Programming Fundamentals

Program Name: **BCS – IT**

Course Code: **CSC 1510**

Course Name: **Programming Fundamentals**

Assignment: **Third(Short Semester)**

Date of Submission: **10th May, 2020**

Submitted By:

Student Name: **Keshav Bhandari**

IUKL ID:

Semester: Second

Intake: September, 2019

Submitted To:

Faculty Name: **Prakash Chandra**

Department: BCS - IT

Programming Fundamentals

1. **Answer the following:**
   a. **Explain static variable:**

      Answer:

      A static variable is common to all the instances (or objects) of the class because it is a class level variable. In other words, you can say that only a single copy of static variable is created and shared among all the instances of the class. Memory allocation for such variables only happens once when the class is loaded in the memory.

      JAVA CODE:

```
class Student {
   int id;
   String name;
   static String college_name = "Sunway International Business School";

   Student(int id, String name) {
      this.id = id;
      this.name = name;
   }

   void display() {
     System.out.println("*****************Students Details*****************");
      System.out.println("College Name: " + college_name);
      System.out.println("Student Name: " + this.name);
      System.out.println("Student Id Number: " + this.id);
   }
}

public class StaticVariable {
   public static void main(String[] args) {
      Student s1 = new Student(1001, "Keshav Bhandari");
      s1.display();
      System.out.println();
      Student s2 = new Student(1002, "Ram Bhandari");
      s2.display();

   }
}
```

Output:

```
C:\Users\VIRUS\Desktop\3rd Assignment>javac StaticVariable.java

C:\Users\VIRUS\Desktop\3rd Assignment>java StaticVariable
*****************Students Details*****************
College Name: Sunway International Business School
Student Name: Keshav Bhandari
Student Id Number: 1001

*****************Students Details*****************
College Name: Sunway International Business School
Student Name: Ram Bhandari
Student Id Number: 1002

C:\Users\VIRUS\Desktop\3rd Assignment>
```

b. Write a java program to count the number of objects created, using the concept of static variable.
Answer:
JAVA CODE:

```
import java.util.Scanner;

class Object {
   String name;
   int age;
   static int numberOfObject = 0;

   Object(String name, int age) {
      this.name = name;
      this.age = age;
      numberOfObject++;
   }

   void display() {
      System.out.println("Name: " + this.name);
      System.out.println("Age: " + this.age);
      System.out.println("The total number of object created is: " + numberOfObject);
   }
}


public class CountObject {
   public static void main(String[] args) {
      Scanner sc = new Scanner(System.in);
```

```
        System.out.println("Enter details of s1 below: ");
        System.out.print("Name: ");
        String name = sc.next();
        System.out.print("Age: ");
        int age = sc.nextInt();
        Object s1 = new Object(name, age);
        System.out.println("==================================");

        s1.display();
        System.out.println();
        System.out.println("Enter details of s2 below: ");
        System.out.print("Name: ");
        name = sc.next();
        System.out.print("Age: ");
        age = sc.nextInt();
        Object s2 = new Object(name, age);
        s2.display();
        System.out.println("==================================");
        System.out.println();
        System.out.println("Total Objects Created is " + Object.numberOfObject);
        sc.close();
    }

}
```

Output:

```
C:\Users\VIRUS\Desktop\3rd Assignment>javac CountObject.java

C:\Users\VIRUS\Desktop\3rd Assignment>java CountObject
Enter details of s1 below:
Name: Keshav
Age: 21
==================================
Name: Keshav
Age: 21
The total number of object created is: 1

Enter details of s2 below:
Name: Ram
Age: 30
Name: Ram
Age: 30
The total number of object created is: 2
==================================

Total Objects Created is 2

C:\Users\VIRUS\Desktop\3rd Assignment>
```

2. **You've been provided a .csv (comma separated values) file of student IDs and test scores. You are to write a class that reads a line from the .csv file, parse the input string into its constituent values, compute an average of the scores for each student, convert the average into a letter grade, and print the student ID, scores, average of the scores and the letter equivalent of the average for each student. Also, track min and max averages, and calculate the class average. After all student data has been processed, print a summary for the class. Your output should match the formatting in the example output below in *both the terminal* and an output file named *gradesout.txt*.**

GRADE is defined as follows:
[100-90]: A
[90-80]: B
[80-70]: C
[70-60]: D
[60-0]: F

```
Student#     Scores                                  Avg   Grade
-----------------------------------------------------------------
12449        99 92 90 95 89 83 91 99 80 90   90.80   A
23452        82 95 83 89 88 90 83 91 99 88   88.80   B
44278        79 83 88 69 50 93 91 58 68 75   75.40   C
15567        75 69 99 63 56 00 25 35 69 54   54.50   F
24555        89 95 86 83 78 99 75 63 94 84   84.60   B
67990        73 84 56 69 55 99 59 55 69 68   68.70   D

---------------- Summary Statistics ----------------------
Total Number of Students: 6
Max Average   90.80  A
Min Average   54.50  F
Class Average 77.13  C
```

