

CSC 210 HOMEWORK 1 FALL 2015

PROFESSOR GODFREY MUGANDA

This assignment will afford you an opportunity to work with Java's Stream API, comparators, maps, and lambda expressions. Hints will be given as to how to proceed, but you are free to come up with alternative solutions.

The goal of the assignment is to write a program that opens a file that has name and age information about some children. The syntax of the file is line-oriented, with information for each child being on a line by itself. The file may have blank lines randomly interspersed with the lines that contain information. Here is a sample file.

```
Mary 10
Joey 5
Suzy 8
Amina 10
Pedro 5

Carlos 6
Lester 7
Malia 5
Sasha 10
Sidney 8
Selma 8

Zoey 6
```

You are required to read this file, parse it, and store the information for each child in an object of the following class:

```
class Kid
{
    String name;
    int age;
    public Kid(String name, int age)
    {
        this.name = name;
        this.age = age;
    }
    @Override
    public String toString()
    {
        StringBuilder builder = new StringBuilder();
        builder.append(name).append(" ").append(age);
        return builder.toString();
    }
}
```

```
    }  
}
```

You must use this class as is, without altering it.

Here is what your program must do. You open the file, read it, parse the information into and store it into an array, or list, or set of `Kid` objects. Your program needs to output the `Kid` information three different ways:

- (1) Same order as found in the file
- (2) In alphabetic order of names
- (3) In descending order of age

Once this is done, your program should output the `Kid` information grouped by age, in ascending order of age, with the names in each age group in alphabetic order. Here is output corresponding to the sample input file.

Here is the list in original order:

```
Mary 10  
Joey 5  
Suzy 8  
Amina 10  
Pedro 5  
Carlos 6  
Lester 7  
Malia 5  
Sasha 10  
Sidney 8  
Selma 8  
Zoey 6
```

Here is the list in alphabetic order of names:

```
Amina 10  
Carlos 6  
Joey 5  
Lester 7  
Malia 5  
Mary 10  
Pedro 5  
Sasha 10  
Selma 8  
Sidney 8  
Suzy 8  
Zoey 6
```

Here is the list in descending order of age:

```
Amina 10  
Mary 10  
Sasha 10  
Selma 8  
Sidney 8  
Suzy 8  
Lester 7  
Carlos 6
```

Zoey 6
Joey 5
Malia 5
Pedro 5

Here are the age Groups:
5 : [Joey, Malia, Pedro]
6 : [Carlos, Zoey]
7 : [Lester]
8 : [Selma, Sidney, Suzy]
10 : [Amina, Mary, Sasha]

To get you started, I have provided some starter code in a zipped-up Netbeans project folder. You can find a link to it off of the course website.

Here are some hints. Use a `List<Kid>` object to hold the information passed from the file. The `Collections` class has a `sort()` method that takes a list to sort and a comparator as its parameters. To achieve the grouping, use a map of type `Map<Integer, Set<String>>` or `Map<Integer, List<String>>` where the keys to the map are kids' ages and the associated value is either a list or set of strings.

This assignment is due Friday at the end of Week 2 at midnight. Submit by zipping up the entire Netbeans folder that contains your program and attaching the zipped-up folder to an email.