# Konzolos alkalmazás

# C# ---------------Visual Studio 2022

1. Feladat

Készítsen konzolos alkalmazást a következő feladatok megoldására, melynek projektjét RealEstate néven mentse el!

Kiválasztjuk az alábbi project formát.

A képen szöveg, képernyőkép, Betűtípus, Multimédiás szoftver látható

Automatikusan generált leírás

Létrehozzuk az alábbi konfigurációban.A képen szöveg, képernyőkép, Betűtípus, szoftver látható

Automatikusan generált leírás

1. Feladat

Készítsen saját osztályokat Ad, Seller és Category azonosítóval, melynek az adattagjainak azonosítóit és láthatósági szintjét az osztálydiagramok szemléltetik! A privát Szakmai záróvizsga online tanórák, 2022. 04. 20. Szoftverfejlesztő és -tesztelő próba vizsgafeladatsor 2 adattagokat egy lakat szimbólum különbözteti meg a publikusaktól. Ékezetes azonosítókat is készíthet, illetve azokat kis- és nagybetűkkel is kezdheti.

A képen szöveg, képernyőkép, tervezés látható

Automatikusan generált leírás

A képen szöveg, képernyőkép, Betűtípus, dokumentum látható

Automatikusan generált leírás

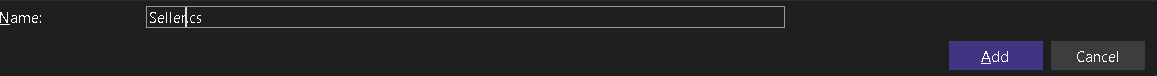
Elkészítjük a „Seller” osztályt.

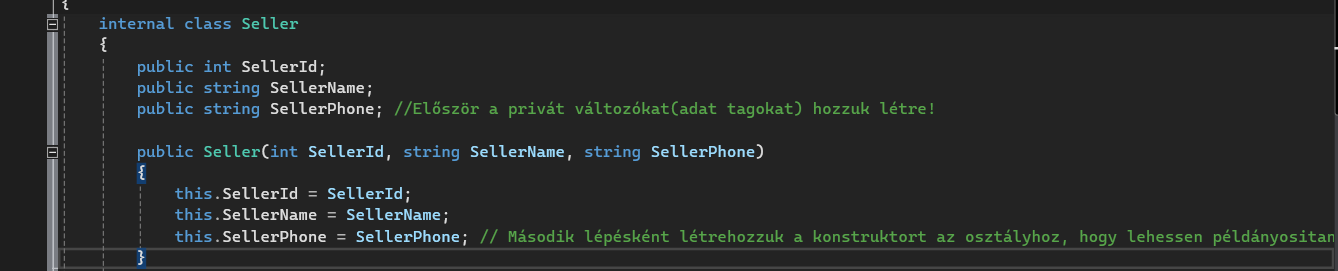
A képen szöveg, képernyőkép, szoftver, Multimédiás szoftver látható

Automatikusan generált leírás

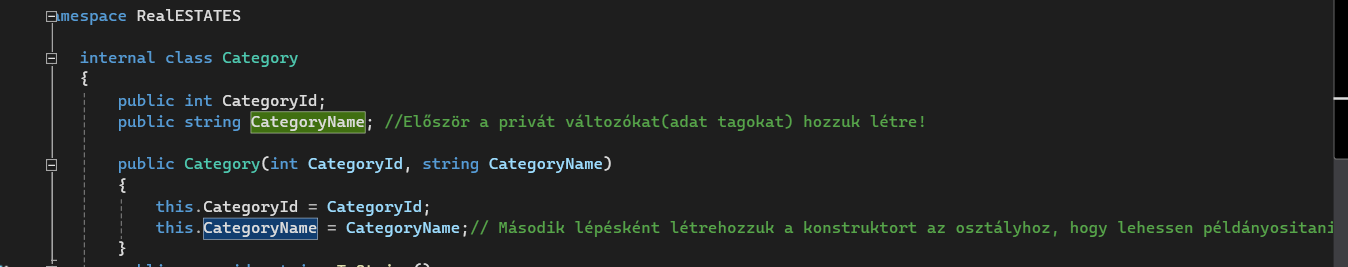
A képen szöveg, Multimédiás szoftver, szoftver, Grafikai szoftver látható

Automatikusan generált leírás



Az osztályba beírjuk az adatokat. Itt a privát osztály változó külön nevet kap, hogy könnyebb legyen vele dolgozni, a publikus változó neve az, amit a feladat kér.  


