Day2_Java_Assignment1

1. Primitive Data Types

Task: Create a program that accepts age, height, and weight of a person and prints them with appropriate data types.

Sample Input:
Age: 25
Height: 5.9
Weight: 68.5

Sample Output:

```
Age: 25

Height: 5.9
Weight: 68.5
```

```java
package Java_assignment;
import java.util.Scanner;

public class Primitive_Data_Types_Example {

    public static void main(String[] args) {
         Scanner sc = new Scanner(System.in);
        System.out.print("Enter Age: ");
        int age = sc.nextInt();

        System.out.print("Enter Height: ");
        float height = sc.nextFloat();

        System.out.print("Enter Weight: ");
        double weight = sc.nextDouble();

        System.out.println("\nAge: " + age);
        System.out.println("Height: " + height);
        System.out.println("Weight: " + weight);
    }
}
```

2. Variables

Task: Declare and initialize different types of variables to store a student's information: ID, name, marks, and grade. Print them.

Sample Input:
ID: 101
Name: Arun
Marks: 89.5
Grade: A

Sample Output:

```
Student ID: 101

Name: Arun
Marks: 89.5
Grade: A
```

```
package Java_assignment;
import java.util.Scanner;

public class VariableExample {

    public static void main(String[] args) {
         Scanner sc = new Scanner(System.in);
        System.out.println("Enter Student ID: ");
        int studentID = sc.nextInt();

        System.out.println("Enter Name: ");
        String name = sc.next();

        System.out.println("Enter Marks: ");
        double marks = sc.nextDouble();

        System.out.println("Enter Grade: ");
        char grade = sc.next().charAt(0);

        System.out.println("\nStudent ID: " + studentID);
        System.out.println("Name: " + name);
        System.out.println("Marks: " + marks);
        System.out.println("Grade: " + grade);
    }

}
```

3. Operators

Task: Accept two numbers and perform arithmetic, relational, and logical operations on them.

Sample Input:
Number1: 10
Number2: 20

Sample Output:

```
Addition: 30
```

```
Greater number: 20
Are both positive? True
```

```
package Java_assignment;
import java.util.Scanner;

public class OperatorExample {

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

        System.out.print("Enter Number1: ");
        int n1 = sc.nextInt();
        System.out.print("Enter Number2: ");
        int n2 = sc.nextInt();

        int addition = n1 + n2;
        int greater = (n1 > n2) ? n1 : n2;
        boolean bothPositive = (n1 > 0) && (n2 > 0);

        System.out.println("\nAddition: " + addition);
        System.out.println("Greater number: " + greater);
        System.out.println("Are both positive? " +
bothPositive);
    }
}
```

4. String Concatenation

Task: Create a greeting message using first name and last name entered by the user.

Sample Input:
First Name: Ravi
Last Name: Kumar

Sample Output:

```
Hello, Ravi Kumar! Welcome to the system.
```

```
package Java_assignment;
import java.util.*;

public class StringConcat {

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

```
        String greet = "Welcome, " + first_name + " " +
last_name + "!.";

        System.out.println(greet);

    }
}
```

5. StringBuilder

Task: Accept a sentence and reverse it using `StringBuilder`.

Sample Input:
Input: Hello Java Learners

Sample Output:

```
Original: Hello Java Learners

Reversed: srenraeL avaJ olleH
```

```
package Java_assignment;
import java.util.*;
public class StringBuilderExample {
public static void main(String[] args) {
        Scanner sc= new Scanner(System.in);
        StringBuilder sb = new StringBuilder();
        String str = sc.nextLine();
        sb.append(str);
        System.out.println("Original: " + sb.toString());
        System.out.println("Reversed: " + sb.reverse());
    }
}
```

6. String API
Task: Count how many times a specific character appears in a string.

Sample Input:
String: banana
Character: a

Sample Output:

```
Character 'a' appears 3 times.
```

```
package StringPrg;
import java.util.*;
```

```
public class CountCharacter {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Scanner sc = new Scanner(System.in);
        String str = sc.nextLine();
        Character ch = sc.next().charAt(0);
        int len = str.length();
        int count = 0;

        while(len != 0) {
            if(str.charAt(len-1) == ch) {
                count++;
            }
            len--;
        }
        System.out.println("Character '" + ch + "'" + "
appears " + count + " times.");
    }

}
```

7. Date, Time, and Numeric Objects

Task: Display the current date and format it as DD-MM-YYYY. Also, show a formatted currency value.

Sample Input:
Date: [current system date]
Amount: 12345.678

Sample Output:

```
Current Date: 20-07-2025

Formatted Amount: ₹12,345.68
```

```
package StringPrg;
import java.time.LocalDate;
import java.time.LocalDateTime;
import java.time.LocalTime;
import java.time.format.DateTimeFormatter;

public class DateTimeEXample

{
```

```
    public static void main(String[] args) {
        LocalDate d = LocalDate.now();
        LocalTime t = LocalTime.now();
        LocalDateTime dt =
LocalDateTime.now();

        System.out.println("Current Date : "+ d);
        System.out.println("Current Time : "+ t);
        System.out.println("Current Date & time: "+ dt);

        DateTimeFormatter f =
DateTimeFormatter.ofPattern("dd-MM-yyyy HH:mm:ss");
        String fdt = dt.format(f);
        System.out.println(fdt);
    }

}
```

Task: Based on a number entered, print whether it's positive, negative, or zero.
Sample Input:
Number: -5
Sample Output:

