

## Tartalomjegyzék

- 1 Mindenek előtt
- 2 1. Asteroid
- 3 2. SpaceShip
- 4 3. További kiegészítések

### Mindenek előtt

Hozzatok létre egy új projectet **Asteroids** néven. Hozzatok benne létre egy **asteroids** package-et. Majd töltsétek le innen a hiányos java file-okat.

### 1. Asteroid

Egészítsétek ki az **Asteroid** osztályt, mely reprezentálni fogja a játékban az aszteroidákat. A következő (privát) adattagjai vannak:

- position: használja a korábban megírt **Position**-t
- size: egész szám

A következő metódusokat kellene megírni hozzá:

- **konstruktorok**
  - ◆ default
  - ◆ **Position** és **int**-ből
  - ◆ két **float** és egy **int**-ből

Ha a konstruktorokat megírtátok az a jutalmatok, hogy elindul a játék, ha az **AsteroidsGame** osztályt megnyitjátok és futtatjátok. Ahhoz, hogy történjen is valami, meg kell írni még a további metódusot:

- **move:**
  - ◆ nem ad vissza semmit
  - ◆ kap két **float**-ot **x** és **y**-t
  - ◆ eltolja az aszteroida pozícióját az (x, y) vektorral.

### 2. SpaceShip

Egészítsétek ki a **SpaceShip** osztályt, mely a játékos űrhajóját reprezentálja. A következő (privát) adattagjai vannak:

- position: használja a korábban megírt **Position**-t
- rotation: **float** ami tárolja fokokban, hogy mennyire van elforgatva a hajó

A konstruktor már meg van írva, a következő metódusokat kellene megírni, ha ezek megvannak akkor a billentyűzet nyilaival lehet irányítani a hajót, space-el lehet löni:

- **move:**
  - ◆ nem ad vissza semmit
  - ◆ kap két **float**-ot **x** és **y**-t
  - ◆ eltolja az űrhajó pozícióját az (x, y) vektorral.
- **turn:**

- ◆ nem ad vissza semmit
- ◆ kap egy **float**-ot
- ◆ a kapott **float**-al elforgatja a hajót (az eddigi rotációhoz hozzáveszi a kapottat), vigyázva arra, hogy 360 fok föle sose menjen (ekkor ugorjon vissza 0 fokra).
- **accelerate**:
  - ◆ nem ad vissza semmit
  - ◆ kap egy **float**-ot
  - ◆ a kapott **float** hosszú, **rotation** irányú vektorral eltolja a hajót, használva a  $\text{Math.cos}(\text{Math.toRadians}())$  és  $\text{Math.sin}(\text{Math.toRadians}())$  függvényeket.

### 3. További kiegészítések

Írjátok meg az **Asteroid** utolsó hiányzó metódusát:

- **destroy**:
  - ◆ egy aszteroida tömböt ad vissza
  - ◆ nem kap semmit
  - ◆ az előadáson látotthoz hasonló aszteroida szétesést valósítja meg, azaz létrehoz annyi új eggyel kisebb aszteroidát amekkora a mérete és visszaadja ezek tömbjét. Igaz ez nem a legjobb megoldás, de most legyenek mind az eredeti aszteroida pozícióján.
  - ◆ Ne zavarjon meg, hogy **AsteroidExtended**-et ad vissza, ezt kezeljétek úgy, mintha simán **Asteroid** lenne.

Randomizálhatjátok az újonnan létrejövő aszteroidák számát / méretét a **Random** osztály segítségével, egy példa:

```
Random rand = new Random();  
int randomInteger = rand.nextInt();  
float randomFloat = rand.nextFloat();
```

Ha mindennel megvagytok és nem találtok már szépíteni valót a kódon nézzetek bele először hogyan működik a **SpaceShipExtended** osztály, ez talán a legolvashatóbb, majd az **AsteroidExtended**, stb.

Ha igazán menők vagyotok, akkor próbáljátok meg elérni, hogy az aszteroidák folyamatosan forogjanak, ehhez a kód legjelentősebb részét az **AsteroidExtended draw** metódusában és a **SpaceShipExtended draw** metódusában találjátok, hasonlítsátok össze ezt a kettőt.