A „Category” osztályt a fenti minta alapján szintén elkészítjük és beírjuk az adatokat.



Az Ad osztályt a fenti minta alapján szintén elkészítjük és beírjuk az adatokat.

A képen szöveg, képernyőkép, szoftver látható

Automatikusan generált leírás

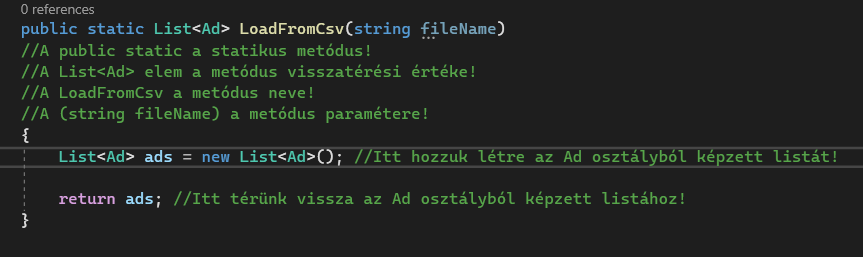
**JAVITÁS! A „FreeOfCharge” változót int-ként add meg, mert bool típussal nem működött!**

A képen szöveg, képernyőkép, szoftver, Multimédiás szoftver látható

Automatikusan generált leírás

1. Feladat

Készítsen statikus metódust az Ad osztályhoz az adatforráshoz történő kapcsolódásra, és az adatforrásban levő adatok betöltésére! A metódus neve utaljon az adatforrásból betöltésre (pl.: LoadFromCsv). A metódus visszatérési értéke az Ad osztályból képzett lista legyen! A metódus paramétere a betöltendő fájl neve legyen!

Ad.cs fájlban  


Program.cs fájlban határozzuk meg a fájl nevét.

A képen szöveg, képernyőkép, Multimédiás szoftver látható

Automatikusan generált leírás

1. Feladat

Szükség esetén az Ad osztály adattagjainak beállítását konstruktor segítségével állítsa be! A paraméterben kapott adatok az adatforrásoktól függően változóak lehetnek.

## EZT MÁR AZ OSZTÁLYOK LÉTREHOZÁSÁNÁL AZ ELSŐ FELADATBAN MEGTETTÜK!

1. Feladat

Olvassa be a választott adatforrás adatait és tárolja az adatokat az Ad osztály segítségével egy olyan összetett adatszerkezetben, amely használatával a további feladatok megoldhatók!

A LoadFromCsv metódusba írjuk ezeket!

A képen szöveg, képernyőkép, Betűtípus látható

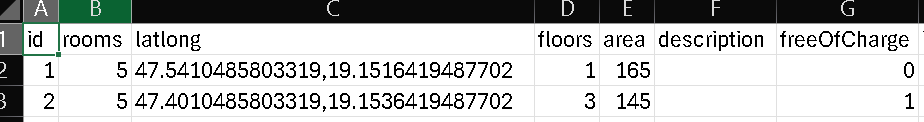
Automatikusan generált leírás

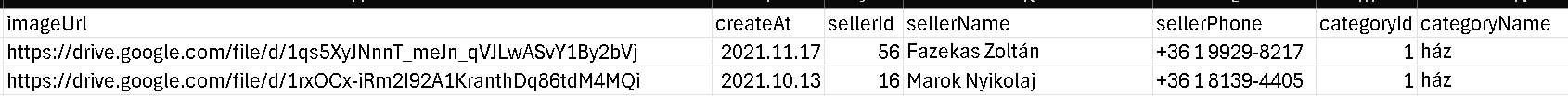
Ezután folytatjuk egy „while” ciklussal, ami végig megy a fájl összes során!

A képen szöveg, képernyőkép, Betűtípus látható

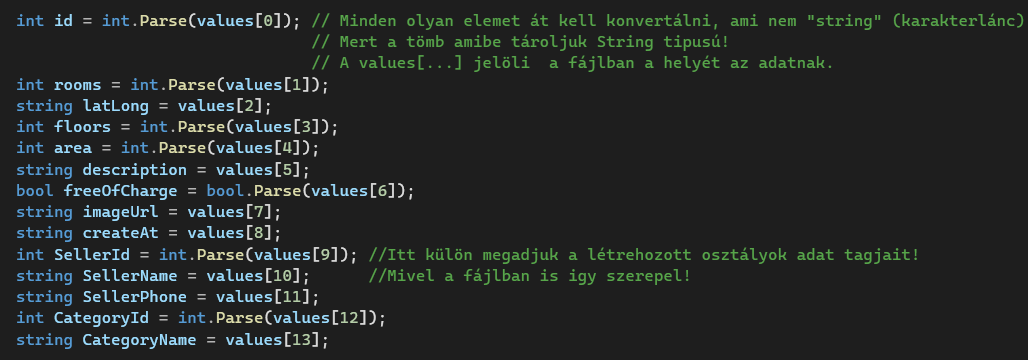
Automatikusan generált leírás

Majd megadjuk a rögzíteni kívánt adatokat és hogy hol található a fájlban.

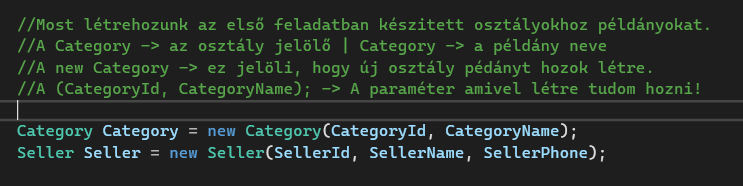




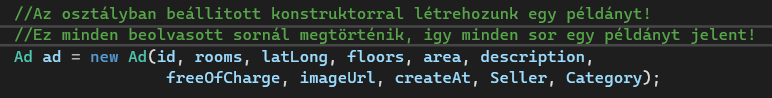
**Itt a tömb elemei a következők!  
values [0] 🡪 id || values [2] 🡪 rooms  
… A tömb(fájl) első eleme a „0” lesz és igy tovább! Itt összesen 14 oszlop van, igy 13 az elem szám, ha a „0” -val kezdjük!**

****

Példányositunk, hogy az itt felvett adatokból tudjunk egy Ad példányt csinálni majd, mivel az Ad osztályban az adattagoknál nem szerepel külön a categoryId, categoryName, SellerId, SellerName és SellerPhone változó!



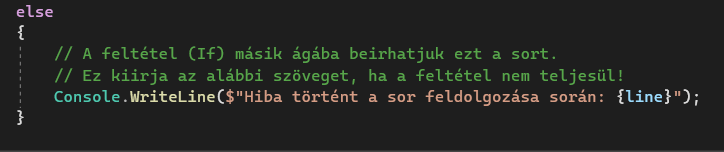
Ezután létrehozunk egy Ad példányt.



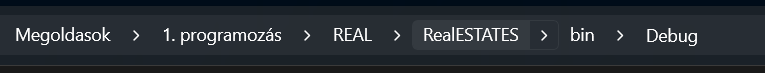
Majd a korábban létrehozott listánkban „ads” eltároljuk a létrehozott „ad” példányt.



Ez egy ellenőrzés, ha valami nem stimmel. (nem kötelező!)



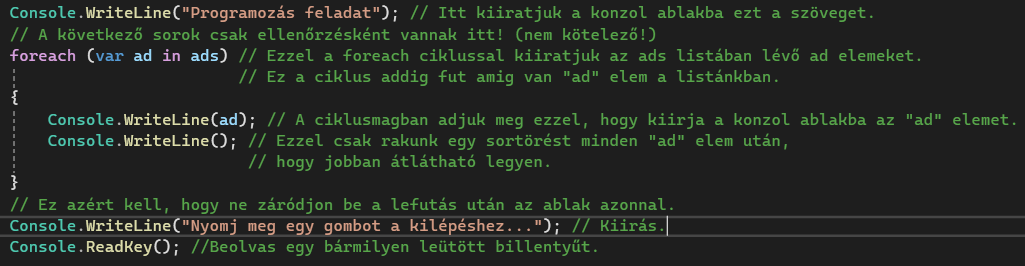
A forrás fájljainkat futtatás előtt be kell másolni a projekt könyvtár …/bin/debug mappájába!



A képen szöveg, képernyőkép, Betűtípus, szoftver látható

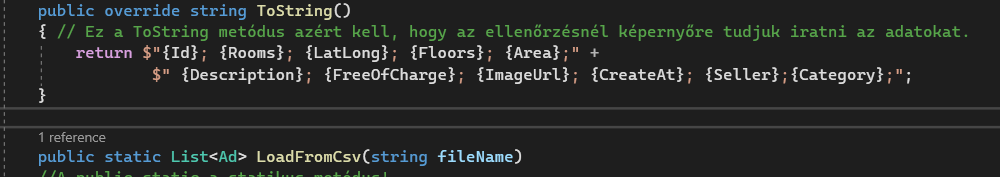
Automatikusan generált leírás

Ez csak egy önellenőrzést szolgál! (nem feltétlen kell!)



