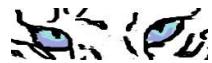


Jó alanyok

Ebben a feladatban diákok és tanárok adataival kell dolgoznod. A feladathoz elkészített interfész és az osztályok külön osztály könyvtárba (Class Library) kerüljenek! A feladat elkészítésénél törekedj az **OOP elvek** és a **clean code** szabályok betartására!

1. Készíts egy interfészt **IVizsgalat** néven, amely előírja a **JoAlanyE** metódus implementálását, amelynek nincs paramétere, és logikai értékkel tér vissza.
2. Készítsd el a **HibasEletkorException** osztályt, amely az **Exception** osztályból származik, és „Az életkor beállítása nem megfelelő!” szöveggel dob hibaüzenetet!
3. Készíts egy osztályt **Szemely** néven!
 - a. Az osztályban egy személy nevét és születési dátumát tároljuk. A személyek neve legyen kívülről lekérdezhető.
 - b. Készíts **Kor** néven tulajdonságot, amely visszaadja a személy életkorát!
 - c. Definiálj konstruktort, amellyel egy példány adatai beállíthatók! Amennyiben az életkor értéke kisebb, mint 14, akkor dobjon hibát a **HibasEletkorException** kivétel osztály használatával!
 - d. Az osztály **ToString** metódusa adja vissza a személy nevét, és zárójelben az életkorát!
4. Készíts egy **Diak** osztályt, amely a **Szemely** osztály leszármazottja, belőle viszont nem származhat további osztály. Az osztály implementálja az **IVizsgalat** interfészt!
 - a. Az osztály tartalmazzon egy adattagot, amely a puskák számát tárolja.
 - b. Definiáljon konstruktort, amellyel egy példány adatai beállíthatók!
 - c. Implementálja a **JoAlanyE** függvényt! Akkor jó alany egy Diák, ha nincs nála puska.
 - d. Az osztály **ToString** metódusa adja vissza a diák nevét, zárójelben az életkorát, valamint, hogy jó alany-e a diák!
5. Készíts egy **Tanar** osztályt, amely a **Szemely** osztály leszármazottja. Az osztály implementálja az **IVizsgalat** interfészt!
 - a. Az osztály tartalmazzon egy adattagot, amely a nála megszerzett év végi jegyek átlagát tárolja.
 - b. Definiáljon konstruktort, amellyel egy példány adatai beállíthatók!
 - c. Implementálja a **JoAlanyE** függvényt! Akkor jó alany egy Tanár, ha 30 évnél fiatalabb, és a nála megszerzett év végi jegyek átlaga legalább 3,5.
 - d. Az osztály **ToString** metódusa adja vissza a tanár nevét, zárójelben az életkorát, valamint, hogy jó alany-e a tanár!
6. Készíts egy **SzemelyFactory** osztályt, amelynek **Factory** metódusa **Szemely** típussal tér vissza! Az 1 string paraméterrel rendelkező metódus az adatfájl 1 adatsorából elkészíti az adatoknak megfelelő diák vagy tanár példányt. Az adatsor adatai pontosvessző karakterrel vannak egymástól elválasztva. A sorban az első adat egy „t” vagy egy „d” betű utal arra, hogy tanár vagy diák példányt kell létrehozni. A következő adat a személy születési dátuma. Az utolsó adat tanároknál az év végi jegyek átlaga, diákoknál pedig a puskák száma.
7. Készíts egy **Nyilvantartas** osztályt, amely egy megfelelő adatszerkezetben tárolja az iskolában tanuló diákokat és tanárokat!
 - a. Készíts az osztályhoz egy egész szám típusú indexert, amely visszaadja, hogy azzal a sorszámmal melyik diádot vagy tanát tároltuk el!
 - b. Készíts egy tulajdonságot vagy metódust, amely visszaadja az összes tárolt diádot!
 - c. Készíts egy tulajdonságot vagy metódust, amely visszaadja az összes tárolt tanárt!

Asztali alkalmazások fejlesztése



8. Készítsd el a futtatható osztályt!

- a. Az osztály nyisson meg és olvasson be egy „input.txt” nevű állományt! A fájlban diákok és tanárok adatai találhatók. Dolgozd fel a sorokat, és hozd létre az objektumokat!
- b. Az egyes objektumok `ToString` metódusának használatával jelenítsd meg a személyek adatait! Hibadobás esetén jelenítsd meg a hibaüzenetet!
- c. Egészítsd ki a futtatható programot úgy, hogy felkészíted a többi hibalehetőség kezelésére is!
- d. Jelenítsd meg, hogy hány diák és hány tanár található a nyilvántartásban!
- e. Add meg a tanárok átlagos életkorát!
- f. Csoportosítsd a diákokat puskák száma szerint, és add meg, hogy melyik csoportba mennyi diák tartozik!

9. Szorgalmi feladatként készíts az alkalmazáshoz unit teszteket!

