

1. Hozz létre egy új projektet **EloadasProject** néven.
C# nyelv esetén a projekt típusa *ClassLibrary* legyen és .NET Core 3.1 vagy újabb keretrendszer verzióval dolgozz.
Java nyelv esetén használjon valamilyen függőség kezelőt (Maven vagy Gradle) a projekt létrehozásakor.
JavaScript nyelv esetén a projektet inicializálja valamely csomagkezelő alkalmazás számára (npm / yarn)
 - a. A projektben hozz létre egy osztályt **Eloadas** néven C# nyelv esetében az automatikusan generált osztályt nevezze át erre.
 - b. C# nyelv esetén adj hozzá a solutionhöz egy új nUnit vagy xUnit projektet **TestEloadasProject** néven.
 - i. Alternatív megoldásként új projekt helyett az EloadasProject-en belül hozz létre egy Tests mappát, ahova a tesztek kerülnek majd. Ebben az esetben a NuGet csomagkezelő segítségével telepítsd a szükséges külső könyvtárakat.
 - c. Java vagy JavaScript nyelv esetén telepítsen automatizált teszteléshez szükséges könyvtárakat (javasolt packagek: java- junit / javascript – jest)
 - d. A projekthez inicializálj egy git repót is. A későbbi feladatok megoldását ebbe a repoba commitold. Az elkészült projektet töltsd fel githubra.
2. Az **Eloadas** osztály
 - a. Az osztály egy adattaggal rendelkezzen. Ez egy **bool** értéket tartalmazó **mátrix** (2 dimenziós tömb) legyen, **foglalasok** néven. A mátrix elemei akkor legyenek igazak, ha az adott hely foglalt.
 - b. Hozz létre az osztályhoz egy **konstruktort**, amely 2 **egész paramétert** vár: **sorokSzama**, **helyekSzama**. Ez a 2 paraméter határozza meg a foglalások mátrix méretét.
 - i. A paramétereknek 0-nál nagyobboknak kell lennie, ellenkező esetben ArgumentExceptionot adjon vissza.
 - c. Valósítsd meg az alábbi függvényeket, C# esetén a get névvel kezdődő függvények esetében get propertyt használjon (pl. getTeli() függvény helyett Teli nevű property):
 - i. **bool lefoglal()** – Az első szabad helyet foglalja le.
 - ii. **int getSzabadHelyek()** – Adja vissza a szabad helyek számát. C# nyelv esetén get propertyt hozz létre **SzabadHelyek** néven.
 - iii. **bool getTeli()** – Adja vissza, hogy az előadás tele van-e. C# nyelv esetén get propertyt hozz létre **Teli** néven.
 - iv. **bool Foglalt(int sorSzam, int helySzam)** – Adja vissza, hogy az adott sor adott helye foglalt-e. A sor és helyszámot 1-től kezdődően várja a függvény. Érvénytelen paraméter esetén ez a függvény is dobjon ArgumentException-t.

Eloadas
-foglalalasok: bool[,]
+Eloadas(sorokSzama:int, helyekSzama:int)
+lefoglal(): bool
+getSzabadHelyek(): int
+getTeli(): bool
+foglalt(sorSzam:int, helySzam:int): bool

1. ábra Az előadás osztály diagramja

3. Teszteld az Eloadas osztályt automatizált tesztekkel. Legyen benne a helyes kimenet ellenőrzése, és a helytelen bemenetekre kapott kivételek ellenőrzése is. A tesztek egy külön fájlban (és amennyiben a teszt keretrendszer megköveteli külön osztályban) készítsd el, melynek nevével utalj rá, hogy az Eloadas osztály tesztéseit tartalmazza. A fájl (és osztály) elnevezésekor ügyelj a választott nyelv elnevezési konvencióira.

NUnit segédlet: <https://segedletek.level14.hu/2019/03/02/nunit.html>