

3. ÓRA

Feladat: (DLL, Reflection, Attributes)

DLL:

- Hozzunk létre egy Class Library Projektet **PersonClasses** néven.
- Készítsünk egy **IPerson** nevű interfészt, majd egy **Person** abstract osztályt, amely megvalósítja az **IPerson** interfészt.
- A **Person** osztály rendelkezzen **Name(string)**, **Age(int)** tulajdonságokkal, ezek értékét konstruktorral állítsuk be. Legyen egy abstract, **string** visszatérésű **Description** metódusa, valamint írjuk felül a **ToString()** metódusát.
- Készítsünk 3 osztályt (**Worker**, **Student**, **Retired**), amelyek mint megvalósítják a **Person** abstract osztályt.

Worker osztály:

- Legyen egy **Job(string)** és egy **Salary(int)** tulajdonsága, valamint írja felül a **Description** metódust.

Student osztály:

- Legyen egy **NeptunID(string)**, egy **Email(string)** és egy **Credits(int)** tulajdonsága, valamint írja felül a **Description** metódust.

Retired osztály:

- Legyen egy **SulejmanEpisode(int)** tulajdonsága, valamint írja felül a **Description** metódust.

Program osztály:

- Hozzunk létre egy **IPerson** objektumokat tartalmazó listát.
- Készítsünk egy **Type** visszatérési értékű **Func** delegált lambda függvénnyel, amely visszaad egy random **Person** osztályból leszármazott típust.
- Töltsük fel a listát tetszőleges számú **Person** objektummal, ehhez használjuk az előző feladatban készített **Func** delegáltat. Valamint generáljunk véletlenszerűen nevet és életkort az objektumoknak.
- Írjuk ki a konzolra, ha hozzáadtunk egy elemet a listához, használjuk a **ToString** metódust.
- Írassuk ki a konzolra a lista összes elemeinek típusát, adattagjait, tulajdonságait és metódusait.
- Hívjuk meg a lista összes elemének a **Description** metódusát és írassuk ki az eredményt a konzolra.

Egészítsük ki a **DLL** fájlunkat egy **NameCheckerAttribute** osztállyal.

- Az attribútumot csak tulajdonságokon lehessen alkalmazni. Legyen egy **Length(int)** tulajdonsága, amely konstruktorból kapjon értéket.
- Adjuk hozzá az attribútumot a **Person** osztály **Name** tulajdonságához, és állítsunk be egy tetszőleges értéket.
- A **Program** osztályban készítsünk egy **NameValidator** nevű metódust, aminek egyetlen **object** bemeneti paraméterrel rendelkezik.
- Kérjük le a **Name** tulajdonságát az objektumnak, majd ezután nézzük meg, hogy van-e **NameCheckerAttribute** attribútuma. Ha van, akkor ellenőrizzük le, hogy megfelelő-e a név és írassuk ki az eredményt a konzolra.
- Végül hívjuk meg a **NameValidator** metódust a lista összes elemére.

Egészítsük ki a **DLL** fájlunkat egy statikus **ClassDetector** osztállyal.

- Az osztály tartalmazzon egy **void** statikus **ClassesToXML** metódust.
- A metódus gyűjtse ki egy **Type** tömbbe az összes olyan osztályt, amely megvalósítja az **IPerson** interfészt. Majd a tömb elemeit rendezzük név szerint növekvő sorrendbe.
- A tömb elemeit írassuk ki egy **classes.xml** fájlba. Az **XML** tartalmazza az osztály nevét és a HashCode-ját.
- Végül hívjuk meg a **Program** osztályban a **ClassesToXML** metódust.