

Tartalomjegyzék

- 1 Előadások
- 2 Interface gyakorló feladatok
 - ♦ 2.1 Síkidomok
 - ♦ 2.2 Repülőgépek
- 3 Hasznos feladat

Előadások

Nézzük át először ezeket az előadásokat:

- A memóriaképről
- Öröklődés és interface-ek

Interface gyakorló feladatok

Síkidomok

Készíts egy **Mesurable** interface-t, aminek az egyetlen metódusa a **weight**, ami **double**-ben az alakzat területét adja vissza.

Készítsd el a Teglalap, Kor, Haromszog osztályokat, úgy hogy mindegyik impementálja ezt az interface-t. Majd írd egy hasonló tesztet rá, mint amilyen a diákon is volt.

Repülőgépek

Két interface: **Passanger**, **Freighter**, az első az utasszállító, ami azt tudja, hogy visszaadja az utasok számát, a második a csomagszállító, ami visszaadja a csomagok súlyát.

Legyen 3 féle repülőgép, egy-egy amelyik csak **Passanger** vagy **Freighter**-t implementál, a 3. pedig mindkettőt.

Írjátok meg értelmesen ezeket az osztályokat (utasszállítóban adattag az utasok száma, csomagszállítóban a csomagok súlya pl). Majd teszteljétek a korábbi módon.

Hasznos feladat

- Írjátok át a felhasználórendszert, hogy tömbök helyett **ArrayList**eket használjon.
- Írjátok egy **PersonalInfo** osztály, ami Stringhez Stringet rendelő **HashMap**ben tárolja a User-ek adatait.

```
HashMap<String, String> hm = new HashMap<String, String>();
```

- Minden User-nek legyen egy ilyen **PersonalInfo**-ja.
- Legyen egy statikus **HashSet** adattagja a **PersonalInfo** osztálynak, amiben tárolva vannak a lehetséges tulajdonságok (a **HashMap** lehetséges kulcsai).
- Úgy működjön az utóbb említett **HashSet**, hogy amikor új tulajdonságot adunk hozzá egy emberhez, akkor automatikusan rakja bele a **HashSet**be a tulajdonságot, ha még nem tartalmazta eddig.

- Ez azt is jelenti, hogy a tulajdonság hozzáadása a **PersonalInfo**hoz (és ezáltal egy **User**hez) egy metódussal működjön amiben csak a kulcs és értéket kell megadni.