Assignment 5

- Due Nov 24 by 11:59pm
- Points 20

All assignments are emailed to cislabs05@gmail.com

Object Relationship and File IO

Write a program to perform statistical analysis of scores for a class of students.The class may have up to 40 students.There are five quizzes during the term. Each student is identified by a four-digit student ID number.

The program is to print the student scores and calculate and print the statistics for each quiz. The output is in the same order as the input; no sorting is needed. The input is to be read from a text file. The output from the program should be similar to the following:

Here is some sample data (not to be used) for calculations:

Stud Q1 Q2 Q3 Q4 Q5

1234 78 83 87 91 86

2134 67 77 84 82 79

1852 77 89 93 87 71

High Score 78 89 93 91 86

Low Score 67 77 84 82 71

Average 73.4 83.0 88.2 86.6 78.6

The program should print the lowest and highest scores for each quiz.

Plan of Attack

Learning Objectives

You will apply the following topics in this assignment:

- File Input operations.
- Working and populating an array of objects.
- Wrapper Classes.

- Object Oriented Design and Programming.

Understanding Requirements

Here is a copy of actual data to be used for input.

Stud Qu1 Qu2 Qu3 Qu4 Qu5

1234 052 007 100 078 034

2134 090 036 090 077 030

3124 100 045 020 090 070

4532 011 017 081 032 077

5678 020 012 045 078 034

6134 034 080 055 078 045

7874 060 100 056 078 078

8026 070 010 066 078 056

9893 034 009 077 078 020

1947 045 040 088 078 055

2877 055 050 099 078 080

3189 022 070 100 078 077

4602 089 050 091 078 060

5405 011 011 000 078 010

6999 000 098 089 078 020

Essentially, you have to do the following:

- Read Student data from a text file.
- Compute High, Low and Average for each quiz.
- Print the Student data and display statistical information like High/Low/Average..

Design

This program can be written in one class. But dividing the code into simple and modular classes based on functionality, is at the heart of Object Oriented Design.

You must learn the concepts covered in the class and find a way to apply.

Please make sure that you put each class in its own .java file.

```
class Student {
    private int SID;
    private int scores[] = new int[5];
    //write public get and set methods for
    //SID and scores
    //add methods to print values of instance variables.
}
/*****************************************************************************/

class Statistics
{
    private int [] lowscores = new int [5];
    private int [] highscores = new int [5];
    private float [] avgscores = new float [5];
    public void findlow(Student [] a) {
        /* This method will find the lowest score and store it in an   array names lowscores. */
    }
    public void findhigh(Student [] a) {
    /* This method will find the highest score and store it in an     array names highscores. */
    }
    public void findavg(Student [] a) {
    /* This method will find avg score for each quiz and store it in an array names avgscores. */
    }
    //add methods to print values of instance variables.
    public void print(int option) {
```

```
        //1 - print low scores, 2 - print high scores, 3 - print avg - 4 print all.
      }
}
*****************************************************************************/


class Util {
   private String fname;
   //add constructor as needed.
     public void readFile(String filename, Student [] stu) {
     //This should be an instance method
     //Reads the file and builds student array.
     //Open the file using FileReader Object.
     //In a loop read a line using readLine method.
     //Tokenize each line using StringTokenizer Object
     //Each token is converted from String to Integer using parseInt method
     //Value is then saved in the right property of Student Object.
     }
}
*****************************************************************************/
//Putting it together in driver class:
    public static void main(String [] args) {
          Student lab2 [] = new Student[40];
          //Populate the student array
          lab2 = Util.readFile("filename.txt", lab2);
          Statistics statlab2 = new Statistics();
          statlab2.findlow(lab2);
          //add calls to findhigh and find average
```

```
        //Print the data and statistics
        }
```

Topics to Learn

Working with Text Files

```
//ReadSource.java -- shows how to work with readLine and FileReader
public class ReadSource {
    public static void main(String[] arguments) {
        try {
            FileReader file = new FileReader("ReadSource.java");
            BufferedReader buff = new BufferedReader(file);
            boolean eof = false;
            while (!eof) {
                String line = buff.readLine();
                if (line == null)
                    eof = true;
                else
                    System.out.println(line);
          }
          buff.close();
      } catch (IOException e) {
          System.out.println("Error -- " + e.toString());
      }
  }
}


//How do you tokenize a String? You can use other ways of doing this, if you like.
   StringTokenizer st = new StringTokenizer("this is a test");
```

```
    while (st.hasMoreTokens()) {
        System.out.println(st.nextToken());
    }
//How to convert a String to an Integer
    int x = Integer.parseInt(String) ;
```