JAVA CODE:

```
import java.io.File;
import java.io.FileWriter;
import java.text.DecimalFormat;
import java.util.Arrays;
import java.util.Scanner;

public class Question2 {

   public static String CheckGrade(double avg) {
      if (avg >= 90 && avg <= 100) {
         return "A";
      } else if (avg >= 80 && avg <= 90) {
         return "B";
      } else if (avg >= 70 && avg <= 80) {
         return "C";
      } else if (avg >= 60 && avg <= 70) {
         return "D";
      } else if (avg >= 0 && avg <= 60) {
         return "F";
      }
      else {
         return "Invalid Input"; // because if average is more than 100 it will also show grade
 F therefore
      }
   }

   // Method for getting the maximum value
   public static double getMax(double[] average) {
      double maxValue = average[0];
      for (int i = 1; i < average.length; i++) {
         if (average[i] > maxValue) {
            maxValue = average[i];
         }
      }
      return maxValue;
   }

   // Method for getting the minimum value
   public static double getMin(double[] average) {
      double minValue = average[0];
      for (int i = 1; i < average.length; i++) {
         if (average[i] < minValue) {
```

```
            minValue = average[i];
        }
      }
      return minValue;
   }

   public static void main(String[] args) {
      int sizeOfArray = 0;
      DecimalFormat twoDigits = new DecimalFormat(".00");
      try {
         Scanner sc1 = new Scanner(new File("scores10.csv"));
         while (sc1.hasNext()) {
            sc1.nextLine();
            sizeOfArray++;
         }
         // scores
         int id[] = new int[sizeOfArray];
         double average[] = new double[sizeOfArray];
         String grade[] = new String[sizeOfArray];
         String score[] = new String[sizeOfArray];
         Arrays.fill(score, "");
         double class_average = 0;
         Scanner sc2 = new Scanner(new File("scores10.csv"));
         for (int i = 0; i < sizeOfArray; i++) {
            String line = sc2.nextLine();
            String arr[] = line.split(",");
            id[i] = Integer.parseInt(arr[0]);
            for (int j = 1; j < arr.length; j++) {
               String a = arr[j];
               score[i] = score[i] + " " + a;
      average[i] = average[i] + Double.parseDouble(arr[j]) / (arr.length - 1);
            }
            double avg = average[i];
            grade[i] = CheckGrade(avg);
            class_average = (class_average + average[i]);
         }
         // output for terminal
         System.out.println("Student# \t\t Scores \t\t\t\t Avg \t\t Grade");
         System.out.println(
               "-----------------------------------------------------------------------------------------------
");
         for (int i = 0; i < id.length; i++) {
            System.out.println(
```

```
                id[i] + " \t\t " + score[i] + " \t\t " + twoDigits.format(average[i]) + " \t\t " + gr
ade[i]);
        }
        // summmary
        // Calling getMax() method for getting max value
        double max = getMax(average);
        // Calling getMin() method for getting min value
        double min = getMin(average);
        // ======================================================
        System.out.println(
                "---------------------------------------Summary Statistics-----------------------------
-----");
        System.out.println("Total Number of Students: " + sizeOfArray);
        System.out.println("Maximum Average: " + twoDigits.format(max) + "\t" + CheckG
rade(max));
        System.out.println("Maximum Average: " + twoDigits.format(min) + "\t" + CheckG
rade(min));
   System.out.println("Class Average: " + twoDigits.format((class_average / sizeOfArray))
+ "\t"
                + CheckGrade(class_average / sizeOfArray));

        // output for gradesout.txt
        FileWriter out = null;
        out = new FileWriter("gradesout.txt");
        out.write("Student# \t\t Scores \t\t\t\t\t\t\t Avg \t\t Grade\n");
        out.write(
                "-------------------------------------------------------------------------------------------
\n");
        for (int i = 0; i < id.length; i++) {
out.write(id[i] + " \t\t " + score[i] + " \t\t " + twoDigits.format(average[i]) + " \t\t " + grade
[i]
                    + "\n");
        }
        // summmary
        // Calling getMax() method for getting max value
        max = getMax(average);
        // Calling getMin() method for getting min value
        min = getMin(average);
        // ======================================================
        out.write(
                "---------------------------------------Summary Statistics-----------------------------
-----\n");
```

```
out.write("Total Number of Students: " + sizeOfArray + "\n");
out.write("Maximum Average: " + twoDigits.format(max) + "\t" + CheckGrade(max) + "\n"
);
out.write("Maximum Average: " + twoDigits.format(min) + "\t" + CheckGrade(min) + "\n"
);
 out.write("Class Average: " + twoDigits.format((class_average / sizeOfArray)) + "\t"
                + CheckGrade(class_average / sizeOfArray) + "\n");
            out.close();
            sc2.close();
            sc1.close();
        } catch (

        Exception e) {
            System.out.println("An error occured.");
            e.printStackTrace();
        }
    }
}
```

Output: