

2. feladat – Szarvasmarhatelep**40 pont**

A Happy-Cows Nyírség Kft. szíven viseli bolygónk jövőjét, és felelős állattartó céggént az állatok jólétét is. Ezért folyamatosan vezetnek be újításokat, amelyek az üvegházhatású gázok mérséklésével vagy a szarvasmarhák kedvezőbb életkörülményeivel vannak összefüggésben. Az egyik telephely egyik istállójában automata fejőrendszert vezettek be, ahol a tejelő szarvasmarhákat teljesen automatikusan fejik, akkor, amikor az állatok azt szükségesnek tartják. Az adatokat egy automata berendezéssel adatbázisba továbbítják. Ennek az adathalmaznak az egyik hétre vonatkozó, már leválogatott és átalakított részével kell ebben a feladatban dolgoznia. Az adatok és a cég is fiktívek. A megoldás során vegye figyelembe a következőket:

- *A képernyőre írást igénylő feladatok eredményének megjelenítése előtt írja a képernyőre a feladat sorszámát (például: 3. feladat)!*
- *Az egyes feladatokban a kiírásokat pontosan a minta szerint készítse el! (Legfeljebb 1 eltérés-pontossággal, az ékezetmentes kiírástól eltekintve.)*
- *Az ékezetmentes kiírások is elfogadottak.*
- *Az azonosítókat kis- és nagybetűkkel is kezdheti, de igyekezzon betartani az adott programozási nyelv konvencióit.*
- *A program megírásakor az állományban lévő adatok helyes szerkezetét nem kell ellenőriznie, feltételezheti, hogy a rendelkezésre álló adatok a leírtaknak megfelelnek.*
- *A megoldását úgy készítse el, hogy az azonos szerkezetű, de tetszőleges bemeneti adatok mellett is helyes eredményt adjon.*

A forrásállomány neve `hozam.txt`, a felépítése a következő: egy sor egy szarvasmarha adott napon lefejt tejét tartalmazza literben. Az adatokat pontosvessző választja el.

A forrásállomány néhány sora:

```
CAE3;3;21
BAB1F2;6;28
BA;5;16
H7;2;24
BAB2;4;21
D1;0;30
BA;2;27
```

Az egyes adatsorokban rendre a következő adatokat tároltuk:

- *azonosító*, a szarvasmarha egyedi azonosítója, csak az angol ábécé nagybetűit és számjegyeket tartalmazhat.
- *nap sorszáma*, 0-6-ig vehet fel értéket, ahol a 0 a vizsgált hét hétfőjét, a 6 a vasárnapját jelöli.
- *menyiség*, az adott napon az adott szarvasmarha által leadott tej mennyisége literben, egész típusú. (Az adatok minimuma 15 liter, maximuma 30 az állományban.)

A minta második sora alapján a BAB1F2 azonosítójú tehén a vizsgált hét vasárnapján 28 liter tejet adott.

A feladat a következő oldalon folytatódik.

A tehének nem minden nap adnak tejet ebben a gazdaságban. Ha egy nap többször is ad tejet egy tehén, az nem több sorban, hanem az adott napon lefejt mennyiség összesítve szerepel az állományban. Minden szarvasmarha legalább egyszer adott tejet a héten. A rendszer nem rögzíti, ha egy szarvasmarha 0 liter tejet adott, azt úgy kezelik, mintha oda sem ment volna.

1. Készítsen **konzolos alkalmazást** tehének néven a következő feladatok megoldására!
2. Projektjében tegye elérhetővé az Ön által használt programozási nyelvnek megfelelő, a forrásoknál megadott `java.txt`, vagy az `csharp.txt` állományokban definiált `Tehen` osztályt. Főprogramjában „happyCows” nevű azonosítóval hozzon létre `Tehen` típusú üres listát! Olvassa be és tárolja el az állomány sorait, és adatsorokként dolgozza fel azokat a következő algoritmus szerint:

```
Szöveges id := adatSor.vág(';')[0];
Szöveges nap := adatSor.vág(';')[1];
Szöveges mennyiseg := adatSor.vág(';')[2];
Tehen aktTehen := new Tehen(id); // aktuális tehén

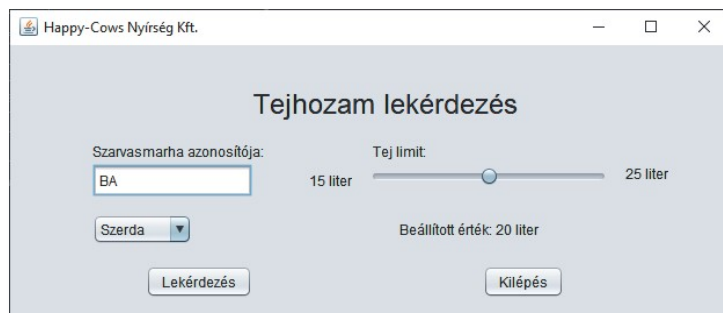
Ha NEM happyCows.tartalmazza(aktTehen) //Ha még nincs a listában
    akkor: happyCows.veddFelListába(aktTehen); //vedd fel
// Ahol van az aktuális tehén a listában,
Egész index=happyCows.index(aktTehen);
// Az adott naphoz rögzítsd a mennyiseg-et:
happyCows[index].eredmenytRogzit(nap, mennyiseg);
```

A Java-ban a listák elemeinek elérésére használt metódus a `lista.get(index)`, a `lista[index]` helyett.

A megadott osztályt tetszőlegesen bővítheti, átalakíthatja a megoldások érdekében.

