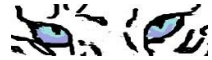


## Mesterek

Ebben a feladatban mesterek munkavállalását kell modellezned. A feladathoz elkészített interfész és az osztályok külön osztály könyvtárba (Class Library) kerüljenek! A feladat elkészítésénél törekedj az **OOP elvek** és a **clean code** szabályok betartására!

1. Készíts egy interfészt **IFoglalhato** néven, amely egy metódust és egy tulajdonságot deklarál. A FoglalhatoNapok metódus nem vár paramétert, és egy egész típusú IEnumerable sorozattal tér vissza. A SzabadnapokSzama csak olvasható tulajdonság.
2. Készítsd el a **TulSokElfoglaltsagException** osztályt, amely az Exception osztályból származik, és „*A mester túl sok munkát vállalt!*” szöveggel dob hibaüzenetet!
3. Írj egy osztályt **MesterEmber** néven, mely implementálja az IFoglalhato interfészt!
  - a. Az osztálynak 3 adata legyen: szöveg típusú adatként tároljuk a mester nevét, egy egész típusú értékben a mester napidíját, egy logikai típusú 31 elemű tömbben a hónap napjainak elfoglaltságát, ahol igaz érték jelzi, hogy a nap szabad, hamis érték, ha foglalt.
  - b. Az osztályhoz paraméteres konstruktor is tartozik, mely 2 paramétert (név, napidíj) vár, és inicializálja az adatokat. Kezdetben minden nap szabad a mesterember.
  - c. A FoglalhatoNapok metódus implementációja adja vissza a szabadnapokat!
  - d. A SzabadnapokSzama tulajdonság adja vissza a hónapban a szabadnapok számát!
  - e. Az osztály ToString metódusa adja vissza szöveggént a mester nevét, napidíját, valamint a szabadnapokat!
  - f. Készíts egy indexert, amely egy nap sorszáma visszaadja logikai értéként, hogy a nap szabad-e vagy foglalt! A napok sorszámozása 1-gyel kezdődik, és 31-ig tart. Rossz index értékekre dobj megfelelő kivételt!
  - g. Készíts MunkatVallal néven egy logikai visszatérési értékkel, és egész típusú paraméterrel rendelkező metódust! A metódus nem tartalmaz megvalósítást!
4. Készíts egy **Burkolo** osztályt, amely a MesterEmber osztály leszármazottja, belőle viszont nem származhat további osztály.
  - a. Az osztály tárolja a szakterület típusát, melynek értéke csak „belső” vagy „külső” lehet, és az adat legyen példány szinten lekérdezhető!
  - b. Az osztály konstruktora kapjon meg minden szükséges adatot paraméterként! Kezdetben minden nap szabad.
  - c. Az osztály ToString metódusa adja vissza szöveggént a mester nevét, a szakterület típusát, a napidíját, valamint a szabadnapokat!
  - d. A MunkatVallal metódus paraméterként kapott értéke a hónap adott számú napját tartalmazza. Ellenőrizd, hogy a burkoló szabad-e még azon a napon. Ha nem szabad, akkor a visszatérési érték legyen hamis. Amennyiben a mesternek 10-nél kevesebb szabadnapja van, akkor dobj TulSokElfoglaltsagException kivételt! Különben foglald le a napot, és a visszatérési érték legyen igaz.
5. Készíts egy **VizvezetekSzerelo** osztályt, amely a MesterEmber osztály leszármazottja, belőle viszont nem származhat további osztály.
  - a. Az osztály tárolja egész típusként, hogy a mester hány év tapasztalattal rendelkezik. Az adat legyen példány szinten lekérdezhető!
  - b. Az osztály konstruktora kapja meg a mester nevét és a tapasztalati évek számát paraméterként! A napidíj legyen a tapasztalati évek száma \* 6000 Ft. Kezdetben minden nap szabad.
  - c. Az osztály ToString metódusa adja vissza szöveggént a mester nevét, tapasztalati évét, a napidíját, valamint a szabadnapokat!



- d. A `MunkatVallal` metódus paraméterként kapott értéke a hónap adott számú napját tartalmazza. A vízvezetékszerelőnek 3 napra van szüksége, hogy a feladatát elvégezze. Egy napra a megadott érték előtt, és egy napra a megadott érték után. Ellenőrizd le, hogy a mester szabad-e még a szükséges napokon, ha igen, foglald le a napokat, és a visszatérési érték legyen igaz. Ha nem szabad, akkor a visszatérési érték legyen hamis. Ügyelj a hónap elejére és a végére!
  - e. Az előző feladatban meghatározottak szerint valósítsd meg a `FoglalhatoNapok` metódust és a `SzabadnapokSzama` tulajdonságot!
6. Készíts egy **MesterFactory** osztályt, amelynek `Factory` metódusa `Mester` típussal tér vissza!
- a. Az 1 string paraméterrel rendelkező metódus az adatfájl 1 adatsorából elkészíti az adatoknak megfelelő mester példányt. Az adatsor adatai pontosvessző karakterrel vannak egymástól elválasztva. A sorban az első adat a mester típusa („b” – burkoló, „v” – vízvezetékszerelő), a 2. adat a mester neve. Burkolóknál a 3. adat a napidíj, a 4. adat pedig a szakterület. Vízvezetékszerelőknél a 3. adat a tapasztalati évek száma.
7. Készítsd el a futtatható osztályt!
- a. Az osztály nyisson meg és olvasson be egy „input.txt” nevű állományt! Dolgozd fel a sorokat, és hozd létre a mesterember objektumokat!
  - b. Szimulálj minden mesterembernek megrendeléseket! A burkolóknak 25, a vízvezetékszerelőknak 10 megrendelése legyen. A megrendelések véletlenszerű napokra történjenek! Az egyes mesterembereknek leadott megrendeléseket és azok eredményét írd ki a `megrendelesek.txt` állományba!
  - c. Egészítsd ki a futtatható programodat úgy, hogy felkészíted a többi hibalehetőség kezelésére is!