



győri szakképzési centrum

Jedlik Ányos
Gépipari és Informatikai
Technikum és Kollégium



9021 Győr, Szent István út 7.

+36 (96) 529-480

+36 (96) 529-448

OM: 203037/003

jedlik@jedlik.eu

www.jedlik.eu

Vizsgáló neve:.....

Szakmai vizsga

Projektfeladat

Az ágazat megnevezése: **Informatika és távközlés**

Szakma megnevezése: **Szoftverfejlesztő és -tesztelő technikus szakma**

A vizsgatevékenység megnevezése:

Asztali- és webes szoftverfejlesztés, adatbázis-kezelés feladatsor

Időtartam: **240 perc**

A vizsgatevékenység aránya a teljes szakmai vizsgán belül:

90%

Jóváhagyta:

Dátum: 2024.

.....

Módos Gábor
igazgató

2024



Vizsgaszervező neve:

Győri SZC Jedlik Ányos Gépipari és Informatikai Technikum és Kollégium

Vizsga dátuma:

.....

Vizsga helyszíne:

Györi SZC Jedlik Ányos Gépipari és Informatikai Technikum és Kollégium
9021 Győr, Szent István út 7.

Használható segédeszköz:

A gyakorlati vizsgatevékenység során a jelölt, a feladat kidolgozása közben az alábbi eszközöket használhatja:

- **internethasználattal rendelkező asztali számítógép**
- **bármilyen írott vagy nyomtatott segédeszköz (saját jegyzet, könyv, stb.)**
- **bármilyen kommunikációra nem alkalmas elektronikus adattárolón található szoftvertermék (pl.: saját pendrive használható, mobiltelefon nem)**
- **Az internetkapcsolatot nem korlátozzuk internetcímekre, vagy időtartamra. Az internet a vizsga teljes időtartama alatt a vizsgázók számára szabadon rendelkezésre áll.**
- **Az internetkapcsolatot a vizsgázó nem használhatja semmilyen mesterséges intelligenciával kapcsolatos kommunikációra, kódgenerálásra. (pl.: ChatGPT, stb.) A vizsga során tilos Visual Studio Code-ban MI alapú kiegészítőket használni (pl.: GitHub CoPilot, Tabnine, Intellicode, AWS CodeWhisperer, stb.).**
- **Az internetkapcsolat használata során a vizsgázó nem tölthet le olyan anyagokat, amik:**
 - **iskolai szerveren helyezkednek el,**
 - **iskola által üzemeltetett távoli meghajtón helyezkednek el (pl.: Google Drive, GitHub),**
 - **az iskolai oktatói, dolgozói saját tulajdonában levő távoli meghajtón helyezkednek el,**
 - **a szakmajegyzékes Szoftverfejlesztő és -tesztelő szakma szakmai vizsgájának felkészítésre szolgáló weboldalon helyezkednek el, (pl. infojegyzet.hu)**
 - **olyan nyilvánosan elérhető szoftvertermékeket, melynek keletkezési időpontja a vizsga kezdési időpontjánál későbbi.**

Mentés:

- **A vizsgabizottság által megadott helyen hozzon létre <név>_<osztály> néven könyvtárat!**
- **A saját könyvtárán belül a megoldásai rendezze a feladatok jellege alapján elnevezett alkönyvtárba!**
- **A frontend és backend feladatrészek esetében a leírás alapján törölje a felesleges állományokat, majd a betömörített megvalósítását helyezze el a megfelelő könyvtárba!**



1 Grafikus és konzolos részt egyaránt tartalmazó asztali alkalmazás fejlesztése - Friends

A következő feladatban a Jóbarátok (Friends) című sorozat epizódjaival kapcsolatos információkat tartalmazó állományok állnak rendelkezésre, melyekkel programozási feladatokat kell megoldania.

A feladat megoldása során vegye figyelembe a következőket:

- *A képernyőre írást igénylő részfeladatok eredményének megjelenítése előtt írja a képernyőre a feladat sorszámát (például: 2. feladat:)!*
- *Az egyes feladatokban a kiírásokat a minta szerint készítse el!*
- *Az ékezetmentes azonosítók és kiírások is elfogadottak.*
- *Az azonosítókat kis- és nagybetűkkel is kezdheti.*
- *A program megírásakor az állományban lévő adatok helyes szerkezetét nem kell ellenőriznie, feltételezheti, hogy a rendelkezésre álló adatok a leírtaknak megfelelnek.*
- *A megoldását úgy készítse el, hogy az azonos szerkezetű, de tetszőleges bemeneti adatok mellett is helyes eredményt adjon!*

Az adatforrások két formában állnak rendelkezésre, melyek közül egyet kell választania a feladat megoldásához.

A. `episodes.json` és `seasons.json` elnevezésű JSON típusú szöveges fájlakként,

B. `episodes.csv` és `seasons.csv` elnevezésű pontosvesszővel tagolt állományokként

Figyelj rá, hogy az 3. feladatrész hasonló nevű fájljai itt nem használhatóak, más szerkezettel és tartalommal bírnak.

Az adatforrásokban a következő adatokat találja meg:

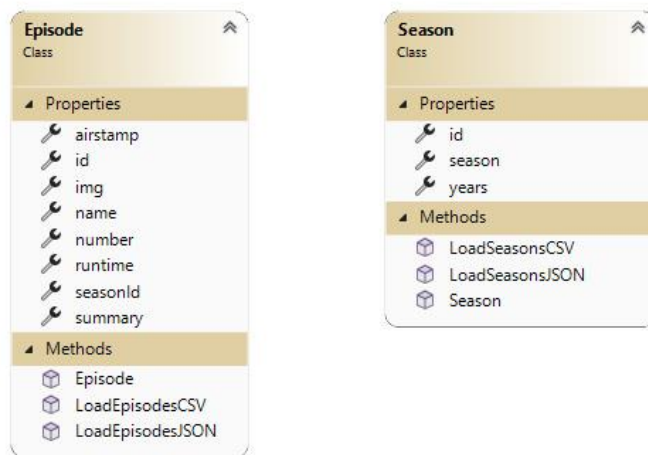
episodes.csv / episodes.json tartalma:

- *id*: az epizód azonosítója, egész szám
- *name*: az epizód címe, szöveg
- *seasonId*: a szezon azonosítója, melyben az epizód megjelent, egész szám
- *number*: az epizód sorszáma az évadon (szezoonon) belül, egész szám
- *airstamp*: az epizód műsorra kerülésének ideje, dátum
- *runtime*: az epizód hossza percben, egész szám
- *img*: az epizódhoz tartozó képállomány elérési útvonala, szöveg
- *summary*: az epizód történetének rövid összefoglalója, szöveg

seasons.csv / seasons.json tartalma:

- *id: a szezon azonosítója, egész szám*
- *season: a szezon sorszáma szöveges formában, szöveg*
- *years: a kezdő és vég évszám kötőjellel elválasztva, melyben a szezon futott, szöveg*

1. Készítsen konzolos alkalmazást a következő feladatok megoldására, melynek projektjét `Friends_Console` néven mentse el!
2. Készítsen saját osztályokat `Episode` és `Season` azonosítóval, melyek adattagjainak és metódusainak azonosítóit osztálydiagramok szemléltetik! Valósítsa meg, hogy az egyes tulajdonságok értékeit csak lekérdezni lehessen más osztályokból, azoknak értéket adni kizárólag a konstruktorban, a példányosítás során lehessen!



3. Készítsen statikus metódust az `Episode` osztályhoz az adatforráshoz történő kapcsolódásra és az adatforrásban lévő adatok betöltésére! A metódus neve a kiválasztott fájlformátumnak megfelelő legyen, JSON állomány esetén `LoadFromJSON`, míg a CSV állomány esetén `LoadFromCSV`! A metódus visszatérési értéke legyen az `Episode` osztályból képzett lista, míg paramétere a betöltendő fájl neve legyen!
(Az osztálydiagramokon található fájlolvasó metódusok közül elegendő azt elkészítenie, amely az Ön megoldásában kiválasztott fájlformátumnak megfelelő névvel rendelkezik!)
4. A 3. feladatban leírtaknak megfelelő módon a `Season` osztályban hozzon létre egy statikus metódust az adatok betöltésére!
5. Olvassa be a választott adatforrás adatait és tárolja az adatokat az `Episode` és `Season` osztály segítségével egy olyan összetett adatszerkezetben, amely használatával a további feladatok megoldhatóak!
6. Határozza meg, hogy hány olyan epizód van, melynek címében szerepel valamelyik női főszereplő neve (*Monica, Rachel, Phoebe*) és írassa ki ezen epizódok címét a



győri szakképzési centrum

Jedlik Ányos
Gépipari és Informatikai
Technikum és Kollégium



9021 Győr, Szent István út 7.

+36 (96) 529-480

+36 (96) 529-448

OM: 203037/003

jedlik@jedlik.eu

www.jedlik.eu

képernyőre a mintának megfelelően! A kiíratásban minden epizód címe csak egyszer jelenjen meg!

7. Határozza meg, hogy hány nap volt a legtöbb, ami eltelt két egymást követő szezon utolsó és első része között! Írassa ki a mintának megfelelően a képernyőre az eltelt napok számát, valamint azt, hogy melyik két évad záró és nyitó epizódja között telt el a legtöbb nap!

Minta:

6.feladat: Összesen 16 db epizód van, mely tartalmazza valamely női főszereplő nevét
Címek:

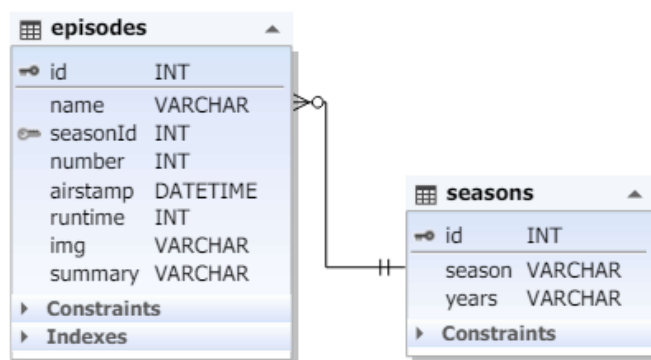
The One With the Fake Monica
The One Where Rachel Finds Out
The One With Phoebe's Husband
The One With Phoebe's Dad
The One Where Ross and Rachel... You Know
The One Where Rachel Quits
The One Where Monica and Richard are Just Friends
The One With Phoebe's Ex-Partner
The One Where Ross and Rachel Take a Break
The One With Phoebe's Uterus
The One With Rachel's Crush
The One With Rachel's New Dress
The One After Ross Says Rachel
The One Where Phoebe Hates PBS
The One With Rachel's Inadvertent Kiss
The One Where Rachel Smokes

7.feladat:

A(z) 4. évad utolsó és a(z) 5. évad első részének megjelenése között telt el a legtöbb idő, 140 nap.

A grafikus alkalmazás elkészítése előtt a lokális adatbáziskezelő rendszerébe importálja be a friends.sql fájlt, majd futtassa azt! A szkript futtatásának eredményeképpen a lokális adatbáziskezelő rendszerében létrejött friends néven az UTF-8 kódolású, magyar egybevetésű adatbázis! A továbbiakban a programozási feladatokat ennek az adatbázisnak a használatával kell megoldania!

Az adatbázis struktúrája:



1. Nyissa meg a `Friends_GUI` nevű WPF projektet a forrásállományok közül! A további feladatokat ebben a projektben kell elkészítenie! A projekt már tartalmazza a feladat megoldásához szükséges `Episode` és `Season` osztályokat, így azokat használhatja a megoldás során!
2. Készítse el a grafikus felületet a mintának megfelelően! A megjelenő ablak reszponzív viselkedésű legyen! Az ablakban levő vezérlők típusai és a feliratok feleljenek meg a mintának!
3. Csatlakozzon a korábban létrehozott `friends` adatbázishoz! Az ablak tetején található legördülő listát töltse fel az egyes szezonok `season` mezőjének adataival. Valósítsa meg, hogy a program indulásakor az első szezon kerüljön kiválasztásra!
4. A legördülő lista alatt található táblázatban jelenjenek meg azok az epizódok, melyek ahhoz az évadhoz tartoznak, amelyet a felhasználó az előző feladatban létrehozott legördülő listában kiválasztott! Az egyes epizódok adatait a mintának megfelelően jelenítse meg! A műsorra tűzés dátuma a magyar formátumnak megfelelően jelenjen meg! Ügyeljen rá, hogy nem kell az epizódok minden adatát megjelenítenie, csak azokat, melyek a mintán is szerepelnek!
5. Valósítsa meg, hogy amennyiben a felhasználó a táblázatban kiválaszt egy epizódot, úgy a táblázat alatt jelenjen meg az adott epizódhoz tartozó kép az ablak bal oldalán, míg az ablak jobb felén az adott rész címe és összefoglaló szövege a mintának megfelelően! Megoldása több pontot ér, ha a feladatot adatkötéssel oldja meg. Valósítsa meg, hogy az összefoglaló szöveg tördelésre kerüljön, ha az hossza miatt nem fér el egy sorban! Az epizód címét igazítsa középre és tegye félkövérré!



Minták:

Friends

Season: first


ID	Title	Season	Number	Airtime	Runtime
646	The One Where It All Began	1	1	1994.09.22 20:00	30
647	The One With the Sonogram at the End	1	2	1994.09.29 20:00	30
648	The One With the Thumb	1	3	1994.10.06 20:00	30
649	The One With George Stephanopoulos	1	4	1994.10.13 20:00	30
650	The One With the East German Laundry Detergent	1	5	1994.10.20 20:00	30
651	The One With the Butt	1	6	1994.10.27 20:00	30
652	The One With the Blackout	1	7	1994.11.03 20:00	30
653	The One Where Nana Dies Twice	1	8	1994.11.10 20:00	30
654	The One Where Underdog Gets Away	1	9	1994.11.17 20:00	30

Az ablak a program indulását követően

Friends

Season: third

ID	Title	Season	Number	Airtime	Runtime
694	The One With the Princess Leia Fantasy	3	1	1996.09.16 20:00	30
695	The One Where No One's Ready	3	2	1996.09.26 20:00	30
696	The One With the Jam	3	3	1996.10.03 20:00	30
697	The One With the Metaphorical Tunnel	3	4	1996.10.10 20:00	30
698	The One With Frank Jr.	3	5	1996.10.17 20:00	30
699	The One With the Flashback	3	6	1996.10.31 20:00	30
700	The One With the Race Car Bed	3	7	1996.11.07 20:00	30
701	The One With the Giant Poking Device	3	8	1996.11.14 20:00	30
702	The One With the Football	3	9	1996.11.21 20:00	30



The One With Frank Jr.

The gang compiles freebie lists, Phoebe tries to get to know her half brother, Joey builds an entertainment center.

Az ablak egy epizód kiválasztását követően



győri szakképzési centrum

Jedlik Ányos
Gépipari és Informatikai
Technikum és Kollégium



9021 Győr, Szent István út 7.

+36 (96) 529-480

+36 (96) 529-448

OM: 203037/003

jedlik@jedlik.eu

www.jedlik.eu

2 Reszponzív viselkedésű weboldal – 2023 Downhill

A következő feladatban a 2023-as downhill világbajnokságról szóló weboldalt kell módosítani a feladatleírás és a minta szerint. Ahol a feladat másként nem kéri, a formázási beállításokat a `style.css` stílusállományban végezze el úgy, hogy az új szelektorokat az állomány végén helyezze el!

Nagyobb felbontású, színes mintákat a kész weboldallról a `minta01.png`, `minta02.png`, `minta03.png`, `minta04.png` és a `minta05.png` állományokban talál, melyet tilos a megoldásában felhasználni!

Nyissa meg az `index.html`, `ranglista.html`, `csapatok.html`, `bolt.html`, a `bolt.js` és a `style.css` állományokat és szerkessze azok tartalmát az alábbiak szerint:

1. Az `index.html` oldalról hiányzik egy záró `html` elem. Derítse fel ezt a hibát, ha kell használja a <https://validator.w3.org/> validátort és javítsa a hibát! (1 pont)
2. A helyszíneket tartalmazó oldalon (`helyszinek.html`) szűrjön be harmadik helyszíneként egy újabb helyszínt! Az új helyszínt a meglévőkkel azonos `html` elemek és tulajdonságok segítségével valósítsa meg!
A helyszín adatait megtalálja a `helyszin.txt` állományban. (1 pont)
3. Állítsa be megfelelő Bootstrap osztályok beszúrásával, hogy a helyszínek 576 képpontnál keskenyebb kijelzőn egy oszlopban, az ennél szélesebb, de 768 képpontnál még keskenyebb kijelzőn 2 oszlopban, az ennél nagyobb kijelzőkön 4 oszlopban jelenjenek meg! (1 pont)
4. A `ranglista.html` oldalon a táblázat első sorának adatcelláit alakítsa át fejléc típusúra! (1 pont)
5. A `style.css` végére szűrjön be megfelelő szelektort, hogy a táblázat fejlécében lévő adatok dőlten jelenjenek meg, és legyenek középre igazítva! (1 pont)
6. A `csapatok.html` oldalon lévő képekhez készítsen új szelektort, már meglévő stílusosztályok felhasználásával úgy, hogy amikor az egyik csapat képe fölé visszük az egeret, akkor a kép körül 5 képpont vastag piros keret jelenjen meg! (Csak a `csapatok.html` oldalon lévő képekre működjön ez a beállítás és ne használjon új osztály vagy azonosító kijelölőt!) (2 pont)
7. A `csapatok.html` oldalon állítsa be, hogy a cím (A 2023-as downhill világbajnokság csapatai) azonos stílussal jelenjen meg, mint a többi oldalon! A megvalósításhoz ne hozzon létre új stílusosztályt és ne használjon `inline` stílust! (1 pont)
8. Végezze el a `bolt.html` oldalon azt a beállítást, amellyel hozzárendelheti az oldalhoz a `bolt.js` fájlt! (1 pont)



győri szakképzési centrum

Jedlik Ányos
Gépipari és Informatikai
Technikum és Kollégium



9021 Győr, Szent István út 7.

+36 (96) 529-480

+36 (96) 529-448

OM: 203037/003

jedlik@jedlik.eu

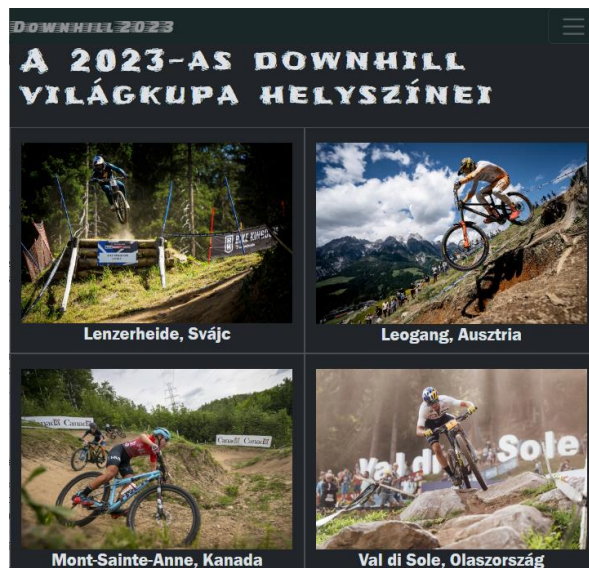
www.jedlik.eu

9. A `bolt.html` oldalon egy űrlap található. Az űrlapon, ha új kerékpárt választunk, jelenjen meg a kiválasztott kerékpár képe! A kép megjelenítéséért felelős függvény a `bolt.js` fájlban már létezik! (1 pont)

Minta oldalak:



index.html (nagy méretű kijelzőn)



helyszinek.html (tablet nézet)



győri szakképzési centrum

Jedlik Ányos
Gépipari és Informatikai
Technikum és Kollégium



9021 Győr, Szent István út 7.

+36 (96) 529-480

+36 (96) 529-448

OM: 203037/003







jedlik@jedlik.eu

www.jedlik.eu

DOWNHILL 2023

DOWNHILL VILÁGRANGLISTA 2023


NŐK

Helyezés	Versenyző	Nemzet	Pont
1	 Valentina Höll	 Austria	1842
2	 Nine Hoffmann	 Germany	1418
3	 Marine Cabirou	 France	1328
4	 Monika Hraštník	 Slovakia	1021

ranglista.html (mobil nézet)


DOWNHILL 2023

A 2023-AS DOWNHILL VILÁGKUPA CSAPATAI



SANTA CRUZ SYNDICATE

A Santa Cruz Syndicate 2006-ban indult azzal a magasztos céllal, hogy a világkupa-



SPECIALIZED GRAVITY

A Team Specialized Gravity története 2016-ban kezdődött a Pure Agency (igazgató:

csapatok.html (közepes kijelző)

DOWNHILL 2023

CSÚCSKATEGÓRIÁS DOWNHILL BICAJOK

Santa Cruz V10



Megveszem

Az ára: 9499 \$

bolt.html (nagy méretű kijelző)



győri szakképzési centrum

Jedlik Ányos
Gépipari és Informatikai
Technikum és Kollégium



9021 Győr, Szent István út 7.

+36 (96) 529-480

+36 (96) 529-448

OM: 203037/003

jedlik@jedlik.eu

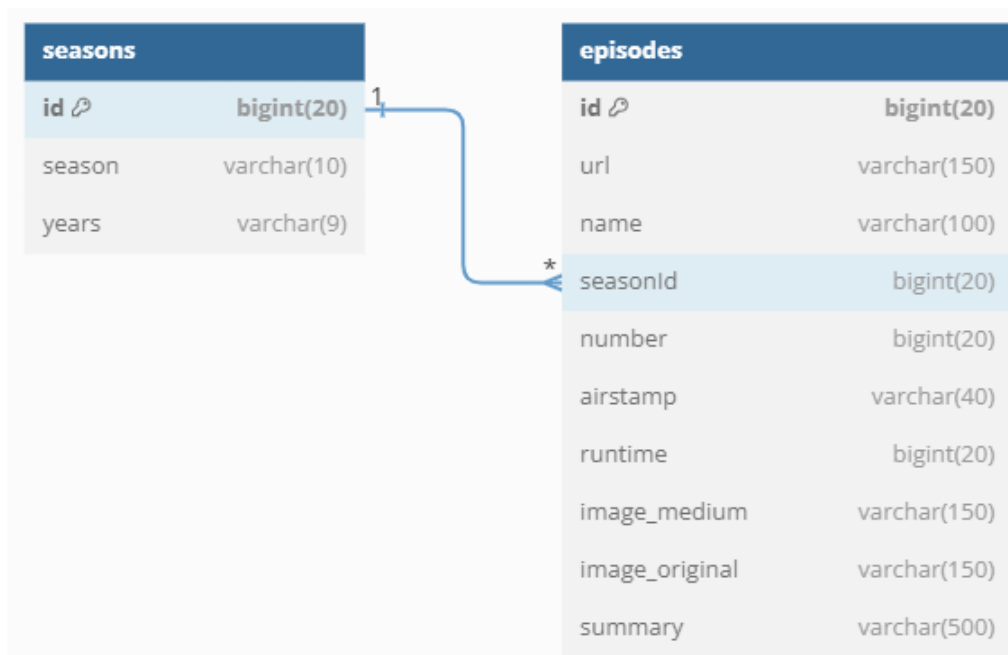
www.jedlik.eu

3 Backend programozás – Friends

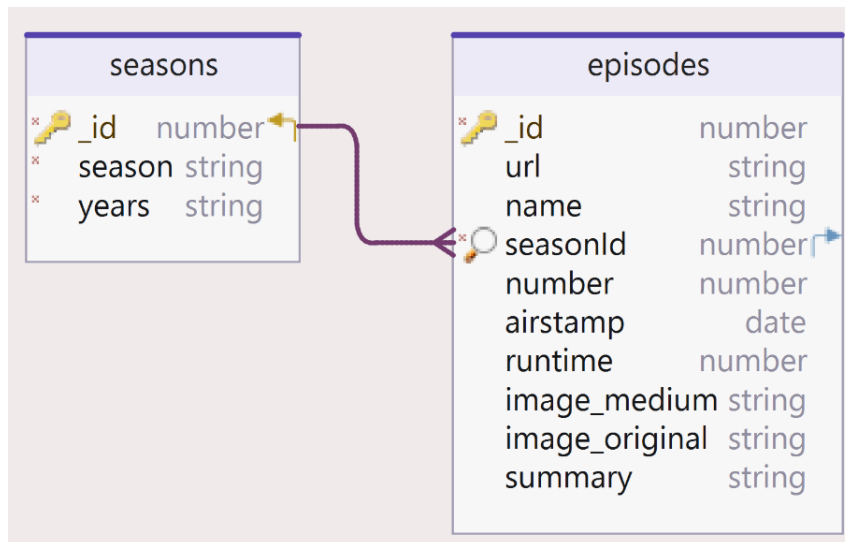
Az alábbi feladatban a Jóbarátok (Friends) sorozat adatait kezelő alkalmazás backend szerverét kell elkészítenie.

1. Hozzon létre backend server projektet az Ön által választott programnyelven, illetve fejlesztési környezetben! A projektmappát „Vezetéknév_Keresznév_backend” formában nevezze el!
2. Készítsen adatbázist `friends` néven!
3. A lenti ábrák a feladathoz tartozó adatbázis tábláit (kollekcióit), azok mezőit, a mezők típusait mutatják a választott technológiától függően. Ezek alapján:
 - a. hozzon létre modelleket a megfelelő adattípusokkal és beállításokkal!
 - b. az adatbázisban hozza létre a két táblát!

Relációs modell:



NoSQL adatbázis modell:



4. Alapértelmezett értékek:

episodes tábla:

image_medium -> „https://bgs.jedlik.eu/no_image.png”

image_original -> „https://bgs.jedlik.eu/no_image.png”

seasonId -> legyen indexelve

A létrehozott végpontok teszteléséhez – a választott technológiától függően – a *episodes* táblához tartozó *episodes.[csv/json]* és a *seasons* táblához a *seasons.[csv/json]* fájlokban található minta adatokat használhatja.

Figyelj rá, hogy az 1. feladatrész hasonló nevű fájljai itt nem használhatóak, más szerkezettel és tartalommal bírnak.

A további feladatok megoldása során ügyeljen arra, hogy a megoldása az elvárt végpontokon kívül egyéb végpontokat ne tartalmazzon! A végpontok funkcionalitását megvalósító kontroller(ek)ben a szükséges metódusokon kívül egyéb (felesleges) kódokat ne hozzon létre!



győri szakképzési centrum

Jedlik Ányos
Gépipari és Informatikai
Technikum és Kollégium



9021 Győr, Szent István út 7.

+36 (96) 529-480

+36 (96) 529-448

OM: 203037/003

jedlik@jedlik.eu

www.jedlik.eu

5. Kérdezze le egy adott évadhoz tartozó epizódok adatait! A válasz tartalma a lenti mintáktól eltérhet a választott technológiának megfelelően, de a mintákban szereplő összes adatot tartalmaznia kell. Ha az évad megnevezése nem létezik az adatbázisban, akkor „*Hibás évad!*” üzenettel és **404 NOT FOUND** státuszkóddal térjen vissza!

Metódus	URL	Body	Válasz
GET	/api/search-by-season/{season} <i>pl.: /api/search-by-season/fourth</i>	üres	JSON array/szöveg

Lehetséges válasz üzenet (**200 OK** státuszkóddal):

```
[
  {
    "id": 719,
    "url": "http://www.tvmaze.com/episodes/40719/friends-4x01-the-one-with-the-jellyfish",
    "name": "The One With the Jellyfish",
    "seasonId": 4,
    "number": 1,
    "airstamp": "1997-09-25T20:00:00-04:00",
    "runtime": 30,
    "image_medium":
    "http://static.tvmaze.com/uploads/images/medium_landscape/45/114628.jpg",
    "image_original":
    "http://static.tvmaze.com/uploads/images/original_untouched/45/114628.jpg",
    "summary": "At the beach house, Ross and Rachel appear on the threshold of a new \r\nera; Phoebe gets to know her birth mother (Teri Garr); Monica is stung \r\nby a jellyfish."
  },
  {
    "id": 720,
    "url": "http://www.tvmaze.com/episodes/40720/friends-4x02-the-one-with-the-cat",
    "name": "The One With the Cat",
    "seasonId": 4,
    ...
  ]
```

Lehetséges válasz üzenet (**404 NOT FOUND** státuszkóddal):

```
{
  "message": "Hibás évad!"
}
```



győri szakképzési centrum

Jedlik Ányos
Gépipari és Informatikai
Technikum és Kollégium



9021 Győr, Szent István út 7.

+36 (96) 529-480

+36 (96) 529-448

OM: 203037/003

jedlik@jedlik.eu

www.jedlik.eu

6. Új évaddal bővül a sorozat. Hozzon létre végpontot az új évad rögzítésére az alábbi beállításokkal! Sikeres rögzítés esetén **201 CREATED** státuszkóddal és az újonnan beszúrt rekord azonosítóját tartalmazó JSON üzenettel térjen vissza! Bármely kötelező mező hiánya esetén **400 BAD REQUEST** hibakóddal térjen vissza és a hibaüzenetben legyen megtalálható a „Hiányos adatok!” szöveg!

Kötelező mezők: season, years

Metódus	URL	Body	Válasz
POST	/api/add-season	JSON	JSON/szöveg

Példa a kérés body-ban elküldött JSON tartalmára:

```
{
  "season": "eleven",
  "years": "2023-2024"
}
```

Válasz üzenet (**201 CREATED** státuszkóddal):

```
{
  "id": 11
}
```

Lehetséges válasz üzenet (**400 BAD REQUEST** státuszkóddal):

```
{
  "message": "Hiányos adatok!"
}
```


7. Hozzon létre végpontot az adatbázisban évad törlésére az alábbi beállításokkal! Az `{id}` helyén a törölni kívánt évad azonosítója szerepeljen! Sikeres törlés esetén **204 NO CONTENT** státuszkóddal térjen vissza! Amennyiben a megadott `id` nem létezik **404 NOT FOUND** státuszkóddal és „Az évad nem létezik!” üzenettel térjen vissza! Amennyiben a megadott évad létezik, de egy vagy több rekord (epizód) hivatkozik rá az „N” oldalon, akkor ne törölje az évadot! Ebben az esetben térjen vissza a **403 FORBIDDEN** státuszkóddal és „Az évad nem törölhető!” üzenettel!

Metódus	URL	Body	Válasz
DELETE	<code>/api/delete-season/{id}</code> pl.: <code>/api/delete-season/11</code>	üres	üres/szöveg

Lehetséges válasz üzenet (**404 NOT FOUND** státuszkóddal):

```
{
  "message": "Az évad nem létezik!"
}
```

Lehetséges válasz üzenet (**403 FORBIDDEN** státuszkóddal):

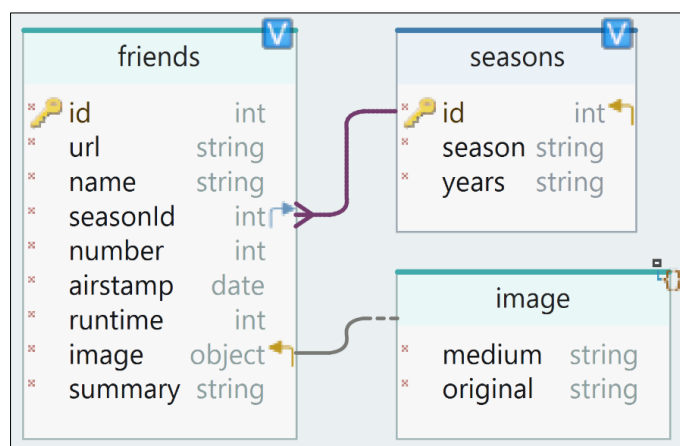
```
{
  "message": "Az évad nem törölhető!"
}
```

8. Hozzon létre Thunder Client vagy Postman kollekciót, melyben az Ön által létrehozott összes útvonalat ellenőrzi! Exportálja a kollekciót a projekt gyökér könyvtárba „Backend_teszt” néven!
9. A választott technológiának megfelelően adja le:
- a teljes projektmappát tömörítve (`node_modules` könyvtár nélkül)
 - az adatbázist létrehozó SQL scriptet vagy állományokat (`db.bat` és `*.json`)
 - az előző feladatban exportált „Backend_teszt” nevű kollekciót

4 Frontend programozás - Friends

A következő feladatban a népszerű Jóbarátok TV-sorozat (Friends) epizódjait bemutató alkalmazást kell készítenie. A webalkalmazást a kiadott leírás szerint kell elkészítenie, a felhasználói felület kialakítását minta segíti, melyet nem kell követnie a megoldásában.

A backend szerver által kezelt adatbázis tábláit és a táblák kapcsolatát a következő ábra szemlélteti:



A feladat elkészítése során a `http://localhost:3000` címen található REST API szerverrel kell dolgoznia. A szerver indításához másolja a források mappából a Backend mappát a „D” megnevezésű főkönyvtárba, majd indítsa el az API szervert a `D:\Backend\backend.bat` állomány futtatásával!

A lokális szerver route-jaival teljesen megegyező online szervert a <https://friends-series.jedlik.cloud> címen érheti el. A feladatok megvalósítására ezt is használhatja.



Ebben a feladatban használandó REST API végpont:

Metódus	URL	Body	Válasz
GET	/api/friends?name_like={ filter }&_page={ page }&_limit=15&_expand=season	üres	JSON array/szöveg

Válasz (200-as státuszkóddal):

A végpont azokkal a Jóbarátok epizódok adataival tér vissza, melyek címében (name mező) megtalálható a megadott {**filter**} karaktersorozat. A végpont „oldalakra tördelve” adja vissza a rekordokat, 15-15 db-ot oldalakként. Az oldal sorszámát 1-től N-ig a {**page**} paraméter határozza meg.

API válasz minta a következő végpont hívásra:

/api/friends?name_like=**where**&_page=**2**&_limit=15&_expand=season

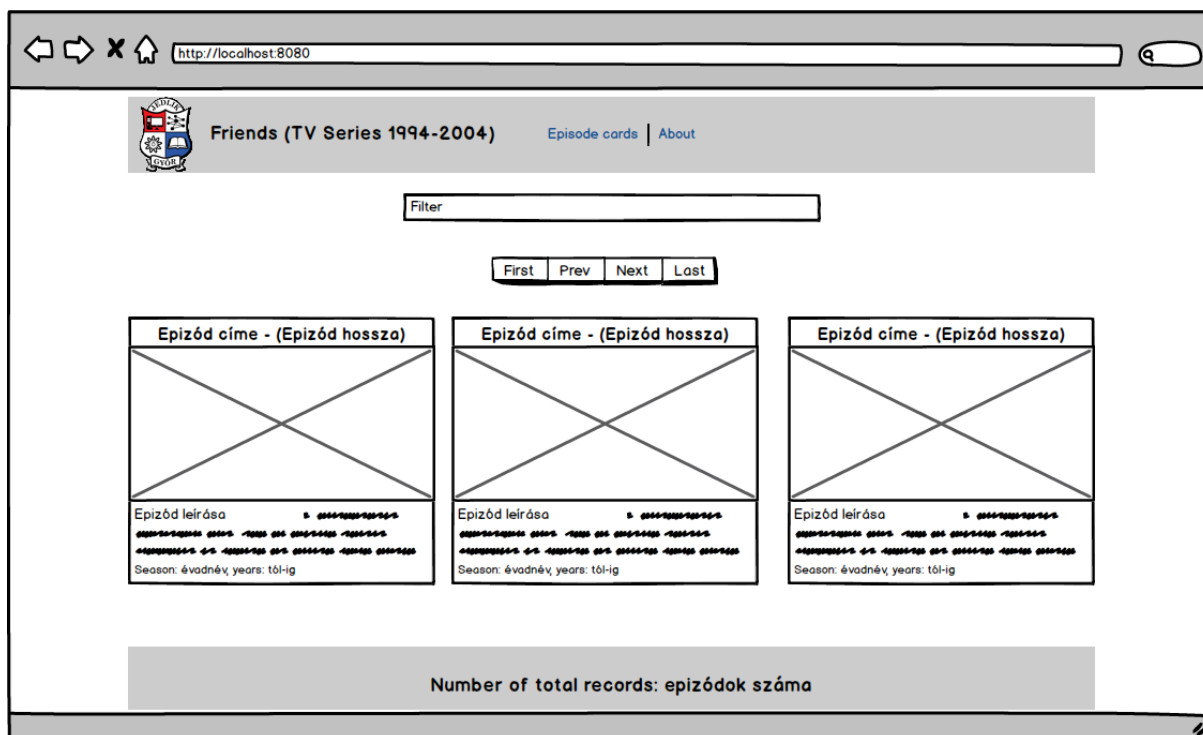
```
[
  {
    "id": 704,
    "url": "http://www.tvmaze.com/episodes...sister",
    "name": "The One Where Chandler Can't Remember Which Sister",
    "seasonId": 3,
    "number": 11,
    "airstamp": "1997-01-09T20:00:00-05:00",
    "runtime": 30,
    "image": {
      "medium": "http://.../medium_landscape/45/114588.jpg",
      "original": "http://.../original_untouched/45/114588.jpg"
    },
    "summary": "A drunken Chandler has a ... Rachel for a job.",
    "season": {
      "id": 3,
      "season": "third",
      "years": "1996-1997"
    },
  },
  ...
]
```



Feladatok

Az Ön által választott frontend keretrendszer (Angular, Vue, stb.) valamelyikével készítse el az alábbi feladatokat! A feladatok elkészítését forrásállományok és minták segítik, melyek a választott technológiának megfelelő alkönyvtárban találhatóak a források mappában. A minták egy-egy lehetséges megoldást mutatnak be, pontosításra a kért funkciók megvalósítása kerül.

1. Hozzon létre frontend projektjének Vezetéknév_Keresztnév_techológia mappát!
2. Az alábbi minta és a kiadott források segítségével építse fel az alkalmazás menü- és állapotsorát!



A feladat vue.js technológiával történő megvalósítását megnézheti a `vue-minta1.jpg` és a `vue-minta2.jpg` állományokban!

3. Oldja meg, hogy az „*Episode cards*” és az „*About*” menüpontok kiválasztásával a megfelelő oldalakat töltsse be (routolja) az alkalmazása! Az alkalmazás indulásakor az „*Episode cards*” oldal jelenjen meg automatikusan!
4. Az „*Episode cards*” oldalon a beviteli mező alapértelmezett értéke a „**where**” szó legyen! Ebben a beviteli mezőben adjuk meg az epizód címére (name) vonatkozó szűrési feltételt.



győri szakképzési centrum

Jedlik Ányos
Gépipari és Informatikai
Technikum és Kollégium



9021 Győr, Szent István út 7.

+36 (96) 529-480

+36 (96) 529-448

OM: 203037/003

jedlik@jedlik.eu

www.jedlik.eu

5. Az „*Episode cards*” oldal betöltésekor jelenítse meg a „where” szóra szűrés esetén megjelenő első 15 epizódot. A backend hívásához az alábbi url használható:
`/api/friends?name_like=where&_page=1&_limit=15&_expand=season`
Az url megtalálható az `api_végpont.txt` állományban is.
Megjelenítendő mezők: `name`, `runtime`, `image.medium`, `summary`,
`season.season`, `season.years`
6. A beviteli mező értékének megváltozásakor az adatokat újra kérdezze le az API végpontról, majd jelenítse meg az érkező epizódok adatait az oldalon!
7. Jelenítse meg a szűrési feltételnek megfelelő rekordok számát az alkalmazás láblécében!
8. Oldja meg, ha az „*About*” oldalra navigálunk, akkor ne jelenjen meg semmi az állapotsorban!
9. A kiadott mintában szereplő parancsgombok, vagy tetszőlegesen választott lapozó-komponens használatával valósítson meg „lapozás” funkciót! Lehesen az első, az előző, a következő és az utolsó oldalakra lapozni! A megvalósításához használja az `_page` paraméterét! Megoldása kevesebb pontot ér, ha a teljes adathalmazt letöltve csak kliens oldalon valósítja meg a lapozást.
10. Oldja meg, hogy az első és utolsó oldal kiválasztása esetén a megfelelő, lapozást vezérlő elemek (parancsgombok) kerüljenek inaktív állapotba!
11. Forráskódját a választott technológiának megfelelően adja le:
 - a. Tömörítés előtt törölje a felesleges állományokat! **Ügyeljen arra, hogy feladatmegoldást tartalmazó mappát/állományt ne töröljön!**
 - b. A forráskódját tömörítse be `Vezetéknév_Keresztnév_technológia.zip` néven!