



November 18

Szoftverfejlesztés gyakorlat - 12. óra

2019

Számonkérés

Feladatlap

Tartalom

A csoport	2
B csoport	4

A csoport

Írj egy **EroErzekeny** interfészt, ami egy **legyozie** metódust tartalmaz. A metódus paramétere egy **EroErzekeny** objektum, és egy logikai értékkel tér vissza. Legyen egy **mekkoraAzEreje** metódusa is, ami nem kér paramétert, és az EroErzekeny erejét fogja visszaadni.

Írj **Sith** interfészt, ami egy visszatérés nélküli **engeddElAHaragod()** metódust tartalmaz

Írj egy **Jedi** absztrakt osztályt, ami implementálja az EroErzekeny interfészt.

- Az osztály a következő private láthatóságú adattagokkal rendelkezik: egy lebegőpontos **erő**, és egy logikai **atallhatE** (átállhat -e a sötét oldalra).
- Az osztály rendelkezzen paraméteres konstruktorral, ami beállítja az adattagokat.
- Legyen egy public **megteremtiAzEgyensulyt** absztrakt metódusa, ami egy logikai értékkel tér vissza.
- Valósítsd meg továbbá az interfész metódusait. Az ero lekérdezésekor add vissza a Jedi erejét. Egy Jedi egy másik Jedi objektumot akkor tud legyőzni, ha az átállhat a sötét oldalra, és ereje kisebb, mint az övé. Az Uralkodo objektumot csak akkor tudja legyőzni, ha ereje kétszer nagyobb.
- Az osztálynak legyen továbbá getter és setter metódusa az adattagjaihoz, és legyen szöveges formára alakítható, kiírva az adattagok értékét.

Írj egy **AnakinSkywalker** osztályt, ami a **Jedi** leszármazottja, és megvalósítja a **Sith** interfészt.

- Az osztálynak egy default konstruktora legyen, ami beállítja Anakin tulajdonságait. Anakin ereje 150, és átállhat a sötét oldalra.
- Ha Anakin elengedi a haragját, akkor ereje egy 0-10 közötti véletlenszerű lebegőpontos számmal nő.
- Anakin akkor teremti meg az egyensúlyt az erőben, ha ereje nagyobb, mint 1000.
- Az osztály legyen továbbá szöveges formára alakítható. Az adattagok értékein kívül írja ki azt is, hogy Anakin Skywalkerrol van szó.

Írj egy **Uralkodo** osztályt, ami implementálja az **EroErzekeny** és **Sith** interfészeket.

- Az osztálynak legyen egy lebegőpontos **gonoszsag** adattagja.
- Az osztály rendelkezzen egy default konstruktorral, ami 100-ra állítja az adattag értékét.
- A metódusai az alábbiak szerint legyenek megvalósítva: az Uralkodo ereje a gonoszságának kétszeresével egyezik meg, és bármilyen EroErzekeny-t képes legyőzni, akinek ereje kisebb, mint az Uralkodó gonoszsága. Ha az Uralkodo elengedi a haragját, akkor gonoszsága 50-el nő.
- Az osztály legyen szöveges formára alakítható, ami kiírja, hogy az Uralkodóról van szó, és megadja a gonoszságát.

Írj egy **StarWars** nevű futtatható osztályt.

- Az osztály rendelkezzen egy **szereplok** statikus függvénnyel, ami egy fájl elérési útját várja paraméterül, visszatérése pedig void. A metódus feladata, hogy a fájlból beolvasott sorokat feldolgozza, és létrehozjon belőlük AnakinSkywalker, vagy Uralkodo objektumpéldányokat, majd ezekre meghívja az engeddElAHaragod metódust annyiszor, ahányszor az aktuális sor írja. Ezeket egy közös kollekcióban tárold le.
- Készíts továbbá egy **sithek** statikus metódust, ami végigmegy a tárolóban tárolt objektumokon, és kiírja őket.

- Hívd meg a main függvényben sorban a fenti két metódust. Minden esetleges kivételt (foleg: IOException) kezelj le vagy kivétel specifikációval, vagy try blokkban!

