

1. Feladat

- Hozz létre egy új konzol alkalmazást BejegyzesProjekt névvel.
- Készíts egy osztály Bejegyzes névvel, az alábbi UML diagram alapján.

Bejegyzes
-szerzo : string -tartalom : string -likeok : int -letrejott : DateTime -szerkesztve : DateTime
+Bejegyzes(szerzo : string, tartalom : string) +Szerzo {get;} : string +Tartalom {get; set;} : string +Likeok {get;} : int +Letrejott {get;} : DateTime +Szerkesztve {get;} : DateTime +Like() : void

Az osztály példányosításakor állítsa be a likeok számát 0-ra, a létrehozás és a szerkesztés idejét pedig az aktuális időpontra.

A tartalom módosításakor változzon a szerkesztés ideje is az aktuális időpontra.

A Like() eljárás növelje a likeok számát 1-el.

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

szerzo – likeok – letrejott

Szerkesztve: szerkesztes

tartalom

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

+) Csak akkor íja ki a szerkesztés dátumát, ha a bejegyzés módosítva lett.

2. Feladat

- Vegyél fel egy két bejegyzést tartalmazó listát
- Kérj be a felhasználótól egy darabszámot, majd ugyanennyi új bejegyzést, amit adj hozzá a listához.

+) A bejegyzesek.txt további bejegyzéseket tartalmaz az alábbi formában:

szerzo;tartalom

Olvasd be a fájlt és a tartalmát add hozzá a listához.

- Ossz ki véletlenszerűen hússzor annyi likeot mint ahány bejegyzés található a listában.
- A felhasználó módosítsa a 2. bejegyzést.
- Írd ki a bejegyzéseket a konzolra.

3. Feladat

- Keresd meg a legnépszerűbb bejegyzést, majd a likeok számát írd ki a konzolra.
 - Döntsd el, hogy van-e olyan bejegyzés, amely 35-nél több likeot kapott.
 - Számold meg, hogy hány olyan bejegyzés van, amely 15-nél kevesebb likeot kapott.
 - Rendezd át a listát a likeok szerint csökkenő sorrendben, majd írd ki újra a bejegyzéseket.
- +) A rendezett lista tartalmát írd ki egy bejegyzesek_rendezett.txt fájlba.