

Iskola szakmai vizsga gyakorló feladat

A `nevek.txt` állományban rögzítettük egy középiskola tanulóinak néhány adatát.

Feltételezheti,

hogy nincs két azonos nevű tanuló egy osztályban. Az állomány tartalma soronként:

- iskola kezdésének éve (2004-2007)
- az osztály betűjele (a-e)
- a diák neve (ékezetek nélkül).

Az adattagok pontosvesszővel vannak elválasztva. Példa (részlet) a `nevek.txt` állományra:

```
2004;d;Vavrek Kristof
2006;e;Hidas Reka
2006;d;Kun Michael
```

Megoldásában:

- A képernyőre írást igénylő részfeladatok eredményének megjelenítése előtt írja a képernyőre a feladat sorszámát (például: 3. feladat)!
 - Az egyes feladatokban a kiírásokat a minta szerint készítse el!
 - Az ékezetmentes kiírások is elfogadottak.
 - Az azonosítókat kis- és nagybetűkkel is kezdheti.
 - A program megírásakor az állományban lévő adatok helyes szerkezetét nem kell ellenőriznie, feltételezheti, hogy a rendelkezésre álló adatok a leírtaknak megfelelnek.
 - A megoldását úgy készítse el, hogy az azonos szerkezetű, de tetszőleges bemeneti adatok mellett is helyes eredményt adjon!
- Oldja meg a következő feladatokat:

1. Készítsen `Iskola` néven konzol típusú projektet, melyben megoldja a következő feladatokat!

2. Olvassa be a `nevek.txt` állományban lévő adatokat! Hozzon létre megfelelő adatszerkezetet az adatok tárolására:

-Hozzon létre osztályt a tanulóknak. Az adattagok legyenek propertyk! (pld. `tanulo.nev...`)

Ha úgy érzi bővítse az osztályt olyan adattagokkal, amelyekkel a feladat megoldásához szükségesnek érzi! Az alapadatok csak konstruktoron keresztül legyenek módosíthatóak! (public get, private set)

-Az osztálynak legyen egy vagy több konstruktora, ami segít feldolgozni az adatokat! PI megkapja a fájl 1 sorát, és beállítja a jellemzőket!

3. Írja ki a képernyőre, a tanulók adatait, illetve, hogy hány tanuló jár az iskolába!

4. Az iskolai rendszergazdának egyedi azonosítókat kell készítenie a számítógép-hálózat használatához. Az azonosítókat a következő módon alakítja ki: első karaktere az évfolyam

utolsó számjegye (pl.: 2006 esetén 6), következő karakter az osztály betűjele, majd a vezetéknév első három karaktere, végül első keresztnév első három karaktere következik. Az azonosítóban mindenütt kisbetűk szerepelnek. Feltételezhetjük, hogy a vezetéknév és az első keresztnév legalább 3 karakteres. Készítsen jellemzőt vagy függvényt, melyben meghatározza a rendelkezésre álló adatokból a tanuló azonosítóját!

Az elkészített jellemzőt/függvényt felhasználva írja ki az adatszerkezetben tárolt első és utolsó tanuló azonosítóját a minta szerint!

5. Készítsen tesztet a függvényhez!

A teszt vizsgáljon hibás és nem hibás eredményeket is!

(A teszt a jelenlegi adatok vizsgálatára szól, független a fájl tartalmától! Tehát nem kell vizsgálnia, ha megváltozik valakinek az osztálya stb...)

Jó minta: Bodnar Szilvia →6cbodszl

Krizsan Vivien Evelin →6ckriviv

Rossz minta, bármi ami biztosan nem jó, tehát részben vagy egészében hibás átalakítás!

6. Írja ki az azonosítók txt-be az összes tanuló nevét és azonosítóját!

7. Alkalmazza a tiszta kód elveit! Kommentelje a programot!

WPF-es feladat

Készítsen grafikus wpf-es alkalmazást a következő feladatok megoldására, melynek projektjét

IskolaWPF néven mentse el!

9. Az alkalmazás grafikus felhasználói felületét alakítsa ki a minta szerint! Az ablak címsorában

a „IskolaWPF” szöveget jelenítse meg!

10. Az ablakon található listába a program induláskor töltse be a `nevek.txt` állomány sorait! A lista elemei a forrásfájl egy-egy sora (továbbiakban tanulója) legyen!

Az adatok tárolásához nyugodtan használja az előző feladatban megírt osztályokat, és tárolja hasonlóan listában, vagy más alkalmas adatszerkezetben az adatokat!

11. Oldja meg, hogy a kijelölt tanuló a „Törlés” parancsgomb lenyomása után törlésre kerüljön a

listából! Ha a listában nincs kijelölt tanuló, akkor törléskor a „Nem jelölt ki tanulót!” szöveg jelenjen meg egy felugró ablakban!

12. Ha az „Állomány mentése” parancsgombra kattintunk, akkor történjen meg a listából a tanulók

mentése a `nevekNEW.txt` állományba, melynek szerkezete a forrásállomány szerinti

legyen! Ha a mentés sikeres volt, akkor a „Sikeres mentés!” felirat jelenjen meg egy felugró ablakban! Ha az állomány mentése sikertelen, akkor a hibaüzenet (a hibához tartozó beépített

üzenet/message) jelenjen meg egy felugró ablakban! Lehetséges hibaokokhoz tartozó beépített

üzeneteket a minták között talál!

Minták a grafikus alkalmazás futására:

