Kiss Júlia

Balogh Tamara

Hodosi Veronika

adatbázis projekt

Tartalom

[Bevezetés 2](#_Toc166624389)

[Adatok keresése 2](#_Toc166624390)

[Logikai modell 3](#_Toc166624391)

[Fizikai modell 4](#_Toc166624392)

[Adatok beolvasása 5](#_Toc166624393)

[Kapcsolatok 7](#_Toc166624394)

[Lekérdezések 8](#_Toc166624395)

[0. Lekérdezés: Az adatbázis összes adatának megjelenítése 8](#_Toc166624396)

[1. Lekérdezés: Csapatok sorrendbe rendezése a kapott gólok és szerzett gólok különbségei alapján 9](#_Toc166624397)

[2. Legnagyobb férőhelyű stadion csapatának elnöke 10](#_Toc166624398)

[3. Fordulóként a gólszámok növekvő sorrendben 10](#_Toc166624399)

[4. Pancho arénában mikor játszottak, kik és összesen hány gólt lőttek 11](#_Toc166624400)

[5. Csapatonként szereplések darabszáma és helyezések átlaga 12](#_Toc166624401)

# Bevezetés

Feladatunk volt, hogy egy SQL adatbázist tervezzünk, aminek minimum 3NF-ben kell lennie, és legalább öt táblát tartalmaz. Ennek az adatbázisnak kell csinálni egy logikai és fizikai modellt, és lekérdezésekkel kiegészíteni.

Repository link: <https://github.com/tamibalogh/adatbazis_beadando.git>

# Adatok keresése

Első küldetésünk az volt, hogy az adott választható témák közül (személyi kölcsönök nyilvántartása, telefon előfizetések nyilvántartása, háziorvosi rendelő adatbázisa, labdarúgó bajnokság mérkőzéseinek nyilvántartása, autósiskola adatbázisa, többfordulós tehetségkutató verseny adatbázisa, fodrászüzlet adatbázisa, autókereskedő adatbázisa) kitaláljuk, hogy milyen adatbázist szeretnénk a megfelelő adatokkal. Mi végül a labdarúgó bajnokság mérkőzéseinek nyilvántartását választottuk, mert a Wikipédián található 2022-2023-as magyar labdarúgó bajnokság (első osztály) táblázatait mi tökéletesnek tartottuk a feladathoz.

Adatok forrása: <https://hu.wikipedia.org/wiki/2022%E2%80%932023-as_magyar_labdar%C3%BAg%C3%B3-bajnoks%C3%A1g_(els%C5%91_oszt%C3%A1ly)?fbclid=IwZXh0bgNhZW0CMTAAAR3ptMbJgj_9FVrAACI_6WcYqdK8_njsIg_c3_kDUkSRABttRhjZDOrNoXQ_aem_AWAYlZlcaBFzD0TAvi8g73bU8w_wUZIiJgU3rZp2BpbPvF10uMPw3yUUFEevLR9_PsbHaLySDGB64B5FFT7jt1Ly>

# Logikai modell

Mivel megtaláltuk már az adatainkat, ezért fel tudtuk térképezni, hogy hogyan is nézzen ki az adatbázisunk. Ebben segített a logikai és fizikai modell feltérképezése.

A Csapatok tábla az egész adatbázisunk szíve. Ehhez majdnem az összes többi táblázat kapcsolódik, hiszen ez tartalmazza a csapatok adatait. A csapatok azonosítóját nem számokkal jelöltük, mert úgy éreztük, hogy akkor nagyon nehezen tudnánk lekövetni, hogy melyik csapathoz melyik azonosító tartozik. Ezért az egyik táblázatból a forrás oldalán átvettük a rövidítéseket, de erről majd később, az adatok beolvasásakor bővebben írunk.

A Stadionok táblában található az egyes csapatok alá tartozó stadionok, azok férőhelyei, és építésüknek éve.

A Gollovolista táblában észrevehető a szezon góllövőlistája, ahova azokat tároljuk, akik minimum öt gólt lőttek.

A Merkozesek táblában meg lehet tekinteni az egyes mérkőzések adatait.

A Fordulok táblában kiolvasható a fordulók dátuma, amivel könnyeben tudjuk követni a mérkőzéseket.

A képen szöveg, képernyőkép, diagram, sor látható

Automatikusan generált leírás

# Fizikai modell

A fizikai modellben meg tudjuk tekinteni, a kulcsokat, és az egyes oszlopok fajtáit. Ezt a logikai modellel egy alkalmazásban csináltuk, az Oracle Datamodeler-ben.

Oracle Datamodeler: <https://www.oracle.com/database/sqldeveloper/technologies/sql-data-modeler/>

A Csapatok táblában mindegyik oszlop szöveget (nvarchar) használ, csak a hosszuk más. Itt az elsődleges kulcs az ID.

A Stadionok táblában az elsődleges kulcs a STADION\_ID, és az idegen kulcs a CSAPATOK\_ID. Itt már vannak oszlopok, amik számokat tartalmaznak, amiket integer-el jelöltünk.

A Gollovolista táblában szintén az idegen kulcs a CSAPATOK\_ID, és az elsődleges kulcs az a JATEKOS neve.

A Merkozesek tábla azért különleges, mert itt elsődleges és idegen kulcsként a HAZAI\_CSAPAT\_ID és a VENDEG\_CSAPAT\_ID is megjelenik, amik összetett kulcsot alkotnak. Mindkét kulcs ugyanúgy az ID-ra mutat a Csapatok táblában.

