

Általános tudnivalók

Ebben az ismertetésben az osztályok, valamint a minimálisan szükséges metódusok leírásai fognak szerepelni. A feladatmegoldás során fontos betartani az elnevezésekre és típusokra vonatkozó megszorításokat, illetve a szövegek formázási szabályait. Segédfüggvények létrehozhatóak, a feladatban nem megkötött adattagok és elnevezéseik is a feladat megoldójára vannak bízva. Törekedjünk arra, hogy az osztályok belső reprezentációját a lehető legjobban védjük, tehát csak akkor engedjük, és csak olyan hozzáférést, amelyre a feladat felszólít, vagy amit azt osztályt használó kódrészlet megkíván!

A beadott megoldásodnak működnie kell a mellékelt tesztprogramokkal, de ez nem elégséges feltétele az elfogadásnak. A megírt forráskód legyen kellően általános és újrafelhasználható!

Használható segédanyagok: [Java dokumentáció](#), legfeljebb egy üres lap és toll. Ha bármilyen kérdés, észrevétel felmerül, azt a felügyelőknek kell jelezni, *NEM* a diáktársaknak!

A feladat összefoglaló leírása

A feladatban autóversenyt ábrázolunk és versenyzőket.

A programhoz tartozik [egységtesztelő](#) és a hozzá tartozó [tesztfájl](#), amely az egyes osztályok funkcionalitását teszteli, illetve a várható pontszámot mutatja.

A feladat részletes ismertetése

Versenyző

Valósítsuk meg a `Versenyzo` osztályt, amelynek az a feladata, hogy egy versenyzőt ábrázoljon.

- Két adattagja van: a `nev`, amely egy szöveg, és a `versenyek`, amely szövegek egy tetszőlegesen hosszú tömbje. Ez utóbbiban szerepelnek azon versenyek nevei, amelyeken a versenyző korábban elindult.
- Az osztály egyik konstruktora egyszerűen kapja meg a versenyző nevét és versenyeinek neveit, és tárolja ezeket el.
- A második konstruktor a versenyző adatait egyetlen szöveggként kapja meg: a `nev`, és a versenyek vesszőkkel vannak elválasztva benne. Ha nincsenek versenyek, akkor a szöveg nem tartalmaz vesszőt. A konstruktor bontsa szét a szöveget, és töltsse fel az adattagokat.

Segítség: használjuk a `String` osztály `split` metódusát a szétbontáshoz.

- Legyen egy osztályszintű `javaVersenyzo` adattagunk, amelyben egy versenyző legyen, akinek a neve legyen `James Gosling`, és egy versenyen vett részt: `Kanada Open`.
- Az adattagokhoz tartozzanak „getter” és „setter” műveletek. Ezek a műveletek és a konstruktorok ne szivárogtassák ki a belső állapotot: a tömb adattagok másolatait adják ki/állítsák be az adattagba, ne egyszerűen a tömb referenciáját.

- Tartalmazzon az osztály saját `toString` megvalósítást. A metódus eredménye legyen `Versenyzo[Versenyzo1, [versenyA, versenyB, versenyC, versenyD]]`, ha a versenyző neve `Versenyzo1`, a versenyei pedig `versenyA`, `versenyB`, `versenyC` és `versenyD`.
- Legyen egy `hasVerseny` függvénye, amely megkapja szövegesen egy verseny nevét, és visszatér azzal, hogy a versenyző részt vett-e ilyen nevű versenyen.

Verseny

Valósítsuk meg a `Verseny` osztályt.

- A versenynek legyen egy `nev` adattagja (szöveg) és tartsuk számon, kik a versenyzok (`Versenyzok` tömbje).
- A konstruktora egy fájlnevet kap szövegesen paraméterként. Feltételezhető, hogy a fájl létezik, és helyesen van kitöltve: az első sorában tartalmazza a verseny nevét, a másodikban a benne található versenyzők számát, a fennmaradó sorokban pedig egy-egy versenyző adatait. A versenyzők adatai olyan alakúak, ahogyan azt a `Versenyzomasodik` konstruktora elvárja. A `Verseny` konstruktora olvassa be a fájl adatait, és töltsse fel az adattagokat.
- Valósítsunk meg itt is `toString` műveletet. A példaként adott `verseny1.txt` fájlban leírt verseny adja az alábbi kimenetet:

```
"Verseny:(Verseny1, Versenyzo[Versenyzo1, [versenyA, versenyB, versenyC, versenyD]],
Versenyzo[Versenyzo2, [versenyA, versenyE]], Versenyzo[Versenyzo3, [versenyB,
versenyC, versenyF]])"
```

- Legyen egy `getVersenyVersenyzoi` művelete. Ez szövegesen megkapja egy verseny nevét, és azokat a versenyzőket adja vissza egy `List` listában, akik részt vettek ilyen nevű versenyen.

Pontozás

A tesztelő által adott pontszám csak becslésnek tekinthető, a gyakorlatvezető levonhat pontokat, vagy adhat részpontokat.

- 0 - 13: elégtelen (1)
- 14 - 19 : elégséges (2)
- 20 - 41 : közepes (3)
- 28 - 33 : jó (4)
- 34 - 41 : jeles (5)

Jó munkát!