Egy minta fájl felépítése az alábbi:

Anakin 5

Uralkodo 8

B csoport

Írj egy **Urhajo** interfészt, ami egy **legyorsuljaE** metódust tartalmaz. A metódus paramétere egy Urhajo objektum, és egy logikai értékkel tér vissza. Legyen egy **milyenGyors** metódusa is, ami nem kér paramétert, és az Urhajo gyorsaságát fogja visszaadni.

Írj **Hiperhajtomu** interfészt, ami egy visszatérés nélküli **hiperUgras()** metódust tartalmaz

Írj egy **LazadoGep** absztrakt osztályt, ami implementálja az Urhajo interfészt.

- Az osztály a következő private láthatóságú adattagokkal rendelkezik: egy lebegőpontos **sebesseg**, és egy logikai **meghibasodhatE**.
- Az osztály rendelkezzen paraméteres konstruktorral, ami beállítja az adattagokat.
- Legyen egy public **elkapjaAVonosugar** absztrakt metódusa, ami egy logikai értékkel tér vissza. Valósítsd meg továbbá az interfész metódusait. A gyorsaság lekérdezésekor add vissza a LazadoGep sebességét. Egy LazadoGep egy másik LazadoGep objektumot akkor tud legyorsulni, ha az meghibásodhat, és a gyorsasága kisebb, mint az övé.
- A MilleniumFalcon objektumot csak akkor tudja legyozni, ha gyorsasága kétszer nagyobb.
- Az osztálynak legyen továbbá getter és setter metódusa az adattagjaihoz, és legyen szöveges formára alakítható, kiírva az adattagok értékét.

Írj egy **XWing** osztályt, ami a LazadoGep leszármazottja, és megvalósítja az Hiperhajtomu interfészt.

- Az osztálynak egy default konstruktora legyen, ami beállítja az X-Wing tulajdonságait. Az X-Wing sebessége 150, és meghibásodhat.
- Ha az X-Wing hiperugrást végez, akkor sebessége egy 0-100 közötti véletlenszerű lebegőpontos számmal nő.
- Az X-Wing-et akkor kapja el a vonósugár, ha meghibásodhat, és sebessége kisebb, mint 10000.
- Az osztály legyen továbbá szöveges formára alakítható. Az adattagok értékein kívül írja ki azt is, hogy egy X-Wingről van szó.

Írj egy **MilleniumFalcon** osztályt, ami implementálja az Urhajo és Hiperhajtomu interfészeket.

- Az osztálynak legyen egy lebegőpontos **tapasztalat** adattagja.
- Az osztály rendelkezzen egy default konstruktorral, ami 100-ra állítja az adattag értékét.
- A metódusai az alábbiak szerint legyenek megvalósítva: a Millenium Falcon gyorsasága a tapasztalatának kétszeresével egyezik meg, és bármilyen Urhajo-t képes legyorsulni, akinek gyorsasága kisebb, mint a Falcon gyorsasága. Ha a Millenium Falcon hiperugrást végez, akkor tapasztalata 500-al nő.
- Az osztály legyen szöveges formára alakítható, ami kiírja, hogy a Millenium Falconról van szó, és megadja a tapasztalatát.

Írj egy **StarWars** nevű futtatható osztályt.

- Az osztály rendelkezzen egy **urhajok** statikus függvényvel, ami egy fájl elérési útját várja paraméterül, visszatérése pedig void. A metódus feladata, hogy a fájlból beolvasott sorokat feldolgozza, és létrehozzon belőlük XWing, vagy MilleniumFalcon objektumpéldányokat, majd ezekre meghívja a hiperUgras metódust annyiszor, ahányszor az aktuális sor írja. Ezeket egy közös kollekcióban tárold le.

- Készíts továbbá egy **hangar** statikus metódust, ami végigmegy a tárolóban tárolt objektumokon, és kiírja őket.
- Hívd meg a main függvényben sorban a fenti két metódust. Minden esetleges kivételt (foleg: IOException) kezelj le vagy kivétel specifikációval, vagy try blokkban!

Egy minta fájl felépítése az alábbi:

XWing 4

MilleniumFalcon 18