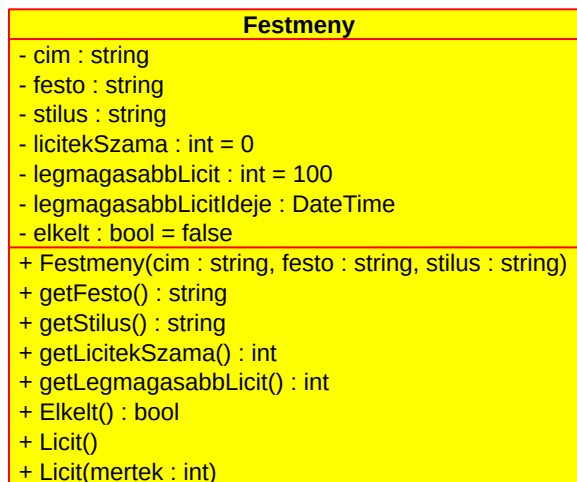


# 1. Feladat

- a) Hozz létre egy új konzol alkalmazást **AukcioProjekt** névvel.  
b) Készíts egy osztályt **Festmeny** névvel, az alábbi UML diagram alapján.



Az osztály példányosításakor állítsa be a licitek számát és a legmagasabb licitet 0-ra, és az elkelt változót hamisra.

c.) A **Licit()** eljárás:

- Ha a festmény már elkelt akkor írja ki hibaüzenetet a konzolra és ne történjen semmi.
- Ha még nem volt licit akkor a kezdeti licit értékével (100\$) licitáljon, majd növelje a licitek számát eggyel, és állítsa be a legutolsó licit idejét az aktuális időpontra.
- Ha már volt licit, akkor állítsa be a legmagasabb licitet 10%-al nagyobbra, növelje a licitek számát eggyel, majd állítsa be a legutolsó licit idejét az aktuális időpontra.
- A legmagasabb licitet úgy állítsa be, hogy az első 2 számjegy után csak 0-k legyenek.  
Pl.: ha 121 lenne a licit akkor 120 legyen, 1210 helyett 1200.

A **Licit(merek:int)** eljárás működése hasonló legyen csak 10% helyett a megadott %-kal növelje a licitet. A paraméter csak 10 és 100 közötti szám legyen. Hibás paraméter esetén, konzolra hibaüzenetet írjon ki, és ne történjen licit.

Ha már volt licit, akkor a licit eljárás, ezt a metódust hívja meg 10%-os mértékkel.

d.) Bővítsd ki az osztályt egy **Kiir()** függvénnyel, ami visszaadja az adatokat az alábbi formában:

*festo : cim (stilus)*  
*elkelt / nem kelt el*  
*legmagasabbLicit \$ - legutolsoLicitIdeje (összesen: licitekSzama db)*

+. ) **Kiir()** függvény létrehozása helyett a beépített **ToString()** függvényt írd felül.

+. ) Ha nem kelt el a festmény akkor ne írja ki, hogy nem kelt el, hanem a 2. sorban írja ki a licit adatait.

## 2. Feladat

- a.) Vegyél fel egy listát legalább 2 festménnyel.
- b.) Kérj be a felhasználótól egy darabszámot, majd ugyanennyi új festmény adatait, amit adj hozzá a listához.
- +.) Bálint Ferenc is el szeretné adni a festményeit az aukción. Ezeket a festmények.csv állomány tartalmazza az alábbi formában:  
`festo;cim;stilus`  
Olvasd be a fájlt és a tartalmát add hozzá a listához!
- c.) A program véletlenszerűen licitáljon a képekre 20 alkalommal.
- d.) A felhasználó konzolon is licitálhasson a festményekre.
- A felhasználó először a festmény sorszámát adja meg. A sorszám megadásánál használjon index eltolást. (Ha a felhasználó 3-at ad meg akkor a lista 2. elemére licitál). 0 megadásával lépjen ki a felhasználó a licitálásból.
  - Nem létező sorszám esetén a program hibaüzenetet írjon ki, majd kérjen be új sorszámot.
  - Nem szám beírása esetén a program hibaüzenetet írjon ki majd álljon le.
  - Ha a festmény elkelt, akkor hibaüzenetet írjon ki, majd kérjen be új sorszámot.
  - A sorszám megadása után kérje be, hogy milyen mértékkel szeretne licitálni a felhasználó.
  - A megadás ne legyen kötelező, ha a felhasználó egyből entert üt le akkor az alap 10%-os licittel lehessen licitálni.
  - Ha a felhasználó nem számot ad meg akkor a program hibaüzenettel álljon le.
  - A sorszám megadása után, ha az adott festményre több mint 2 perce érkezett utoljára licit akkor állítsa be elkeltre, majd hibaüzenetet írjon ki, majd kérjen be új sorszámot
  - Miután a felhasználó befejezte a licitálást, az összes olyan festmény, amire érkezett licit legyen elkelt.
- e.) Írd ki a festményeket a konzolra!

## 3. Feladat

- a.) Keresd meg a legdrágábban elkelt festményt, majd az adatait konzolra.
- b.) Döntsd el, hogy van-e olyan festmény, amelyre 10-nél több alkalommal licitáltak.
- c.) Számold meg, hogy hány olyan festmény van, amely nem kelt el.
- d.) Rendezd át a listát a Legmagasabb Licit szerint csökkenő sorrendben, majd írd ki újra a festményeket.
- +.) A rendezett lista tartalmát írd ki egy festmenyek\_rendezett.csv nevű fájlba.