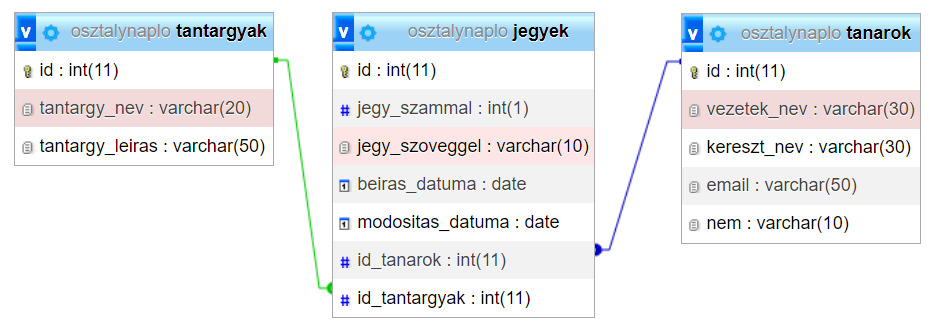
# Backend programozás

A teljesítmény értékelése fontos feladat az oktatás során. A következő feladatban iskolai adatbázist kell készítenie a jegyek, tantárgyak és tanárok adatainak a rögzítésére.

1. Készítsen adatbázist „osztalynaplo” néven.
2. A lenti ábra alapján:
   1. Hozza létre a táblákat megfelelő adattípusokkal és beállításokkal!
   2. Állítsa be a táblák kapcsolatát az ábrán látható módon. (1 tanárhoz és 1 tantárgyhoz több jegy is tartozhat.)
3. Oldja meg, hogy a tanár törlése esetén törlődjenek a hozzátartozó jegyek is, az adatbázisból.
4. Töltse fel a táblákat a forrás fájlban megadott rekordokkal a teszteléshez:A képen szöveg látható

   Automatikusan generált leírás

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!

1. Hozzon létre RestFull projektet C# programnyelven, a Visual Studio fejlesztési környezetben! A projektmappát „Vezetéknév\_Keresztnév\_backend” formában nevezze el! A feladat megoldása során ékezetmentes azonosítókat és állományneveket is használhat.
2. Adja hozzá a projekthez a következő package-eket:

A képen szöveg látható

Automatikusan generált leírás

1. A létrehozott adatbázis modellt a

**Scaffold-DbContext "server=localhost;database= osztalynaplo;user=root;password=;ssl mode=none;" mysql.entityframeworkcore -outputdir Models –f**

utasítás segítségével implementálja be a projektjébe az osztályok létrehozásához.

1. Állítsa be a modelleknél, a „jegyek” osztályban, hogy az összetett adatlekérés ne okozzon redundanciát a „JsonIgnore” segítségével, vagy illessze be a megfelelő helyre a következő sort:

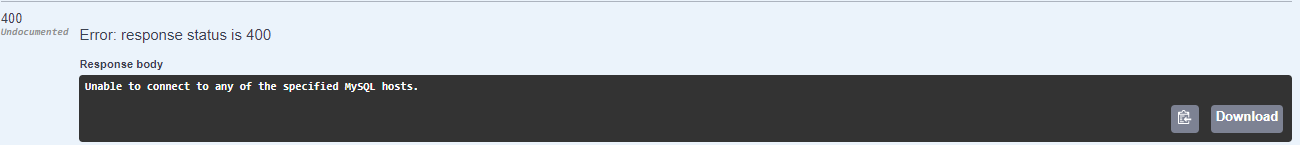
**builder.Services.AddControllers().AddJsonOptions(x => x.JsonSerializerOptions.ReferenceHandler = ReferenceHandler.IgnoreCycles);**

1. A képen Téglalap látható

   Automatikusan generált leírásHozzon létre végpontot az összes tantárgy és vele együtt minden adatának a kilistázására.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Metódus** | **Paraméter** | **URL** | **Body** | **Válasz** |
| **GET** |  | **https://localhost:7214/api/Tantargy/feladat9** | **üres** | **JSON** |
| **Lehetséges válasz üzenet (200 Success státuszkóddal):** | | | | |

Lehetséges válasz hibaüzenet üzenet (**400 Error** státuszkóddal):



1. A képen Téglalap látható

   Automatikusan generált leírásHozzon létre végpontot, ahol képes szűrni a tantárgy nevére és visszaadni a paraméterként megadott tantárgyat és hogy milyen jegyeket adtak a tanárok az adott tárgyból.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Metódus** | **Paraméter** | **URL** | **Body** | **Válasz** |
| **GET** | **Informatika** | **https://localhost:7214/api/Tantargy/feladat10** | **üres** | **JSON** |
| **Lehetséges válasz üzenet (200 Success státuszkóddal):** | | | | |

A képen Téglalap látható

Automatikusan generált leírás Lehetséges válasz hibaüzenet üzenet (**400 Error** státuszkóddal):

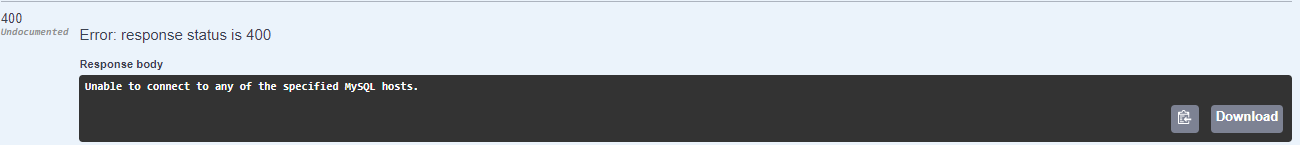
1. Hozzon létre végpontot az adatbázisban szerepelő tanárok adatainak a lekérdezésére az alábbi beállításokkal és paraméterekkel! Oldja meg, hogy a válasz üzenetben a „tanarok” tábla **összes** mezője és az általuk adott **összes** jegy minden adataival szerepeljenek.

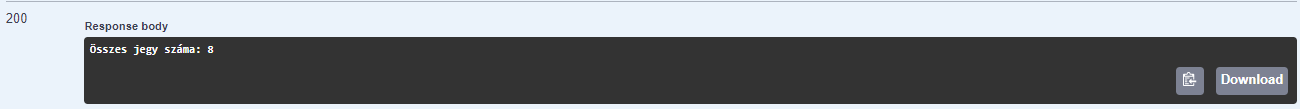
A válasz tartalma a lenti mintáktól eltérhet a választott technológiának megfelelően, de a „tanarok” táblában szereplő összes adatot tartalmaznia kell! Amennyiben nem sikerül az adatlekérés, akkor a válasz keletkezett hibaüzenet legyen 400-as hibakóddal.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Metódus** | **Paraméter** | **URL** | **Body** | **Válasz** |
| **GET** |  | **https://localhost:7214/api/Tanar/feladat11** | **üres** | **JSON** |
| **Lehetséges válasz üzenet (200 Success státuszkóddal):** | | | | |

A képen Téglalap látható

Automatikusan generált leírás

 Lehetséges válasz hibaüzenet üzenet (**400 Error** státuszkóddal):

1. Hozzon létre végpontot az adatbázisban szerepelő összes jegy számának a lekérdezésére az alábbi minta alapján!

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Metódus** | **Paraméter** | **URL** | **Body** | **Válasz** |
| **GET** |  | **https://localhost:7214/api/Jegy/feladat12** | **üres** | **JSON** |
| **Lehetséges válasz üzenet (200 Success státuszkóddal):** | | | | |

