

## 1. Primitive Data Types

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

```
package day2Assignment;

import java.util.Scanner;

public class Datatype {

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the age: ");
        int age = sc.nextInt();
        System.out.println("Enter height: ");
        float height = sc.nextFloat();
        System.out.println("Enter Weight: ");
        double weight = sc.nextDouble();
        System.out.println("Enter age: " + age);
        System.out.println("Enter height: " + height);
        System.out.println("Enter Weight: " + weight);
        sc.close();
    }

}
```

## 2. Variables

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

```
package day2Assignment;

import java.util.Scanner;

public class variable {

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the Student ID: ");
        int ID = sc.nextInt();
        sc.nextLine();
        System.out.println("Enter Name: ");
        String Name = sc.nextLine();
        System.out.println("Enter Marks: ");
        double Marks = sc.nextDouble();
        System.out.println("Enter Grade: ");
        char Grade = sc.next().charAt(0);
    }

}
```

```

        System.out.println("Enter the Student ID: " + ID);
        System.out.println("Enter Name: " +Name);
        System.out.println("Enter Marks: " +Marks);
        System.out.println("Enter Grade: " +Grade);
        sc.close();
    }
}

```

### 3. Operators

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

```

package day2Assignment;

import java.util.Scanner;

public class oper {

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter number1: ");
        int num1 = sc.nextInt();
        System.out.println("Enter number2: ");
        int num2 = sc.nextInt();
        int addition = num1 + num2;
        int greater = (num1 > num2) ? num1 : num2;
        boolean bothPositive = (num1 > 0 && num2 > 0);
        System.out.println("Addition: " + addition);
        System.out.println("Greater number: " + greater);
        System.out.println("Are both positive? " + bothPositive);
        sc.close();
    }

}

```

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

```

package day2Assignment;

import java.util.Scanner;

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

```

```

        System.out.println("First Name: ");
        String Name = sc.nextLine();
        System.out.println("Last Name: ");
        String Name2 = sc.nextLine();
        System.out.println("Hello," + Name + " " +Name2+ "!
Welcome to the System.");
        sc.close();
    }

}

```

### 5. StringBuilder Task: Accept a sentence and reverse it using StringBuilder.

```

package day2Assignment;

import java.util.Scanner;

public class bul {

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("input: ");
        String str = sc.nextLine();
        StringBuilder sb = new StringBuilder(str);
        String rev = sb.reverse().toString();
        System.out.println("original: "+str);
        System.out.println("reversed: "+rev);
        sc.close();
    }

}

```

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

```

package day2Assignment;

public class Char {
    public static void main(String[] args) {
        String input = "banana";
        char target = 'a';
        int count = 0;
        for (int i = 0; i < input.length(); i++) {
            if (input.charAt(i) == target) {
                count++;
            }
        }
        System.out.println("Character '" + target + "' 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]**

```
package day2Assignment;

import java.time.LocalDateTime;
import java.time.format.DateTimeFormatter;

public class CurrentDateTime {
    public static void main(String[] args) {

        LocalDateTime now = LocalDateTime.now();

        DateTimeFormatter formatter =
DateTimeFormatter.ofPattern("dd-MM-yyyy HH:mm:ss");
        String formattedDateTime = now.format(formatter);

        System.out.println("Current Date and Time: " +
formattedDateTime);
    }
}
```

**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.**

```
package day2Assignment;

public class NumberCheck {
    public static void main(String[] args) {
        int number = -5;

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

**9. Conditions Task: Accept marks and display the grade using if-else. Sample Input: Marks: 76 Sample Output: Grade: B**

```
import java.util.Scanner;

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

```

System.out.print("Enter marks (0 to 100): ");
int marks = scanner.nextInt();

if (marks >= 90 && marks <= 100) {
    System.out.println("Grade: A (Excellent)");
} else if (marks >= 80) {
    System.out.println("Grade: B (Very Good)");
} else if (marks >= 70) {
    System.out.println("Grade: C (Good)");
} else if (marks >= 60) {
    System.out.println("Grade: D (Satisfactory)");
} else if (marks >= 50) {
    System.out.println("Grade: E (Needs Improvement)");
} else if (marks >= 40) {
    System.out.println("Grade: F (Just Passed)");
} else if (marks >= 0) {
    System.out.println("Grade: Fail");
} else {
    System.out.println("Invalid input! Marks should be
between 0 and 100.");
}

scanner.close();
}
}

```

**10. Switch Task: Build a simple calculator using switch to perform operations (+, -, \*, /).**  
**Sample Input:**

```

import java.util.Scanner;

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

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

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

        System.out.print("Enter second number: ");
    }
}

```

```

double num2 = scanner.nextDouble();

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!");
}

scanner.close()
}

```

**Number1: 10 Number2: 5 Operation: \* Sample Output: Result: 50**

**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**

```

public class EvenNumbers {
    public static void main(String[] args) {
        int N = 5;
        for (int i = 0; i < N; i++) {
            System.out.print(i * 2 + " ");
        }
    }
}

```

**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**

```

import java.util.Scanner;

```

```

public class AverageCalculator {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        int[] numbers = new int[5];
        int sum = 0;

        System.out.println("Enter 5 numbers:");
        for (int i = 0; i < 5; i++) {
            numbers[i] = scanner.nextInt();
            sum += numbers[i];
        }

        double average = sum / 5.0;
        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!**

```

public class DayMessage {
    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("Almost weekend!");
                break;
            case SATURDAY:
            case SUNDAY:
                System.out.println("Enjoy your weekend!");
                break;
            default:
                System.out.println("Keep going!");
        }
    }
}

```

**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**

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

    // Constructor
    public Student(String name, int marks) {
        this.name = name;
        this.marks = marks;
    }

    // Display method
    public void display() {
        System.out.println("Student Name: " + name);
        System.out.println("Marks: " + marks);
    }

    public static void main(String[] args) {
        Student student1 = new Student("Riya", 87);
        student1.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**

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

    // Constructor
    public Employee(String name, double salary) {
        this.name = name;
        this.salary = salary;
    }

    // Display method
    public void display() {
        System.out.println("Name: " + name);
        System.out.println("Salary: " + salary);
    }
}

// Subclass: Manager
class Manager extends Employee {
    String department;

    // Constructor
}
```



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

    // Override display method
    @Override
    public void display() {
        super.display(); // Call Employee display
        System.out.println("Department: " + department);
    }

    // Main method to test
    public static void main(String[] args) {
        Manager mgr = new Manager("Raj", 50000, "Sales");
        mgr.display();
    }
}
```