Kellenek „ToString” metódusok, hogy olvashatóan írassuk ki a képernyőre!

Ad.cs

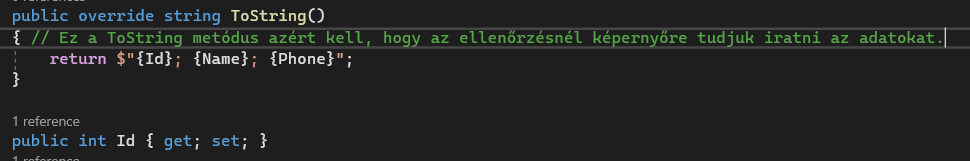


Category.cs

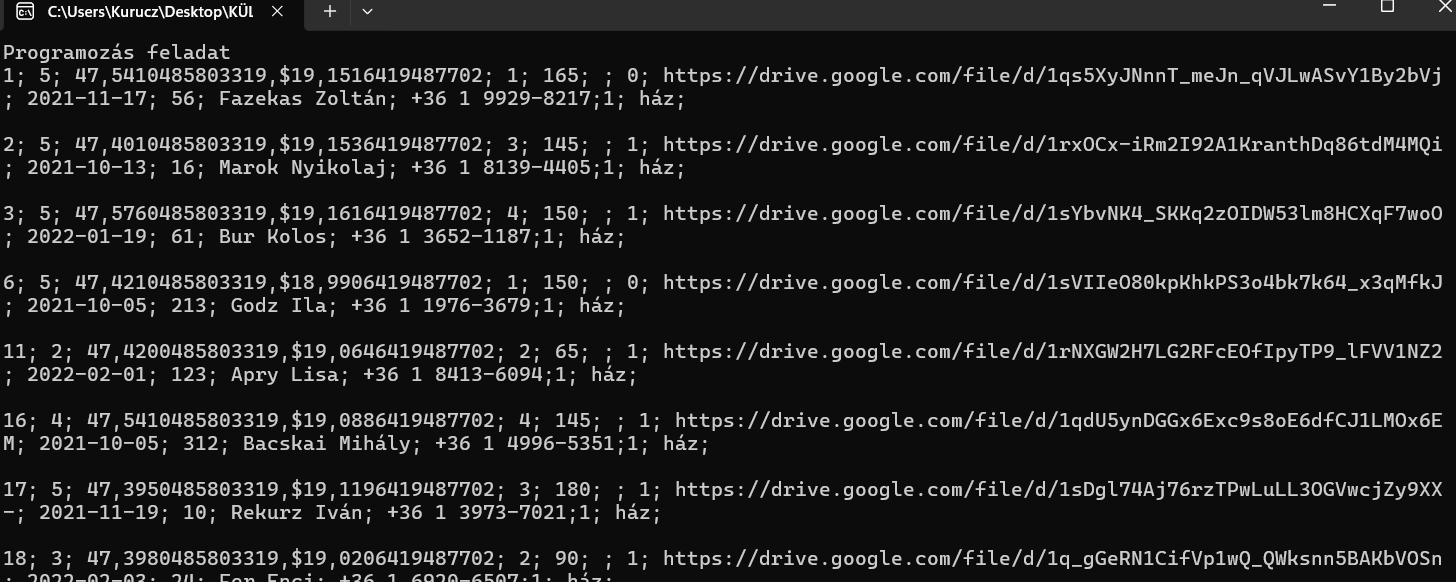
A képen szöveg, képernyőkép, Betűtípus látható

Automatikusan generált leírás

Seller.cs



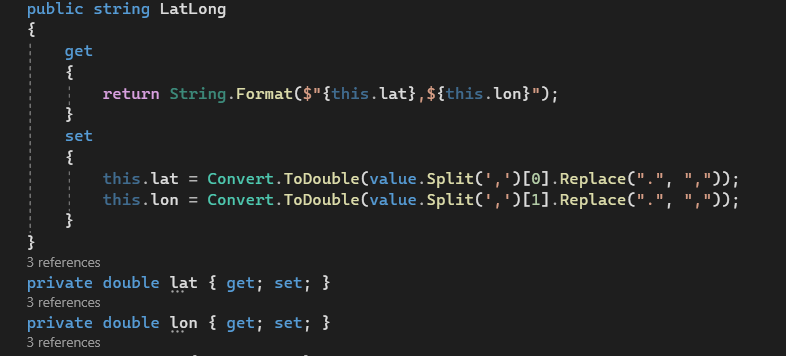
Ha eddig mindent jól csináltunk a következő a kimenet.



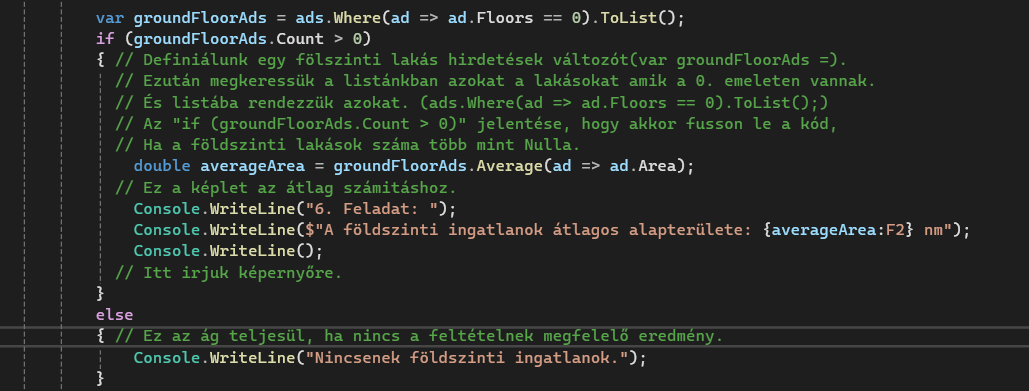
1. Feladat

Határozza meg és írja ki a minta szerint az eladásra kínált földszinti ingatlanok átlagos alapterületét! Az eredményt a minta alapján két tizedesjegy pontossággal írja ki!

Módosítjuk és hozzáadjuk a következő változókat az Ad osztályhoz.



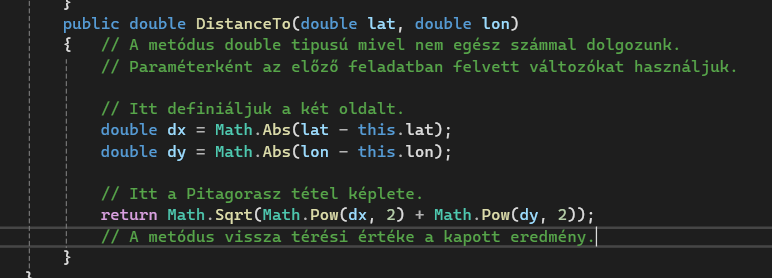
A Program.cs állományban megirjuk a függvényt hozzá.



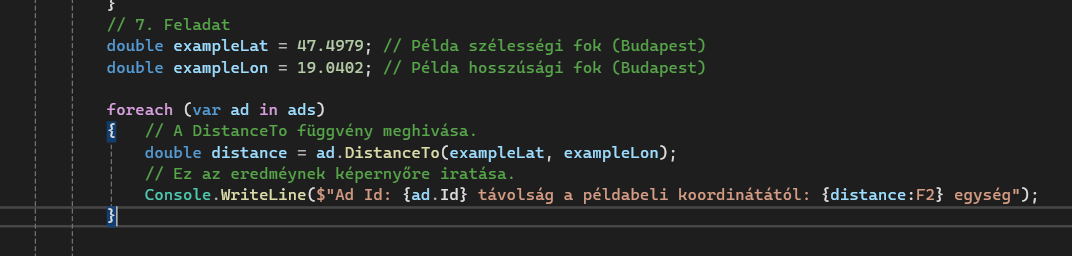
1. Feladat

Készítsen DistanceTo néven valós értékkel visszatérő metódust, amelynek segítségével adott ingatlan esetében meg lehet állapítani egy GPS koordinátától való távolságot Pitagorasz-tétel segítségével!

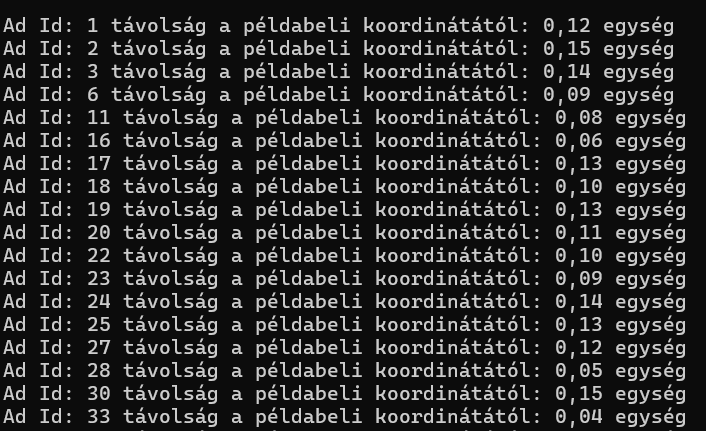
Létrehozzuk a „DistanceTo” metódust és a pitagorasz tételt beleírjuk.



Ezután a Program.cs fájlban meghívjuk és kiíratjuk ellenőrzésként.



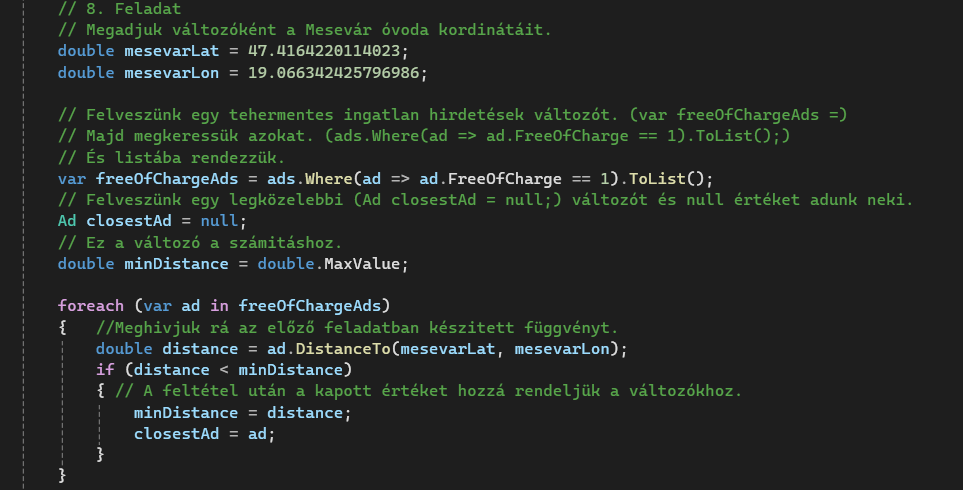
A kimenet!

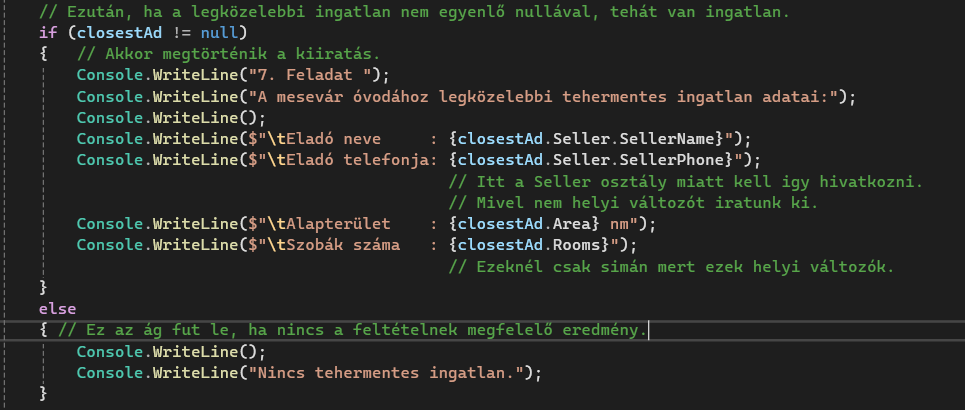


1. Feladat

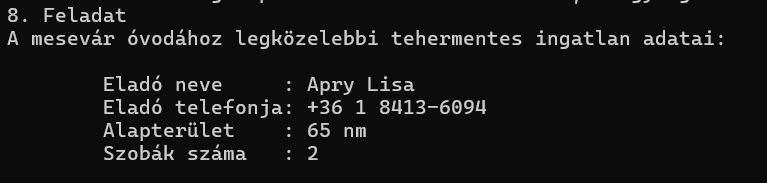
A Mesevár óvoda Budán a 47.4164220114023, 19.066342425796986 GPS koordinátán helyezkedik el. Keresse ki és írja ki a minta alapján annak a tehermentes ingatlannak az adatait, melyik a legközelebb van légvonalban a Mesevár óvodához!

A Program.cs fájlban dolgozunk.





A kimenet pedig a következő.



A teljes kimenet pedig.

