorok

# Összehasonlító operátorok

SQL	Mat.	Megnevezés
<	<	kisebb
>	>	nagyobb
<=	≤	kisebb egyenlő
>=	≥	nagyobb egyenlő
=	=	egyenlő
<>	≠	nem egyenlő

táblázat: összehasonlító operátorok

Megjegyzés:

- A != is a nem egyenlőt jelenti (MySQL, MS SQL, Oracle, ...)
- A ^= is a nem egyenlőt jelenti (Oracle)
- A <> formátum felel meg az ANSI szabványnak, ez az elvárt!

# Tartalom

- 3 Műveleti jelek
  - Összehasonlító operátorok
  - Logikai operátorok

# Logikai operátorok

SQL	Mat.	Megnevezés
NOT	$\neg$	nem
AND	$\wedge$	és
OR	$\vee$	vagy

táblázat: logikai operátorok

Az operátorok precedencia szerint csökkenő sorrendben vannak feltüntetve.

Linkek:

- [MySQL dokumentáció: Operátorok precedenciája](#)



# Tartalom I

## 4 Feltételes lekérdezések (WHERE)

- Egyszerű feltételek megadása
- Tagadás
- ÉS/VAGY alkalmazása

# Tartalom

## 4 Feltételes lekérdezések (WHERE)

- Egyszerű feltételek megadása
- Tagadás
- ÉS/VAGY alkalmazása

# Feltételek megadása (WHERE)

Listázzuk ki a **benzines** autók minden adatát

MySQL

```
SELECT *  
FROM `autok`  
WHERE `uzemanyag` = 'benzin';
```

rendszám	gyarto	tipus	kategoria	uzemanyag	szin
XXX-111	Opel	Adam	M1	benzin	piros
AA-AX-1234	Ford	Fiesta	M1	benzin	sárga

# Megadott mezők és feltételek

Listázzuk ki a **benzines** autók **gyártóját** és **típusát**!

MySQL

```
SELECT `gyarto`,`tipus`  
FROM `autok`  
WHERE `uzemanyag` = 'benzin';
```

gyarto	tipus
Opel	Adam
Ford	Fiesta

# Tartalom

## 4 Feltételes lekérdezések (WHERE)

- Egyszerű feltételek megadása
- Tagadás
- ÉS/VAGY alkalmazása

# Egyszerű tagadás

Listázzuk ki a **nem benzines** autókat.

MySQL

```
SELECT * FROM `autok`  
WHERE `uzemanyag` <> 'benzin';
```

rendszám	gyarto	tipus	kategoria	uzemanyag	szin
AAA-555	Honda	Jazz	M1	hibrid	kék
ABC-123	Ford	Focus	M1	diesel	kék

- Használhattuk volna a `!=` operátort, csak az SQL szabvány nem azt tartalmazza.
- Egyszerű feltételek esetén ilyen egyszerű tagadni.

# Tartalom

## 4 Feltételes lekérdezések (WHERE)

- Egyszerű feltételek megadása
- Tagadás
- ÉS/VAGY alkalmazása

# Logikai operátorok: ÉS (AND)

Listázzuk ki a **benzines Fordok** minden adatát!

MySQL

```
SELECT *  
FROM `autok`  
WHERE `uzemanyag` = 'benzin'  
      AND `gyarto` = 'Ford';
```

rendszám	gyarto	tipus	kategoria	uzemanyag	szin
AA-AX-1234	Ford	Fiesta	M1	benzin	sárga



# Logikai operátorok: VAGY (OR)

Listázzuk ki a **piros** vagy **sárga** színű autók minden adatát!

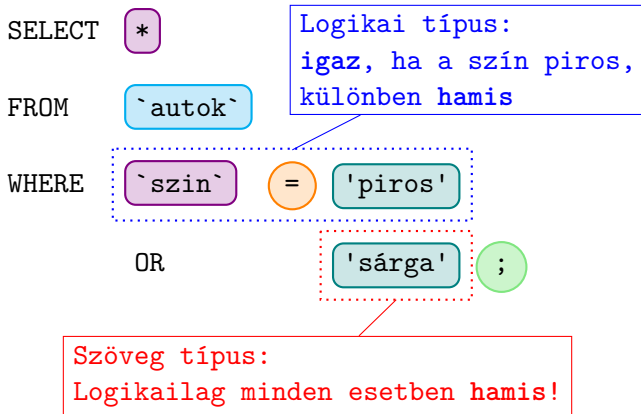
```
SELECT *  
FROM autok  
WHERE szin = 'piros'  
       OR 'sárga';
```

logikai hiba

rendszám	gyarto	tipus	kategoria	uzemanyag	szin
XXX-111	Opel	Adam	M1	benzin	piros

- Hol van a sárga autó?

