

OBJEKTUMORIENTÁLT PROGRAMOZÁS 2010.

12. GYAKORLAT

Cél:

- szöveges állományok,
- szövegfeldolgozás: `StringTokenizer`, `Scanner`
- rendezettséget biztosító tároló: `TreeSet<T>`
- asszociatív tömb: `TreeMap`, `HashMap`

1. feladat **Scanner + TreeSet<String>**

Egy szöveges állomány gyümölcsneveket tartalmaz tetszőleges sorrendben, bizonyos gyümölcsneveket akár többször is tartalmazhat.

Végezzük el a következőket:

- dolgozzuk fel a szöveges állományt,
- jelenítsük meg a kimeneten a gyümölcsneveket alfabetikus sorrendben.

Segítség:

```
//Tároló létrehozása
TreeSet<String> set = new TreeSet<String>();
...
//Tároló listázása
Iterator<String> it = set.iterator();
while( it.hasNext())
    System.out.println( it.next());
```

2. feladat **Scanner + TreeSet<Integer>**

Egy szöveges állomány egész számokat tartalmaz. Olvassuk ki és helyezzük el a számokat egy rendezettséget biztosító tárolóban (**`TreeSet<Integer>`**). Írassuk ki a számokat egy kimeneti szöveges állományba úgy, hogy minden sor csak egyetlen számot tartalmazzon.

Segítség:

```
PrintWriter fout = null;
try {
    fout = new PrintWriter(new FileWriter("out.txt"));
}
catch (FileNotFoundException ex) {
    //Hibakezeles
}
fout.println("Hello " + (10+2));
```

3. feladat **HashMap<Character, Integer>**

Készítsünk karakterelőfordulási statisztikát egy szöveges állományra.

OBJEKTUMORIENTÁLT PROGRAMOZÁS 2010.

12. GYAKORLAT

4. feladat Rendezettségi kritériumok: **Comparator** interfész

Készítsen egy `Szemely` osztályt a következő tagokkal (vezetéknév, keresztnév, születési év). Írja felül az `equals` metódust is.

A `Szemely` osztályhoz készítsen két *rendezési kritériumot*:

- egy név szerinti rendezést, amely a vezetéknév és a keresztnév alapján alfabetikus sorrendet biztosít,
- egy születési év szerinti növekvő sorrendet megadó rendezési kritériumot.

Készítsen egy programot, amely egy szöveges állományból személyadatokat olvas be (minden sorban egyetlen személy adatai vannak) majd elhelyezi egy név szerinti alfabetikus sorrendet biztosító tárolóba. Írassa ki a tárolót egy `nevszerintrendezett.txt` állományba.

Helyezze át az adatokat egy születési év szerinti növekvő sorrendet biztosító tárolóba, majd ezt is írassa ki egy `szulesiesievszerintrendezett.txt` szöveges állományba

```
Set<Szemely> set1 = new TreeSet<Szemely>( new NevComparator() );
```

```
Set<Szemely> set2 = new TreeSet<Szemely>( new EvComparator() );
```