```
The number is negative.
```

```
package Java_assignment;

import java.util.Scanner;

public class FlowControl {

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

         System.out.print("Enter a number: ");
         int number = sc.nextInt();

         if (number > 0) {
             System.out.println("The number is positive.");
         } else if (number < 0) {
             System.out.println("The number is negative.");
         } else {
             System.out.println("The number is zero.");
         }
     }
}
```

8. Conditions

Task: Accept marks and display the grade using `if-else`.

Sample Input:
Marks: 76

Sample Output:

```
Grade: B
```

```
package Java_assignment;
import java.util.Scanner;

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

        System.out.print("Enter marks: ");
        int marks = sc.nextInt();

        if (marks >= 90) {
```

```
            System.out.println("Grade: A");
        } else if (marks >= 75) {
            System.out.println("Grade: B");
        } else if (marks >= 60) {
            System.out.println("Grade: C");
        } else if (marks >= 40) {
            System.out.println("Grade: D");
        } else {
            System.out.println("Grade: F (Fail)");
        }
    }
}
```

9. Switch

Task: Build a simple calculator using `switch` to perform operations (+, -, *, /).

Sample Input:
Number1: 10
Number2: 5
Operation: *

Sample Output:
```
Result: 50
```

```
package Java_assignment;
import java.util.Scanner;

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

        System.out.print("Enter first number: ");
        double num1 = sc.nextDouble();

        System.out.print("Enter second number: ");
        double num2 = sc.nextDouble();

        System.out.print("Enter operation (+, -, *, /): ");
        char operator = sc.next().charAt(0);

        double result;

        switch (operator) {
            case '+':
                result = num1 + num2;
                System.out.println("Result: " + result);
```

```
                break;
            case '-':
                result = num1 - num2;
                System.out.println("Result: " + result);
                break;
            case '*':
                result = num1 * num2;
                System.out.println("Result: " + result);
                break;
            case '/':
                if (num2 != 0) {
                    result = num1 / num2;
                    System.out.println("Result: " + result);
                } else {
                    System.out.println("Error: Division by
zero is not allowed.");
                }
                break;
            default:
                System.out.println("Invalid operator.");
        }

        sc.close();
    }
}
```

10. Loops and Branching

Task: Print the first N even numbers using a loop.

Sample Input:
N = 5

Sample Output:

```
0 2 4 6 8
```

```
package Java_assignment;
import java.util.*;
public class LoopExample {
    public static void main(String[] args) {
        Scanner sc=new Scanner(System.in);
        int n=sc.nextInt();
        for(int i=0;i<n*2;i+=2) {
            System.out.print(i+ " ");
        }
    }
}
```

11. Arrays

Task: Accept 5 numbers, store them in an array, and display their average.

Sample Input:
Numbers: 10, 20, 30, 40, 50

Sample Output:
```
Average: 30.0
```

```java
package Java_assignment;

import java.util.Scanner;

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

         int[] numbers = new int[5];
         int sum = 0;

         for (int i = 0; i < 5; i++) {
             numbers[i] = sc.nextInt();
             sum += numbers[i];
         }

         double average = (double) sum / 5;

         System.out.println("Average: " + average);
         }
}
```

12. Enum

Task: Create an enum for days of the week. Print a message depending on the day.

Sample Input:
Day: MONDAY

Sample Output:
```
Start of the work week!
```

```
package Java_assignment;
import java.util.Scanner;

public class EnumExample {
    enum Day {
        MONDAY, TUESDAY, WEDNESDAY, THURSDAY, FRIDAY,
SATURDAY, SUNDAY
    }

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

        System.out.print("Enter day: ");
        String input = sc.next().toUpperCase
        try {
            Day day = Day.valueOf(input);

            switch (day) {
                case MONDAY:
                    System.out.println("Start of the work
week!");
                    break;
                case FRIDAY:
                    System.out.println("Last working day of
the week!");
                    break;
                case SATURDAY:
                case SUNDAY:
                    System.out.println("It's weekend, time
to relax!");
                    break;
                default:
                    System.out.println("It's a weekday.");
            }

        } catch (IllegalArgumentException e) {
```

```
            System.out.println("Invalid day entered.");
        }
    }
}
```

13. OOPs Concepts

Task: Create a `Student` class with fields for name and marks. Create an object and display its data.

Sample Input:
Name: Riya
Marks: 87

Sample Output:

```
Student Name: Riya

Marks: 87
```

```
package Class_objects;
public class Student {
    String name;
    int marks;

    public Student(String name, int marks) {
        this.name = name;
        this.marks = marks;
    }
    public void dislpayData() {
        System.out.println("Student name : " + name);

    System.out.println
("Student marks : " +
marks);
    }

    public static void main(String[] args) {
        Student s1 = new Student("Priya", 92);
        s1.dislpayData();
    }
}
```

14. Inheritance

Task: Create a class `Employee` and a subclass `Manager` that extends `Employee` and adds department information.

Sample Input:
Name: Raj
Salary: 50000
Department: Sales

Sample Output:

```
Name: Raj

Salary: 50000
Department: Sales
```

Employee.Class

```
package Class_objects;

class Employee {

    String name;

    double salary;

    public Employee(String name, double salary) {

        this.name = name;

        this.salary = salary;

    }

    public void displayInfo() {

        System.out.println("Name: " + name);

        System.out.println("Salary: " + salary);

    }

}
```

Manager.class

```
package Class_objects;


class Manager extends Employee {

    String department;

    public Manager(String name, double salary, String
department) {

        super(name, salary);

        this.department = department;

    }

    public void displayInfo() {

        super.displayInfo();

        System.out.println("Department: " + department);

    }

    public static void main(String[] args) {

        Manager m1 = new Manager("Raj", 50000, "Sales");

        m1.displayInfo();

    }

}
```