

Javasolt feladatok a Szoftvertervezés és -fejlesztés I. kiegészítő kurzus 8. heti alkalmára

Bankszámlák kezelésére alkossunk egy objektum orientált programot. Minden egyes bankszámla legyen a BankSzamla osztály egy példánya. A BankSzamla osztály az alábbi (privát láthatóságú) adattagokkal rendelkezzen: számlaszam, nev (számlatulajdonos neve), egyenleg.

Valósítsa meg az alábbi konstruktorokat, tulajdonságokat és példányszintű metódusokat. Minden metódust és tulajdonságot teszteljen a Main() metódusból.

- `BankSzamla(string számlaszam, string nev, double egyenleg)`
Három paraméteres konstruktor, amelyen keresztül kezdeti értékek adhatók az adattagoknak.
- `BankSzamla(string nev, double egyenleg)`
Kétparaméteres konstruktor, mely kezdeti értéket ad a nev és egyenleg adattagoknak. A számlaszám véletlenszerűen álljon elő. Formátuma: NNNNNNNN-NNNNNNNN, ahol N egy tetszőleges számjegyet jelöl.
- `Bankszamla(string számlaszam, string nev)`
Kétparaméteres konstruktor, mely kezdeti értéket ad a számlaszam és nev adattagoknak. Az egyenleg értéke legyen alapértelmezetten 0.0.
- `Bankszamla(string nev)`
Egyparaméteres konstruktor, mely kezdeti értéket ad a nev adattagnak. A számlaszam véletlenszerűen álljon elő, a fentebb megadott mintának megfelelően, az egyenleg pedig 0.0 legyen.
- `Bankszamla()`
Paramétermentes konstruktor. A tulajdonos neve "Anonymus", a számlaszámot és egyenleg értéket pedig az egyparaméteres konstruktornak megfelelően állítja be.
- Mindegyik adattaghoz készítsen getter és setter tulajdonságokat is. A számlaszam setterénél figyeljen arra, hogy csak a fentebb megadott formátumú stringeket lehessen az adattagba írni. A nev setterénél figyeljen arra, hogy egy névben legalább két olyan szónak kell lenni, aminek kezdőbetűje nagybetű. Amennyiben érvénytelen adat kerülne beírásra, akkor ehelyett a régi érték maradjon meg.
- `string Megjelenit()`
A metódus előállít egy stringet, aminek segítségével a számla aktuális adatai megjeleníthetők.
- `bool Fizet(int osszeg)`
Az egyenleg értéke csökken a paraméter értékével, ha elég pénz van a számlán. Ilyenkor a visszatérési érték true, ha viszont nincs elég pénz, akkor nem fizetünk és a visszatérési érték false.
- `void Novel(int osszeg)`
A számlára pénz érkezik, így az egyenleg értéke növekszik a paraméter értékével.
- `void Kamatozik(double szazalek)`
A számla egyenlege a paraméternek megfelelő százalékkal növekszik (pl. 2,5%).