# Logikai operátorok: VAGY (OR)



- Mivel a második feltétel mindig hamis, így olyan, mintha ott se lenne.

# Logikai operátorok: VAGY (OR)

Listázzuk ki a **piros**, vagy **sárga** színű autók minden adatát!

MySQL

```
SELECT *  
FROM `autok`  
WHERE `szin` = 'piros'  
      OR `szin` = 'sárga';
```

rendszám	gyarto	tipus	kategoria	uzemanyag	szin
XXX-111	Opel	Adam	M1	benzin	piros
AA-AX-1234	Ford	Fiesta	M1	benzin	sárga

# Logikai operátorok: VAGY (OR)

Listázzuk ki a **hibrid** és a **benzines** autók minden adatát!

MySQL

```
SELECT *  
FROM `autok`  
WHERE `uzemanyag` = 'benzin'  
      OR `uzemanyag` = 'hibrid';
```

rendszám	gyarto	tipus	kategoria	uzemanyag	szin
XXX-111	Opel	Adam	M1	benzin	piros
AAA-555	Honda	Jazz	M1	hibrid	kék
AA-AX-1234	Ford	Fiesta	M1	benzin	sárga

- A köznyelvben használt "és" logikailag lehet, hogy "**vagy**"-ot jelent.

# Tartalom I

## 5 Rendezés (ORDER BY)

# ORDER BY

MySQL

```
SELECT * FROM `autok`  
ORDER BY `gyarto`;
```

- Az **ORDER BY** után sorolható fel, hogy melyik mező(k) alapján, növekvő vagy csökkenő sorrendbe rendezve adja vissza adatokat
- **ASC** növekvő (alapértelmezett)
- **DESC** csökkenő

# Növekvő sorrend ASC

Jelenítsük meg az autók minden adatát a gyártók szerinti **növekvő** sorrendben.

MySQL

```
SELECT * FROM `autok`  
ORDER BY `gyarto` ASC;
```

rendszam	gyarto	tipus	kategoria	uzemanyag	szin
ABC-123	Ford	Focus	M1	diesel	kék
AA-AX-1234	Ford	Fiesta	M1	benzin	sárga
AAA-555	Honda	Jazz	M1	hibrid	kék
XXX-111	Opel	Adam	M1	benzin	piros

# Csökkenő sorrend DESC

Jelenítsük meg az autók minden adatát a gyártók szerinti **csökkenő** sorrendben.

MySQL

```
SELECT * FROM `autok`  
ORDER BY `gyarto` DESC;
```

rendszam	gyarto	tipus	kategoria	uzemanyag	szin
XXX-111	Opel	Adam	M1	benzin	piros
AAA-555	Honda	Jazz	M1	hibrid	kék
ABC-123	Ford	Focus	M1	diesel	kék
AA-AX-1234	Ford	Fiesta	M1	benzin	sárga



## Azonosak esetén...

A példán látható, hogy a Ford Focus és Ford Fiesta növekvő és csökkenő rendezés esetén is ugyanabban a sorrendben jelentek meg. Vesszővel felsorolhatunk több rendezési szempontot is.

# Összetett rendezés példa 1

Jelenítse meg az autók adatát **gyártókszerint csökkenő**, míg **típus szerint növekvő** sorrendben!

MySQL

```
SELECT * FROM `autok`  
ORDER BY `gyarto` DESC, `tipus` ASC;
```

rendszam	gyarto	tipus	kategoria	uzemanyag	szin
XXX-111	Opel	Adam	M1	benzin	piros
AAA-555	Honda	Jazz	M1	hibrid	kék
AA-AX-1234	Ford	Fiesta	M1	benzin	sárga
ABC-123	Ford	Focus	M1	diesel	kék

# Rendezés sorszám alapján

MySQL

```
SELECT `nev`, `netto`  
FROM `termekek`  
ORDER BY 2;
```

- A SELECT után felsorolt n-edik mező szerint is rendezhetőek az adatok
- A mezőket 1-től indexeli
- A fenti lekérdezés a második mező, azaz a nettó ár alapján rendez

## Figyelem!

Nem az eredeti tábla, hanem a SELECT után felsorolt mezők sorszáma az, ami számít!

# Rendezés és feltétel

Jelenítsük meg a **kék** színű autók minden adatát a gyártók szerinti **növekvő** sorrendben.

MySQL

```
SELECT * FROM `autok`  
WHERE `szin` = 'kék'  
ORDER BY `gyarto` ASC;
```

rendszám	gyarto	tipus	kategoria	uzemanyag	szin
ABC-123	Ford	Focus	M1	diesel	kék
AAA-555	Honda	Jazz	M1	hibrid	kék