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

//Answer-----------------------------------------

package Task1;



public class PrimitiveDataTypes {



public static void main(String[] args) {

// TODO Auto-generated method stub

int age = 25;          // integer for age

float height = 5.9f;   // float for height

float weight = 68.5f;  // float for weight



// Print the values

System.out.println("Age: " + 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
□
□


//Answer-----------------------------------------


package Task1;


public class Variable {


public static void main(String[] args) {


// TODO Auto-generated method stub


int studentId = 101;


String name = "Arun";


double marks = 89.5;
```

```
char grade = 'A';


System.out.println("Student 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
```

```
//Answer-----------------------------------
```

```
package Task1;
import java.util.Scanner;

public class Operator {

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


System.out.print("Enter Number1: ");
int number1 = sc.nextInt();
```

```
        System.out.print("Enter Number2: ");
        int number2 = sc.nextInt();


        int sum = number1 + number2;


        int greater = (number1 > number2) ? number1 : number2;


        boolean areBothPositive = (number1 > 0 && number2 > 0);


        System.out.println("Addition: " + sum);
        System.out.println("Greater number: " + greater);
        System.out.println("Are both positive? " + areBothPositive);

        sc.close();
    }
}
```

□

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.
```

```
//Answer-------------------------------------
```

```
package Task1;

import java.util.Scanner;

public class StringContact {

public static void main(String[] args) {

// TODO Auto-generated method stub

  Scanner sc = new Scanner(System.in);
```

```
        System.out.print("Enter First Name: ");
        String firstName = sc.nextLine();
        System.out.print("Enter Last Name: ");
        String lastName = sc.nextLine();
        String message = "Hello, " + firstName + " " + lastName + "! Welcome to the system.";
        System.out.println(message);
        sc.close();
}
}
```

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

//Answer-------------------------------------------------
```

```
package Task1;
import java.util.Scanner;

public class ReverseStringBuilder {

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

        System.out.print("Input: ");
        String input = sc.nextLine();

        // Use StringBuilder to reverse the input
        StringBuilder sb = new StringBuilder(input);
        String reversed = sb.reverse().toString();
```

```
        // Output
        System.out.println("Original: " + input);
        System.out.println("Reversed: " + reversed);

        sc.close();
    }
}
```

□

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.
```

Answer----------

```
package Task1;

import java.util.Scanner;

public class CharacterCount ;

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter a string: ");

 String input = sc.nextLine();

  System.out.print("Enter a character to count: ");

        char ch = sc.next().charAt(0);
```

```
        int count = 0;

        for (int i = 0; i < input.length(); i++) {

            if (input.charAt(i) == ch) {

                count++;

            }

        }

        System.out.println("Character '" + ch + "' appears " + count + " times.");

        sc.close();

    }

}
```

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
```

```
//Answer-----------------------------------------------
```

```
package Task1;
```

```
import java.text.NumberFormat;

import java.time.LocalDate;

import java.time.format.DateTimeFormatter;

import java.util.Locale;

public class DateTime {

   public static void main(String[] args) {

      // Get current date

      LocalDate currentDate = LocalDate.now();

      DateTimeFormatter formatter = DateTimeFormatter.ofPattern("dd-MM-yyyy");

      String formattedDate = currentDate.format(formatter);

      double amount = 12345.678;

      Locale indiaLocale = new Locale("en", "IN");

      NumberFormat currencyFormatter = NumberFormat.getCurrencyInstance(indiaLocale);

      String formattedAmount = currencyFormatter.format(amount);

      System.out.println("Current Date: " + formattedDate);

      System.out.println("Formatted Amount: " + formattedAmount);

   }

}
```

8. Flow Control

Task: Based on a number entered, print whether it's positive, negative, or zero.

Sample Input:
Number: -5

Sample Output:

```
The number is negative.
```

```
//Answer-------------------------------------
```

```
package Task1;

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.");

      }

      sc.close();

   }

}
```

9. Conditions

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

Sample Input:
Marks: 76

Sample Output:
```
Grade: B
```

```
//Answer-----------------------------
```

```
package Task1;

import java.util.Scanner;

public class Conditions {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        System.out.print("Enter marks: ");

        int marks = sc.nextInt();

        String grade

        if (marks >= 90) {

            grade = "A";

        } else if (marks >= 75) {

            grade = "B";

        } else if (marks >= 60) {

            grade = "C";

        } else if (marks >= 40) {

            grade = "D";

        } else {

        grade = "F";

        }

        System.out.println("Grade: " + grade);

        sc.close();

    }

}
```

10. Switch

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

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

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

```
//Answer--------------------------------------------------
```

```
package Task1;


import java.util.Scanner;


public class Switch {

    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 operation = sc.next().charAt(0);

        double result;

        switch (operation) {

            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.");
            }
            break;
        default:
            System.out.println("Invalid operation.");
    }
    sc.close();
  }
}
```

11. Loops and Branching

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

Sample Input: N
= 5

Sample Output:
□
```
0 2 4 6 8
```

```
//Answer-------------------------------------
```

```
package Task1;

public class LoopsBranching {

    public static void main(String[] args) {

        int N = 5;

        System.out.println("First " + N + " even numbers:");

        for (int i = 0; i < N; i++) {

            System.out.print(2 * i + " ");

        }

    }

}
```

12. 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
```

```
//Answer-----------------------------------------
package Task1;

public class Arrays {
```

```
    public static void main(String[] args) {
        int[] numbers = {10, 20, 30, 40, 50};
        int sum = 0;
        for (int num : numbers) {
            sum += num;
        }
        double average = (double) sum / numbers.length;
        System.out.println("Average: " + average);
    }
}
```

13. 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!
```

```
//Answer----------------------------------------
```

```
package Task1;

public class Enum {

    enum Day {

        MONDAY, TUESDAY, WEDNESDAY, THURSDAY, FRIDAY, SATURDAY, SUNDAY

    }

    public static void main(String[] args) {

        Day today = Day.MONDAY;

        switch (today) {

            case MONDAY:

                System.out.println("Start of the work week!");

                break;

            case FRIDAY:
```

```
            System.out.println("Weekend is near!");
            break;
        case SUNDAY:
            System.out.println("Relax, it's Sunday!");
            break;
        default:
            System.out.println("It's a regular day.");
    }
  }
}
```

14. 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

//Answer--------------------------------------
```

```
package Task1;

class Student {
   String name;
   int marks;
   Student(String name, int marks) {
      this.name = name;
```

```
    this.marks = marks;

    void display() {

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

        System.out.println("Marks: " + marks);

    }

}

public class OopsConcept {

    public static void main(String[] args) {

        Student s1 = new Student("Riya", 87);

        s1.display();

    }

}
```

15. 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

//Answer-------------------------------------------------
package Task1;

class Employee {
    String name;
    double salary;
```

```
    Employee(String name, double salary) {
        this.name = name;
        this.salary = salary;
    }
}

class Manager extends Employee {
    String department;

    Manager(String name, double salary, String department) {
        super(name, salary);
        this.department = department;
    }

    void display() {
        System.out.println("Name: " + name);
        System.out.println("Salary: " + salary);
        System.out.println("Department: " + department);
    }
}

public class Inheritence {  // File name should be Inheritence.java
    public static void main(String[] args) {
        Manager m1 = new Manager("Raj", 50000, "Sales");
        m1.display();
    }
}
```

□