

## LAB # 2

### Good practices of programming

#### Lab Task:-

1. Create a design for the mark sheet by taking runtime value of student name, total marks, obtained marks and calculate its percentage, grade and GPA. Use good practices of programming that we have studied and ensure that the outcomes should be presented in a proper Viewable approach.

#### Code:-

```
Marksheet.java x
1  import java.util.Scanner;
2  public class Marksheet {
3      public static void main(String[] args) {
4          Scanner sc = new Scanner(System.in);
5
6          System.out.print("Enter Student Name: ");
7          String name = sc.nextLine();
8
9          System.out.print("Enter Total Marks: ");
10         double totalMarks = sc.nextDouble();
11
12         System.out.print("Enter Obtained Marks: ");
13         double obtainedMarks = sc.nextDouble();
14
15         if (obtainedMarks > totalMarks || obtainedMarks < 0 || totalMarks <= 0) {
16             System.out.println("Invalid marks entered.");
17             return;
18         }
19         double percentage = (obtainedMarks / totalMarks) * 100;
20         String grade;
21         if (percentage >= 90)
22             grade = "A+";
23         else if (percentage >= 80)
24             grade = "A";
25         else if (percentage >= 70)
26             grade = "B";
27         else if (percentage >= 60)
28             grade = "C";
29         else if (percentage >= 50)
30             grade = "D";
31         else
32             grade = "F";
```



```

34         double gpa;
35         switch (grade) {
36             case "A+":
37                 gpa = 4.0;
38                 break;
39             case "A":
40                 gpa = 3.7;
41                 break;
42             case "B":
43                 gpa = 3.0;
44                 break;
45             case "C":
46                 gpa = 2.0;
47                 break;
48             case "D":
49                 gpa = 1.0;
50                 break;
51             default:
52                 gpa = 0.0;
53         }
54         System.out.println("\n==== MARK SHEET =====");
55         System.out.println("Student Name : " + name);
56         System.out.println("Total Marks : " + totalMarks);
57         System.out.println("Obtained Marks: " + obtainedMarks);
58         System.out.println(percentage);
59         System.out.println("Grade      : " + grade);
60         System.out.println("GPA       : " + gpa);
61         System.out.println("=====");
62     }
63 }

```

### Output:-

```

C:\Users\LAPCOM\.jdk\openjdk-25\t
Enter Student Name: Rayyan
Enter Total Marks: 500
Enter Obtained Marks: 435

==== MARK SHEET =====
Student Name : Rayyan
Total Marks : 500.0
Obtained Marks: 435.0
87.0
Grade      : A
GPA       : 3.7
=====

Process finished with exit code 0

```

2. Create a class Rectangle with attributes length and width, each of which defaults to 1. Provide methods that calculate the rectangle's perimeter and area. It has set and get methods for both length and width. The set methods should verify that length and width are each floating-point numbers larger than 0.0 and less than 20.0. Write a program to test class Rectangle.



**Code:-**

```

1  class Rectangle { no usages
2      private double length = 1.0; 4 usages
3      private double width = 1.0; 4 usages
4
5      public double getLength() { no usages
6          return length;
7      }
8      public void setLength(double length) { no usages
9          if (length > 0.0 && length < 20.0) {
10             this.length = length;
11         } else {
12             System.out.println("Invalid length. Must be > 0.0 and < 20.0");
13         }
14     }
15     public double getWidth() { no usages
16         return width;
17     }
18     public void setWidth(double width) { no usages
19         if (width > 0.0 && width < 20.0) {
20             this.width = width;
21         } else {
22             System.out.println("Invalid width. Must be > 0.0 and < 20.0");
23         }
24     }
25     public double getPerimeter() { no usages
26         return 2 * (length + width);
27     }
28     public double getArea() { no usages
29         return length * width;
30     }
31 }

```

```

1  public class Main {
2      public static void main(String[] args) {
3          Rectangle rect = new Rectangle();
4
5          System.out.println("Default length: " + rect.getLength());
6          System.out.println("Default width: " + rect.getWidth());
7          System.out.println("Default perimeter: " + rect.getPerimeter());
8          System.out.println("Default area: " + rect.getArea());
9
10         rect.setLength(15.5);
11         rect.setWidth(10.0);
12
13         System.out.println("\nAfter setting new length and width:");
14         System.out.println("Length: " + rect.getLength());
15         System.out.println("Width: " + rect.getWidth());
16         System.out.println("Perimeter: " + rect.getPerimeter());
17         System.out.println("Area: " + rect.getArea());
18     }
19 }

```

**Output:-**

```

C:\Users\LAPCOM\.jdk\openjdk-25\bin
Default length: 1.0
Default width: 1.0
Default perimeter: 4.0
Default area: 1.0

After setting new length and width:
Length: 15.5
Width: 10.0
Perimeter: 51.0
Area: 155.0

Process finished with exit code 0

```