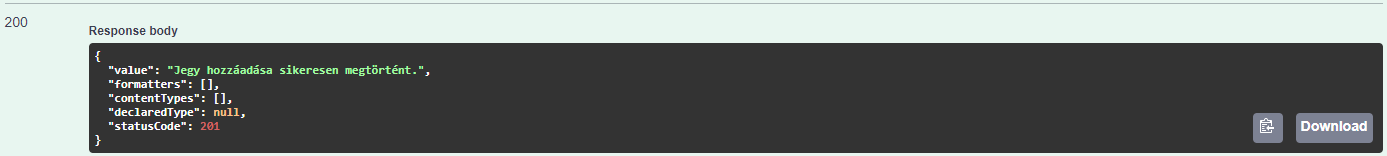
A képen Téglalap látható

Automatikusan generált leírás Lehetséges válasz hibaüzenet üzenet (**400 Error** státuszkóddal „Adatbázis nem elérhető!”):

1. Ha az **Asp.net Core 6**-os verzióját választotta a feladat megoldásához, akkor illessze be a megfelelő helyre a következő kódrészletet:

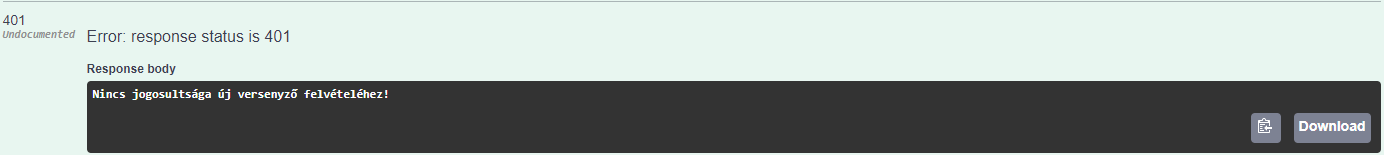
**builder.Services.AddControllers(options => options.SuppressImplicitRequiredAttributeForNonNullableReferenceTypes = true);**

Hozzon létre végpontot új jegy rögzítésére az alábbi feltételekkel és beállításokkal! A jegy rögzítése csak akkor lehetséges, ha a felhasználó be van jelentkezve és van jogosultsága hozzá. Ez akkor lehetséges, ha a paraméterben kapott „UserID” értéke „FKB3F4FEA09CE43C”. Oldja meg, hogy a főprogram tárolja ezt az azonosítót „UID” néven és a végpont hasonlítsa össze ezt a paraméterében kapott értékkel. Sikeres rögzítés esetén 201 CREATED státuszkóddal és a „Jegy hozzáadása sikeresen megtörtént.” JSON üzenettel térjen vissza! Hiba esetén 400 BAD REQUEST hibakóddal és a hibára utaló üzenettel térjen vissza! Amennyiben jogosultsági probléma lép fel, akkor 401 Unauthorized hibakóddal és „Nincs jogosultsága új versenyző felvételéhez!” üzenettel térjen vissza.



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Metódus** | **Paraméter** | **URL** | **Body** | **Válasz** |  |
| **POST** | FKB3F4FEA09CE43C | **https://localhost:7214/api/Jegy/feladat13?uid=FKB3F4FEA09CE43C** | **JSON** | **JSON** |  |
| **Lehetséges válasz üzenet (201 Success státuszkóddal):** | | | | |  |

Lehetséges válasz hibaüzenet üzenetek (**401 Error** státuszkóddal „Nincs jogosultsága új versenyző felvételéhez!”):



1. A „CORS” műveletek beállításához illessze be a következő két programrészletet a megfelelő helyekre.
2. **services.AddCors(c => { c.AddPolicy("AllowOrigin", options => options.AllowAnyOrigin().AllowAnyMethod().AllowAnyHeader()); });**
3. **app.UseCors(options => options.AllowAnyOrigin().AllowAnyMethod().AllowAnyHeader());**
4. Projektmappáját a választott technológiának megfelelően adja le:
   1. Tömörítés előtt törölje a felesleges állományokat! Ügyeljen arra, hogy feladatmegoldást tartalmazó mappát/állományt ne töröljön!
   2. A forráskódját tömörítse be Vezetéknév\_Keresztnév\_backend.zip néven!
   3. A tömörített állomány tartalmazza az adatbázist létrehozó SQL scriptet vagy JSON állományt is!