**Feladat: "ScholarshipApplicant" model és repository készítése**

**Cél:**

Készítsd el a **ScholarshipApplicant** modellt és a hozzá tartozó repository osztályt! A modelt és az adatokat egy SQLite adatbázisban fogjuk tárolni, amelyet a **scholarship.csv** fájl alapján scaffoldolunk.

**1. Adatok előkészítése**

A tesztadatokat egy CSV fájl tartalmazza (**scholarship.csv**):

email name amount

alice@example.com Alice Johnson 1500

bob@example.com Bob Smith 2000

charlie@example.com Charlie Brown 1800

diana@example.com Diana White 2200

**1. ScholarshipApplicant modell létrehozása**

Hozz létre egy ScholarshipApplicant osztályt, amely tartalmazza az alábbi elemeket:

* **Privát változók:** Két privát változót, amely tárolja az ösztöndíj összegét, valamint egy email címet.
* **Konstruktor:** Az osztály konstruktorában az email cím és a név inicializálása történik, ahol ellenőrizni kell kivétellel, hogy ezek nem üresek.
* **Publikus tulajdonságok:** A név amely lekérdezhető és módosítható, az email cím és az ösztöndíj összege amely a privát változókat adja vissza és azoknak ad értéket!
* **Metódus az ösztöndíj összegének növelésére:** Egy metódus, amely biztosítja, hogy csak pozitív számokkal növelhető az ösztöndíj összege. Ha a szám nem pozitív, dobjon kivételt
* **ToString metódus:** Egy metódus, amely visszaad egy formázott szöveget a névvel, email címmel és az aktuális ösztöndíj összeggel.

### ****2. Model réteg tesztelése****

Az elkészült ScholarshipApplicant osztály működésének ellenőrzésére készíts egy tesztet a Program.cs fájlban! Az alábbi lépéseket hajtsd végre:

1. **Hozz létre egy ScholarshipApplicant példányt üres email címmel,** és ellenőrizd, hogy kivételt dob-e.
2. **Hozz létre egy ScholarshipApplicant példányt "odon@nyertes.hu" email címmel és "Ösztöndíj Ödön" névvel,** majd írasd ki a konzolra az adatait.
3. **Próbálj meg -30000 összeget hozzáadni az ösztöndíjhoz**, és ellenőrizd, hogy kivételt dob-e.
4. **Adj hozzá 20000 és 30000 összeget az ösztöndíjhoz**, majd írasd ki az objektumot a konzolra.

**3. ScholarshipContext**

1. Scaffoldold az adatbázist a **scholarship.csv** fájl alapján.
2. A generált DbContext osztályban módosítsd a DbSet-et és az Entity konfigurációt úgy, hogy az általunk definiált **ScholarshipApplicant** osztályt használja.
3. A ScholarshipApplicant osztályban az **email címet használd kulcsként!**

A scaffoldolt DbContext osztályban végezd el a következő módosításokat:

1. Módosítsd a DbSet-et úgy, hogy az általad létrehozott ScholarshipApplicant osztályt használja.
2. Állítsd be az email mezőt elsődleges kulcsként.
3. Biztosítsd, hogy az amount mező egész számként (INT) legyen tárolva az adatbázisban.
4. Konfiguráld a táblanevet az adatbázisban.

**4. Repository osztály létrehozása**

Hozz létre egy ScholarshipRepo osztályt, amely felelős az adatbázis műveletekért!

**Feladat:**

1. A repository osztály feleljen az adatbázisban tárolt adatok eléréséért.
2. Implementálj egy metódust, amely visszaadja az adatbázisban lévő ösztöndíjpályázók számát.