|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| NÉV: |  | NEPTUN KÓD: |  |  |  |  |  |  | **X** CSOPORT |
| **!** | A beadandó solution elnevezési mintája: NEPTUNKOD\_X, ahol a neptunkód a saját neptunkódot jelöli. | | | | | | | | |
| A beadott dolgozat fordítási hiba esetén nem értékelhető. Kommentezett részre pont nem szerezhető. | | | | | | | | |
| A feltöltendő file-t a megfelelő kurzushoz töltse fel, egyéb esetben a ZH nem értékelhető. | | | | | | | | |
| Minden részfeladat elvégzését, a megadott nevű külön metódussal, illetve osztállyal valósítsa meg. | | | | | | | | |

**Feladat 0. 1 pont**

Hozzon létre egy dll generálására alkalmas projektet ConsoleLoggerLibrary néven. Hozzon létre benne egy osztály ConsoleLogger néven, benne egy void ConsoleLog metódust, amely egy object típust fogad. Az object-et írja ki a konzolra, a ToString metódus segítségével. Állítsa elő a dll állományt, majd azt adja hozzá a másik projekthez.

**Feladat 1. 0 pont**

Készítsen egy IWorker interfészt, ami az alábbi tulajdonságokat írja elő:

*string Name { get; set; } string Dept { get; set; } string Rank { get; set; } string Phone { get; set; } string Room { get; set; }*

Készítsen egy Worker osztályt, amely valósítsa meg az IWorker interfészt. Egészítse ki ezt az osztályt egy string Email {get; set;} tulajdonsággal.

**Feladat 2. 1 pont**

Készítsen egy EmailValidatorAttribute osztályt, amely rendelkezzen egy char Character és egy int Length tulajdonsággal. Az osztályra tegyen megszorítást, hogy csak tulajdonságokra lehessen alkalmazni. Az előzőekben létrehozott (Worker) email tulajdonságra alkalmazza az attribútumot, értéknek adja meg a @ karaktert, valamint az 5 értéket, mint hosszt.

**Feladat 3. 3 pont**

Készítsen egy Validator osztályt, amelyben egy bool CheckEmail metódus segítségével vizsgálja meg, hogy a paraméternek kapott object rendelkezik-e Email tulajdonsággal, s amennyiben igen, úgy vizsgálja meg, hogy az attribútumban megadottaknak eleget tesz-e az értéke. Ha igen, igaz értékkel térjen vissza, egyéb esetben hamissal. A feladat elvégzését reflexióval valósítsa meg.

**Feladat 4. 5 pont**

Készítsen egy Detector osztályt, benne egy void DetectWorkerClasses metódussal. A metódus futásidőben vizsgálja meg reflexió segítségével az aktuális osztályokat, ezek nevét kérje le fordított ABC sorrendbe rendezve egy tömbbe. Figyeljen, hogy csak azokat az osztályokat kérje le, amelyek az IWorker interfészt megvalósítják. A látványosabb teszteléshez készítsen a Worker osztályból három darab leszármazottat (FirstFloorWorker, SecondFloorWorker, ThirdFloorWorker). Ezekben további dolgok nem lesznek elhelyezve. A lekért típusokat írja ki XML fájlba (workerClasses.xml néven) figyelve az XML struktúra betartására. Írja ki az osztályokat nevét és a nevek hashkódját. A gyökérben attribútumként helyezze el, hogy hány osztály van.

**Feladat 5. 5 pont**

Hozzon létre egy Func delegáltat, amely egy fájl nevet kap bemenetnek (string) és egy IEnumerable<Worker> típussal tér vissza. A delegáltba hozzon létre egy névtelen függvényt, amelyben a kapott fájlt (workers.xml) beolvassa és egy List-et állít elő. Elegendő csak az email címeket kiválasztani a Worker objektumok előállításakor. Ezt követően hívja meg a delegáltat és az előállt kimenetet validálja le email címek alapján. Az eredményt a dll-ben kapott ConsoleLogger segítségével írassa ki.

**Feladat 6. 5 pont**

Olvassa be a workers.xml állományt és hajtsa végre rajta a következő lekérdezéseket:

* 6.1. kérdezze le a tamásokat
* 6.2. kérdezze le, hogy az egyes intézetekben hányan dolgoznak, majd rendezze ezeket darabszám alapján csökkenő sorrendbe (a kimenet egy új névtelen osztályban legyen DEPT és COUNT mezőkkel)
* 6.3. kérdezze le a dolgozók neveit és email címeit, akik a BE épület 3. szintjén dolgoznak
* 6.4. kérdezze le, hogy átlagosan mennyi a kereset az egyes intézetekben (a kimenet egy új névtelen osztályban legyen DEPT és AVGSAL mezőkkel)

**Feladat 7. 1 pont**

Az egyes részfeladatokat a Main részből tesztelje.

*- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -   
A feladat megoldása közben alkalmazza a tanult elveket és szintaktikákat.  
A feladat megoldására 90 perc áll rendelkezésre.  
A kész feladatot* ***teljes egészében*** *(egész Solution mappa), .zip (nem .rar / .7zip / stb.) formába tömörítse be, majd az ismertetett ZH leadó felületen, a megfelelő kurzust kiválasztva töltse fel!*