3. Hány szarvasmarha adatait tartalmazza a forrás? A beolvasott adatok alapján az eredményt írja ki a képernyőre!
4. Készítsen jellemzőt/metódust a `Tehen` osztályban `hetiTej` néven, amely az adott tehén heti összesített tejmennyiségét adja meg!
5. Készítsen egész típusú értékkel visszatérő jellemzőt/metódust a `Tehen` osztályban `hetiAtlag` néven, amely egy szarvasmarha átlagos fejési eredményét adja vissza, de csak abban az esetben, ha legalább 3 nap adott tejet. Ellenkező esetben a jellemző/metódus `-1` értéket adjon vissza! A kerekítéshez tetszőleges módszert választhat. Használja fel az előző feladatban készített jellemzőt/metódust a feladat megoldásához!
6. Határozza meg a legalább 3 nap tejet adó szarvasmarhákat! A meghatározott szarvasmarhák adatait soronként írja ki a `joltejelok.txt` állományba, a kimeneti minta alapján. A tehén azonosítóját a heti átlagmennyiség követi, amelyet a `hetiAtlag` metódussal/jellemzővel határozott meg. A két adat között az elválasztó karakter szóköz legyen. Hány tehén adatait írta ki az állományba? Az eredményt a minta szerint jelenítse meg!

7. A Happy-Cows Nyírség Kft.-nél úgy gondolták, hogy az újszülött borjak azonosítása során az anya azonosítóját használják fel, és ezt egészítik ki két új karakterrel jobb oldalon. A forrásállományban a BA jelű tehénnek több leszármazottja szerepel, így közvetlen leszármazottja a BAB2 vagy a BAB1 jelű tehén, míg ennek az utódja a BAB1F2 azonosítójú szarvasmarha. Kérjen be a felhasználótól egy azonosítót! Számolja meg, hogy a bekért azonosítójú tehénnek összesen hány leszármazottja van az állományban! Az eredményt a minta szerint jelenítse meg a képernyőn. (Ha a megadott tehén nem létezne az állományban, annak nyilván 0 leszármazottja van.)
8. Készítsen **grafikus alkalmazást** a tehenészet munkájának megkönnyítésére! A grafikus alkalmazás projektjét `tehenAdatok` néven hozza létre! A konzolos alkalmazásból tetszőlegesen emelhet át kódrészleteket.
- a. Alakítsa ki a felhasználói felületet az alábbi minta alapján! Az elrendezésre és a feliratokra figyeljen, a színeket és a méreteket tetszőlegesen választhatja meg, ha az nincs szabályozva. A beviteli mezők alapértékei a mintán látható értékek legyenek.



- b. A „Kilépés” gombra az alkalmazás záródjon be.
- c. A konzolos alkalmazásban használt adatokat kell itt is használnia. A `javaCows.txt`, illetve a `csharpCows.txt` állományok tartalmazznak egy nyelvspecifikus adatszerkezetet a szükséges adatokkal. Másolja be a kódjába a megfelelő helyre az adatszerkezetet az adatokkal, és a továbbiakban ezzel dolgozzon!
- d. A legördülő listából ki lehessen választani egy napot hétfőtől vasárnapig. Alapértelmezett érték legyen a szerda.
- e. A csúszka segítségével be lehessen állítani 15 és 25 közötti egész számot. Az alapértelmezett érték legyen 20.
- f. A csúszkán beállított aktuális érték a minta szerint legyen leolvasható. (Kövesse a kiírás a csúszka változását.)
- g. A szöveges beviteli mezőben egy szarvasmarha kódját kell megadnia. Tesztelésre ajánljuk a „BA” és a „BAB2” azonosítókat.

A feladat a következő oldalon folytatódik.

- h. A „Lekérdezés” gombra kattintva a mintának megfelelően jelenítse meg az adott szarvasmarhára vonatkozó adatokat, a következők szerint:
- Az eredmény a minta szerint egy felugró ablakban jelenjen meg.
 - Ha az adott napon nem volt fejés, jelenítse meg a „Nem volt fejés” feliratot a minta szerint. (Hibásan megadott kód esetén is ez a felirat jelenjen meg.)
 - Ha volt fejés, de a csúszkán beállított értéknél kevesebb tejet adott, akkor „Az adott napon a mennyiség nem érte el a limitet!” felirat jelenjen meg.
 - Ha volt fejés és elérte a limitet vagy azon is túl volt, akkor meg kell jeleníteni, hogy hány liter a fejés eredménye.

Minta a konzolos alkalmazáshoz:

```
3. feladat
Az állomány 22 tehén adatait tartalmazza.

6. feladat
20 darab sort írtam az állományba.

7. feladat
Kérem adja meg egy tehén azonosítóját!
BA
A leszármazottak száma: 6
```

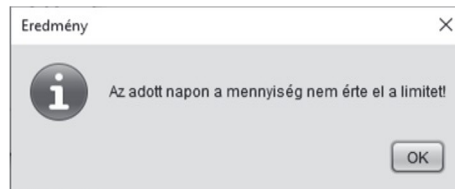
Minta a kimeneti állományhoz:

```
BAB1F2 26
BA 23
H7 22
BAB2 23
D1 26
```

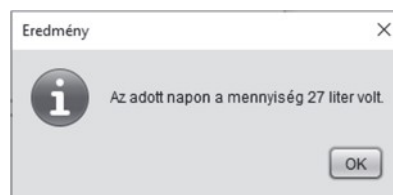
További minták a grafikus alkalmazáshoz:



Eredmény, ha a bemeneti kód „BABAR”



Eredmény, ha a szombaton vizsgáljuk a „BA” tehenet, 20 literes limittel



Eredmény az alapértékek esetén