# 4. Rendezés és számított mezők

## A. Előkészületek

Ebben a leckében is a *koltok* adatbázist fogjuk használni.

1. Indítsd el az XAMPP programot, futtasd a web és a MySQL szervereket, majd indítsd el a PHPMyAdmin programot!
2. Kezdj egy új adatbázist *koltok* néven!
3. Importáld a táblát és az adatokat a *koltok.sql* fájlból!

## B. Rendezés

Az eddigi SELECT parancsok olyan sorrendben listázták ki a sorokat, amilyenben bevittük őket a táblába. Ezen természetesen lehet változtatni a **rendezés** megadásával, az ORDER BY kulcsszóval.

Listáztasd ki az összes sort az *evek* táblából a költők születési éve szerint növekvő sorrendben!

SELECT \* FROM evek ORDER BY szuletett;

Babits Mihály és Juhász Gyula ugyanabban az évben születtek. (Mikor?) Ha két vagy több sorban megegyezik a rendezéshez használt mező, akkor ezen sorok között a sorrendet újabb rendezési szempont(ok) megadásával határozhatod meg.

Add meg például, hogy akik azonos évben születettek, azok egymás között a haláluk éve szerinti sorrendben legyenek! Ellenőrizd a sorrendet!

SELECT \* FROM evek ORDER BY szuletett,meghalt;

Nem csak növekvő, hanem **csökkenő** sorrendet is be lehet állítani a DESC kulcsszóval. Állíts be névsor szerint csökkenő sorrendet!

SELECT \* FROM evek ORDER BY nev DESC;

## C. Helyezés

Előfordul, hogy a rendezett listának csak az első néhány sorát szeretnénk kiíratni. (Például egy verseny első három helyezettjét.)

Írasd ki a három legrégebben született költő adatait a LIMIT kulcsszó segítségével!

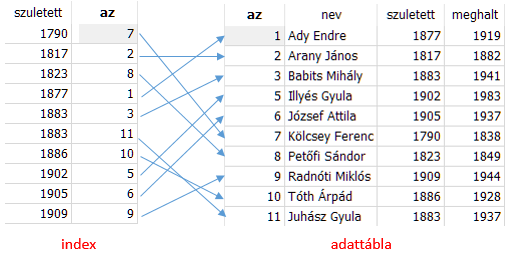
SELECT \* FROM evek ORDER BY szuletett LIMIT 3;

## D. Indexelés

Az **indexek** a könyvek végén lévő indexekhez hasonlóan a keresést és a rendezést gyorsítják meg. Működésük alapja egy lista, amely a rekordok sorrendjét tartalmazza az adott mező vagy mezők szerint.

Ha például gyakran kell a táblát születési idő szerint rendezni, akkor a rendezés felgyorsítható egy szuletett mező szerinti indexeléssel.

Ekkor a program eltárolja a születési idő szerinti sorrendet egy indexbe:



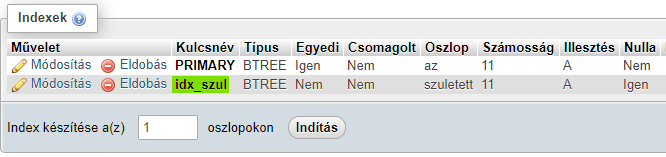
Az adatok listázása az index szerinti sorrendben halad, és az adattáblából veszi az adatokat.

Egy táblához több index is készíthető, így különböző sorrendeket is lehet tárolni.

Készíts egy indexet idx\_szul néven, amely a szuletett mező szerint indexeli az evek táblát:

CREATE INDEX idx\_szul ON evek(szuletett);

Az index az evek tábla Szerkezet lapján is megjelenik, sőt létrehozni, módosítani és törölni is lehet itt. Nézd meg!



Utána listáztasd újra születési sorrendben a költőket! (Mivel nagyon kevés adat van, nem fogod észre venni a különbséget az időben.)

## E. Két érték közötti szűrés

Írasd ki azoknak a költőknek az adatait, akik 1919 és 1944 között haltak meg, ezt a két értéket is beleértve! Rendezd a listát a költők halálának éve szerint növekvő sorrendben! Ezt kétféleképpen is elvégezheted, az eredmény ugyanaz lesz. Próbáld ki!

Első megoldás:

SELECT \* FROM evek WHERE meghalt BETWEEN 1919 AND 1944 ORDER BY meghalt;

Második megoldás:

SELECT \* FROM evek WHERE meghalt>=1919 AND meghalt<=1944 ORDER BY meghalt;

## F. Üres mezők

Az üresen hagyott mezőkbe a NULL érték kerül. Erre is lehet szűrni. Listáztasd ki a *masolat* táblából először azokat a költőket, akiknél üres a hely mező, majd azokat, akiknél nem üres! Figyeld meg, hogyan kell megadni a feltételt!

SELECT \* FROM masolat WHERE hely IS NULL;

SELECT \* FROM masolat WHERE hely IS NOT NULL;

Gyakorlásképpen az üresen hagyott hely mezőkbe írasd be a ’nincs adat’ szöveget! Ellenőrizd az eredményt!

UPDATE masolat SET hely='nincs adat' WHERE hely IS NULL;

## G. Számított mezők

Vannak olyan adatok, amelyeket azért nem tárolunk, mert a többi adatból kiszámíthatóik. Ilyen például az, hogy az egyes költők hány évig éltek. Írasd ki ezt a számított mezőt az evek táblából!

SELECT nev,meghalt-szuletett FROM evek;

A SELECT után tetszőleges kifejezéseket írhatsz vesszőkkel elválasztva, ezeket a program minden sorra kiszámítja és kiírja. A lista fejlécében a kifejezés jelenik meg az oszlop fölött.

A kifejezéshez megadhatsz egy nevet is, és akkor olvashatóbb lesz a fejléc. Nem csak számított mezőkhöz, hanem normál mezőkhöz is megadhatsz így nevet:

SELECT nev AS Név,meghalt-szuletett AS Év FROM evek;

## H. Feladatok

Végezd el a következő feladatokat az adatbázisban!

1. Írasd ki a leghosszabb ideig élt költő nevét és korát az *evek* táblából! Parancs:
2. Javítsd ki Petőfi Sándor születési helyét a *masolat* táblában Kiskőrösre!   
   Parancs:
3. Exportáld az adatbázist a *koltok.sql* nevű fájlba! Nézd meg, hogyan adja meg az indexet az ALTER TABLE paranccsal! (Ezt is lehet használni a CREATE INDEX helyett.)