# Létrehozás

## Új adatbázis létrehozása

Most elkészítjük az előző leckében már használt *koltok* adatbázist.

1. Indítsd el az XAMPP programot, futtasd a web és a MSQL szervereket, majd indítsd el a PHPMyAdmin programot!
2. Most SQL paranccsal hozzuk létre az adatbázist. Nyisd meg a konzolt, és írd be a következő parancsot:  
   CREATE DATABASE koltok   
   CHARACTER SET utf8mb4   
   COLLATE utf8mb4\_hungarian\_ci;  
   Indítsd el a parancsot (Ctrl+Enter), majd nézd meg az eredményt!
3. A létrehozás után a további munkához ki kell jelölni az adatbázist. Most elég rákattintani a nevére a bal oldalon. Ha parancssorban (vagy programból) dolgozunk, akkor a következő parancsot kell futtatni a kijelöléshez:  
   USE koltok;  
   Jelölj ki egy másik adatbázist, majd próbáld ki!

## Tábla létrehozása

Egy tábla létrehozásakor meg kell adni a tábla nevét, valamint a mezők neveit és típusait. A nevekben csak az angol ABC betűit és számokat használjunk, ékezetes betűket és szóközt ne!

A MySQL leggyakoribb **adattípusai**:

* INT vagy INTEGER: előjeles egész szám,
* DOUBLE vagy REAL: tizedes tört,
* VARCHAR(n): legfeljebb n karakterből álló szöveg (n<=255),
* TEXT: tetszőleges hosszúságú szöveg,
* DATE: dátum (A dátumokat szövegként is kezelhetjük.)
* BOOLEAN: logikai érték (számként tárolja, 1 = igaz, 0 = hamis)

Rengeteg más adattípust is használhatunk, ezeket később ismerjük majd meg.

A táblák összekapcsolásához szükség van egy olyan mezőre, amely egyértelműen azonosítja a sorokat. Ezt nevezik **elsődleges kulcsnak**. Ennek értékét mindig meg kell adni (nem lehet üres), és minden rekordban (sorban) különbözőnek (egyedinek) kell lennie.

Készítsd el most az *evek* táblát! Minden költőnek:

* lesz egy azonosítója (az - egész),
* megadjuk a nevét (nev – szöveg, legfeljebb 40 karakter),
* a születési évét (szuletett – egész)
* és halálának évét (meghalt – egész).

Az elsődleges kulcs az `az` mező lesz.

Azt szeretnénk, hogy ne nekünk kelljen beírni az azonosítókat, hanem a program 0-tól kezdve automatikusan egyesével számozza a rekordokat. Ezt az AUTO\_INCREMENT megadásával fogjuk beállítani. Így biztosak lehetünk benne, hogy nem lesz két ugyanolyan azonosítójú rekord. Ha törlünk egy rekordot, akkor sem használja fel újra a számát.

Az adattáblában az üres mezőkbe NULL kerül. Ez jelzi, hogy nincs benne semmi.

Ha azt szeretnénk, hogy kötelező legyen megadni egy mező tartalmát, akkor a NOT NULL-t is meg kell adni a mező neve után. Az elsődleges kulcsot mindig kötelező megadni, így ott erre nincs szükség (de megadható).

Írd (vagy másold) be a következő SQL parancsot (kisbetűkkel is lehet) a konzolra, majd futtasd le:

CREATE TABLE evek (  
 az INT AUTO\_INCREMENT,   
 nev VARCHAR(40) NOT NULL,   
 szuletett INT,   
 meghalt INT,  
 PRIMARY KEY (az)  
);

A parancsot többször nem lehet futtatni, mert már létezik ilyen nevű tábla. A hibaüzenetek elkerülése érdekében így is megadhattuk volna a parancsot:

CREATE TABLE IF NOT EXISTS evek (  
 az INT AUTO\_INCREMENT,   
 nev VARCHAR(40) NOT NULL,   
 szuletett INT,   
 meghalt INT,  
 PRIMARY KEY (az)  
);

Ez azt jelenti, hogy a parancs csak akkor próbálja meg létrehozni a táblát, ha még nem létezik.

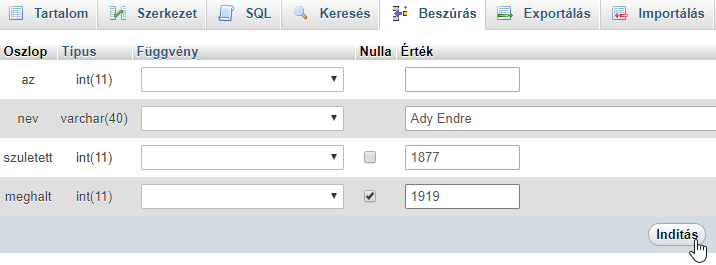
Jelöld ki az *evek* táblát, és nézd meg a Szerkezet lapon a tábla szerkezetét!

Figyeld meg, hogy az INT típus neve után zárójelben odaírta, hogy 11! Ez azt jelenti, hogy kiíráskor legfeljebb 11 számjegy jelenhet meg. Megadhattunk volna zárójelben egy kisebb számot is (pl. INT(4)).

Megjegyzés: Ha a PHPMyAdmin állítja elő a parancsokat, akkor fordított aposztófok közé teszi a neveket, de erre általában nincs szükség.

Nézd meg a Tartalom lapon, hogy a tábla még üres!

## Adatok bevitele

Az adatok bevitelét a PHPMyAdmin a Beszúrás lapon segíti. Próbáld ki először ezt! Töltsd ki az adatokat az alábbiak szerint, majd kattints az Indítás gombra! Az `az` mezőre ne adj meg értéket, mert azt automatikusan számozza!  


Figyeld meg, hogy parancs végrehajtása után átvált az SQL lapra, és mutatja, milyen SQL parancsot hajtott végre! Nézd meg a parancsot!

Ellenőrizd a Tartalom lapon!

A tábla végéhez az INSERT INTO paranccsal lehet egy új sort hozzáadni. Vidd be először Arany János adatait a konzolon a következő paranccsal, majd listázd ki a tábla tartalmát:

INSERT INTO evek (nev,szuletett,meghalt) VALUES ('Arany János', 1817, 1882);   
SELECT \* FROM evek;

Megjegyzés: Ha minden mezőre sorban megadtunk volna értéket, akkor az első zárójelet a mezőnevekkel elhagyhattuk volna.

Vidd be a többi költő adatait is! Nem szükséges mindhez külön parancs, hanem egy paranccsal is megoldható:

INSERT INTO evek (nev,szuletett,meghalt) VALUES   
('Babits Mihály',1883,1941),  
('Csokonai Vitéz Mihály',1773,1805),  
('Illyés Gyula',1902,1983),  
('József Attila',1905,1937),  
('Kölcsey Ferenc',1790,1838),  
('Petőfi Sándor',1823,1849),  
('Radnóti Miklós',1909,1944),  
('Tóth Árpád',1886,1928);

Ellenőrizd az eredményt!

Szeretnénk Juhász Gyula adatait is bevinni a táblába, de nem tudjuk a szükséges évszámokat:

INSERT INTO evek (nev) VALUES ('Juhász Gyula');

Nézd meg, mi került a táblába!

Figyeld meg a következőket:

* A számozást automatikusan folytatta
* A nev mezőbe bekerült a megadott név.
* Az évszámok helye üresen maradt. Valójában ide a NULL érték került, amely az üres mezőt jelenti.

## Exportálás

Az exportálás során egy szöveges fájlba írjuk azokat az utasításokat, amelyekkel bárhol, bármikor egy ugyanilyen szerkezetű és tartalmú adatbázist létrehozhatunk. (Másik adatbázis-kezelőben is!)

Jelöld ki az **adatbázist**, és válaszd az Exportálás fület, majd kattints az Indítás gombra!

Add meg a mentés helyét és nevét (koltok.sql), és mentsd a fájlt!

Nézd meg a koltok.sql tartalmát! Figyeld meg, hogy elég bonyolult. Nem baj, ha még nem értesz mindent!

## Tábla törlése

Egy táblát tartalmával együtt (!) a DROP TABLE paranccsal törölhetsz. Vigyázat! Utána nem lehet visszavonni! Próbáld ki:

DROP TABLE evek;

Ellenőrizd a törlést, majd zárd be a programot!

## Feladatok

1. Egy új adattáblát szeretnénk létrehozni *tanulok* néven, amelyben a tanulók neve (legfeljebb 50 karakter), születési éve és tanulmányi átlaga szerepel.   
   Feltételezzük, hogy nincs két ugyanolyan nevű tanuló, ezért a nev lesz az elsődleges kulcs.  
   Egészítsd ki az alábbi parancsot:  
   \_\_\_\_\_\_ TABLE tanulok (  
    nev \_\_\_\_\_\_\_,  
    szulev \_\_\_\_\_\_\_\_,  
    atlag \_\_\_\_\_\_\_,  
    \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
   );
2. Írd ide azt az SQL parancsot, amellyel Nagy Béla adatait adhatod hozzá a tanulok táblához, aki 2001-ben született és átlaga 4.23!  
   \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
3. Írd ide azt a parancsot, amellyel letörölhetnénk az előbb létrehozott táblát a tartalmával együtt!  
   \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_