

Hozzon létre egy konzolos alkalmazást *Monogramja_Bank* néve a vizsgafelhasználó nevével ellátott meghajtó gyökré könyvtárába!

0. Osztály létrehozása

- Hozzon létre egy *BankAccount* osztályt, amely egy banki számlát reprezentál. Az osztály rendelkezzen az alábbiakkal:

1. Osztályon kívül nem látható adattagok:

- `accountHolder` – A számla tulajdonosának neve.
- `balance` – A számla aktuális egyenlege, lebegőpontos szám.
- `accountNumber` – A számla egyedi azonosítója; számlaszám, amit szöveggént tárolunk.

2. Getterek, Setterek:

- Állítson be minden adattagra egy bárhol lekérhető gettert és az osztályban lekérhető settert!

3. Konstruktorkok:

- Az első konstruktor, amely három paramétert vár: a számla tulajdonosának nevét, az egyenleget és az egyedi azonosítót.
- A második konstruktor, amely csak a tulajdonos nevét várja, és alapértelmezett értékeket ad az egyenleghez 0-t és az azonosítóhoz "".

4. Példányosítás nélkül használható adattag és metódus:

- Készítsen `accountCount` néven egy példányosítás nélkül *csak az osztályban használható* adattagot, amely a létrehozott banki számlák darabszámát tárolja!
- Készítsen `GetAccountCount` néven paraméter nélküli példányosítás nélkül használható metódust, amely *bárhol visszaadja* a létrehozott számlák számát!
- Minden példányosítás során növekedjen az `accountCount` adattag értéke eggyel!

5. Metódusok:

- Készítsen **Deposit** néven metódust, amely lehetővé teszi pénz befizetését a számlára. Az egyenleg növekszik a befizetett összeggel, amit paraméterben adunk meg!
- Készítsen **Withdraw** metódust, amely pénz felvételét teszi lehetővé a számláról. A felvett összeg levonásra kerül, *ha az elérhető egyenleg ezt lehetővé teszi*. A levont összeget paraméterben kapja meg!
- Készítsen `ToString` metódusot, ami kiírja az osztály adattagjait szöveges formátumban!

6. Ellenőrzés

- Hozzon létre egy-egy példányt a két konstruktorral a lent megjelölt minták segítségével! (Ha nem sikerült az egyiket megcsinálnia, akkor eggyel hozzon létre két példányt. Ha egyiket sem sikerült megcsinálnia, akkor készítsen szabadon két konstruktort és használja azokat!) Írassa ki a példányokat!
- Futtassa a Deposit metódust mindkét esetben két véletlen értékkel, amit [10,1000] között generál! Írassa ki a példányokat!
- Futtassa a Withdraw metódust mindkét esetben 500-as értékkel! Írassa ki a példányokat!

Minta

pelda1: „Kis Pista”, 123.478, „112233445566”

pelda2: „Nagy Rozália”

Kiíratás:

```
h:\2024-2025\12.cb AAF 2csop\-= Dolgozatok\Doga 2\Megoldas_Bank\Megoldas_Bank\bin\Debug\Megoldas_Bank.exe
Pédányosítás során:
[Meglévő ügyfelek száma: 1] Account Holder: Kis Pista, Balance: 123,48 Ft, Account Number: 112233445566
[Meglévő ügyfelek száma: 2] Account Holder: Nagy Rozália, Balance: 0,00 Ft, Account Number:

Véletlen érték növelése során:
[Meglévő ügyfelek száma: 2] Account Holder: Kis Pista, Balance: 931,48 Ft, Account Number: 112233445566
[Meglévő ügyfelek száma: 2] Account Holder: Nagy Rozália, Balance: 689,00 Ft, Account Number:

500-zal való csökkentés során:
[Meglévő ügyfelek száma: 2] Account Holder: Kis Pista, Balance: 431,48 Ft, Account Number: 112233445566
[Meglévő ügyfelek száma: 2] Account Holder: Nagy Rozália, Balance: 189,00 Ft, Account Number:
```

```
h:\2024-2025\12.cb AAF 2csop\-= Dolgozatok\Doga 2\Megoldas_Bank\Megoldas_Bank\bin\Debug\Megoldas_Bank.exe
Pédányosítás során:
[Meglévő ügyfelek száma: 1] Account Holder: Kis Pista, Balance: 123,48 Ft, Account Number: 112233445566
[Meglévő ügyfelek száma: 2] Account Holder: Nagy Rozália, Balance: 0,00 Ft, Account Number:

Véletlen érték növelése során:
[Meglévő ügyfelek száma: 2] Account Holder: Kis Pista, Balance: 340,48 Ft, Account Number: 112233445566
[Meglévő ügyfelek száma: 2] Account Holder: Nagy Rozália, Balance: 13,00 Ft, Account Number:

500-zal való csökkentés során:
[Meglévő ügyfelek száma: 2] Account Holder: Kis Pista, Balance: 340,48 Ft, Account Number: 112233445566
[Meglévő ügyfelek száma: 2] Account Holder: Nagy Rozália, Balance: 13,00 Ft, Account Number:
```