

Programozás 2

– 1. zárthelyi dolgozat –

Czier Norbert

2022-10-13

1. feladat – Termék (1 pont)

Írjon egy Product nevű osztályt, amivel egy terméket tudunk reprezentálni. A Product osztályt egy Product.java állományba helyezze el. Az osztálynak egyetlen konstruktora legyen. A konstruktornak meg kell adni a termék nevét, mennyiségét és egységárát. A termék osztállyal van még egy azonosítója is, ami meg egyezik az aktuális példány számával. A Product osztálynak négy attribútuma van: id, name, quantity és unitPrice.

```
Product p1 = new Product("Tej", 3, 400);
Product p2 = new Product("Alma", 10, 600);
Product p3 = new Product("Körte", 12, 600);
```

Egy Product típusú példány id és name attribútumait ne lehessen módosítani.

Példányosítás után az példányokat a következőképpen akarjuk használni. A megjegyzésekben az elvárt kimenet látható:

```
System.out.println(p1);           // Termék: Tej, Mennyiség: 3, Egységár: 400ft
System.out.println(p1.getId());   // 1
System.out.println(p3.getId());   // 3
System.out.println(p3.getName()); // Körte
System.out.println(p3.setQuantity(13)); // 13
System.out.println(p3.getUnitPrice()); // 600
System.out.println(p3.getTotalPrice()); // 7800 (13*600)
System.out.println(p1.compare(p2)); // -1
System.out.println(p2.compare(p3)); // 0
System.out.println(p3.compare(p1)); // 1
```

Magyarázat: A compare() metódus két termék árát hasonlítja össze. Ha a bal oldali terméknek kisebb az egység ára, akkor -1-et adunk vissza. Ha a két terméknek azonos az egység ára, akkor 0 a visszatérési érték. Ha a bal oldali terméknek nagyobb az egység ára, akkor +1-et kell visszaadni.

2. feladat – számológép (1 pont)

Írjon egy programot, ami parancssori argumentumként vár két egész számot és egy operátort. A program végezze el az adott műveletet a két számon.

A következő operátorokat lehet használni: +, -, x, /. (A '*' helyett a 'x' lesz a szorzás szimbóluma).

A program három parancssori argumentumot vár. Ha eltér az argumentumok száma, akkor informatív módon értesítsük a felhasználót.

Ha megkaptuk a három argumentumot, akkor feltételezzük, hogy a sorrend: szám, operátor, szám. Ezt nem kell ellenőrizni.

Futási példák:

```
$ java Main
Hiba! Adj meg pontosan három argumentumot!
$ echo $?
1
```

```
$ java Main 23
Hiba! Adj meg pontosan három argumentumot!
$ echo $?
1
```

```
$ java Main 23 x 23 2
Hiba! Adj meg pontosan három argumentumot!
$ echo $?
1
```

```
$ java Main 30 x 2
60
```

```
$ java Main 12 + a
Hiba! Nem érvényes szám.
```

```
$ java Main 12 - 2
10
```

3. feladat – helyettesítés (1 pont)

Írjon egy programot, ami interaktív módon bekér a felhasználótól egy szöveget és minden olyan karaktert, ami nem betű, azt egy '_' jellel helyettesíti.

Futási példák:

```
$ java Main
Adjon meg egy szöveget: Ez az 1.ZH
Az átalakított szöveg:  Ez_az___ZH
```

```
$ java Main
Adjon meg egy szöveget: Remelem 5-os lesz a ZH
Az átalakított szöveg:  Remelem___os_lesz_a_ZH
```

4. feladat – háromszög (1 pont)

A `haromszog.txt` file-ban háromszögek oldalhosszai szerepelnek szóközzel elválasztva. Írjon egy programot, ami megmondja, hogy a file-ban hány darab érvényes háromszög szerepel.

Egy érvényes háromszögben bármelyik két oldal összege nagyobb kell, hogy legyen, mint a fennmaradó oldal.

Például tegyük fel, hogy az állomány a következőket tartalmazza:

```
5 25 10
3 4 5
```

Az 5, 25, 10 oldalhosszúságú háromszög nem érvényes, mert $5 + 10$ nem nagyobb, mint 25. A 3, 4, 5 oldalhosszúságú háromszög érvényes, mert $3 + 4 > 5$, $3 + 5 > 4$, $4 + 5 > 3$.

Futási példa egy kisebb méretű input fájl esetén:

```
$ cat pelda.txt
5 25 10
3 4 5

$ java Main
1
```