A Fordulok táblában az elsődleges kulcs a FORDULO\_ID. Itt van olyan oszlopunk, ami dátumokat (date) tartalmaz.

A képen szöveg, képernyőkép, sor, Betűtípus látható

Automatikusan generált leírás

# Adatok beolvasása

Mivel az adatokat Wikipédiáról szereztük, ezért könnyen le lehetett őket tölteni CSV formájában. Az adatok beolvasására, meg később a lekérdezésekre is, az SQL Online Editor-t használtuk.

SQL Online Editor: <https://sqliteonline.com/>

A Csapatok adatait a Csapatok adatai táblázatból vettük, ahonnan kivágtuk a szerelés, mezszponzor és a csapatok szezonja oszlopokat és hozzáadtuk az ID-t, ami a stadionok táblázatában voltak.

A képen szöveg, Betűtípus, képernyőkép, szám látható

Automatikusan generált leírásA képen szöveg, képernyőkép, szám, Párhuzamos látható

Automatikusan generált leírás

A Stadion tábla adatait az előbb említett táblázatból vettük ki. A STADION\_ID-t adtuk hozzá, ami csak egy számláló legfőképp.

A képen szöveg, képernyőkép, szám, Betűtípus látható

Automatikusan generált leírás

A Gollovolista táblának adatait a Góllövőlista táblázatból vettük. Itt a csapatok teljes nevét kicseréltük a CSAPAT\_ID-val.

A képen szöveg, képernyőkép, szám, Betűtípus látható

Automatikusan generált leírás

A Merkozesek tábla adatait a Mérkőzések táblázatból vettük. A CSAPATOK\_ID-t szétválasztottuk, és két külön kulcsot csináltunk belőle, mert különben kétszer több adatunk lett volna, ami 33 forduló mentén azt jelentette volna, hogy összesen 33\*6\*2=396 sornyi adat lett volna ebben a táblában. Így csak 198 sor van a táblában. A kapott és szerzett gólok a hazai csapat szempontjából vannak fenntartva.

A képen szöveg, képernyőkép, szám, Betűtípus látható

Automatikusan generált leírás

A Fordulok tábla is a Mérkőzések táblázat adatait használta fel, hiszen ott vannak feltűntetve az egyes fordulók hivatalos mérkőzésnapjaik. A FORDULO\_ID megegyezik a fordulók számával.

# Kapcsolatok

Külön kitérek a táblák közötti kapcsolatra. A Csapatok és a Stadionok tábla közötti kapcsolat a legkülönlegesebb, hiszen azok egy az egyhez kapcsolatot folytatnak, hiszen minden csapatnak csak egy stadionja van. Az összes többi kapcsolat, ami a Csapatok táblából indul egy a többhöz kapcsolat, szintúgy igaz ez a Fordulókból kiindult kapcsolatra is. Ez azt jelenti, hogy egy adat egy táblában többször jelenik meg.

# Lekérdezések

Mint ahogy már előbb meg lett említve, a lekérdezéseket ugyanott végeztük, mint az adatok beolvasását, az SQL Online Editor-ban.

## Lekérdezés: Az adatbázis összes adatának megjelenítése

Ellenőrző lekérdezés, amivel be tudjuk bizonyítani, hogy jól teljesítettük a az adatok berakását az adatbázisba. Ennek a megoldása és eredménye látható.

A képen szöveg, képernyőkép, szoftver, Multimédiás szoftver látható

Automatikusan generált leírás

## Lekérdezés: Csapatok sorrendbe rendezése a kapott gólok és szerzett gólok különbségei alapján

Itt legfőképp arra kellett figyelni, hogy ha a csapat a vendégcsapatnak volt beírva az adott sorba, akkor nekik fordítva vannak a kapott és szerzett gólok. Ezt össze is lehet hasonlítani a Wikipédián talált egyik táblázat adataival, amivel megegyezik. Ennek a megoldása és eredménye itt megtekinthető.

A képen szöveg, képernyőkép, szoftver látható

Automatikusan generált leírás

A képen szöveg, képernyőkép látható

Automatikusan generált leírás

## Legnagyobb férőhelyű stadion csapatának elnöke

Itt csak egyetlen egy adat jelenik meg megoldásként, amiért használtuk a TOP utasítást. Ennek az eredménye és megoldása olvasható ki.

A képen szöveg, képernyőkép, Betűtípus látható

Automatikusan generált leírás

## Fordulóként a gólszámok növekvő sorrendben

Ennek rengeteg megoldása van, például használhattuk volna a PARTITION BY módszerét is, de mi végül a klasszikust választottuk. Ennek a megoldása és eredménye található.

A képen szöveg, képernyőkép, szoftver látható

Automatikusan generált leírás

## Pancho arénában mikor játszottak, kik és összesen hány gólt lőttek

Itt egy specifikus eredményt kerestünk, nem úgy, mint az előzőknél, ahol vagy a legelső adatot, vagy egy külön táblát kértünk le. Ennek a megoldása és eredménye látható.

A képen szöveg, képernyőkép, szoftver látható

Automatikusan generált leírás

## Csapatonként szereplések darabszáma és helyezések átlaga

A legutolsó lekérdezésnél ugyanaz igaz, mint a harmadik lekérdezésre, hogy ezt is megoldhattuk volna a PARTITION BY használatával. Ennek a megoldása és eredménye itt megtekinthető.

A képen szöveg, képernyőkép, szoftver látható

Automatikusan generált leírás