**Day:- 1**

**Data types, Variables, Operators, Loops & Conditionals, Functions & Arrays**

**1)Data types:**

Data types specify what kind of data a variable can hold.

* **Primitive types:** int, float, double, char, Boolean.
* **Non-primitive types:** String, arrays, objects.

🔹 **Example program:**

public class DataTypes {

public static void main(String[] args) {

int age = 20;

float height = 5.9f;

double weight = 65.5;

char grade = 'A';

boolean isPass = true;

String name = "John";

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

System.out.println("Age: " + age);

System.out.println("Height: " + height);

System.out.println("Weight: " + weight);

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

System.out.println("Passed: " + isPass);

}

}

**Taking Input from the User**

import java.util.Scanner;

public class DataTypes {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter your name: ");

String name = sc.nextLine();

System.out.print("Enter your age: ");

int age = sc.nextInt();

System.out.print("Enter your height (in feet): ");

float height = sc.nextFloat();

System.out.print("Enter your weight (in kg): ");

double weight = sc.nextDouble();

System.out.print("Enter your grade (A/B/C): ");

char grade = sc.next().charAt(0); // read 1st character

System.out.print("Did you pass? (true/false): ");

boolean isPass = sc.nextBoolean();

// Display output

System.out.println("\n--- Student Details ---");

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

System.out.println("Age: " + age);

System.out.println("Height: " + height);

System.out.println("Weight: " + weight);

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

System.out.println("Passed: " + isPass);

sc.close();

}

}

**2) Variables**

Variables store data in memory.

* Declare with type: int x = 5;

🔹 **Example program:**

Top of Form

import java.util.Scanner;

public class Data {

public static void main(String[] args) {

Scanner sc=new Scanner(System.in);

System.out.println("Enter A value");

int a = sc.nextInt();

System.out.println("Enter B Value");

int b= sc.nextInt();

int c;

c=a+b;

System.out.println("Addition Is \t"+c);

sc.close();

}

}

**3) Operators**

**Theory:**

* **Arithmetic:** +, -, \*, /, %
* **Relational:** ==, !=, <, >, <=, >=
* **Logical:** &&, ||, !

🔹 **Example:**

import java.util.Scanner;

public class Data {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        System.out.println("Enter A Value");

        int a = sc.nextInt();

        System.out.println("Enter B Value");

        int b=sc.nextInt();

        int c;

        c=a-b;

        System.out.println("Subtraction Is \t"+c);

        sc.close();

    }

}

**✅ 4) Loops & Conditionals**

**if-else**

**Theory:** Used for decision-making.

🔹 **Example:**

import java.util.Scanner;

public class Data {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.println("Enter A Value");

int a = sc.nextInt();

System.out.println("Enter B Value");

int b=sc.nextInt();

if(a<b)

{

System.out.println("A is bigger than B");

}

else

{

System.out.println("A is lessthan B");

}

sc.close();

}

}

➔ for loop

**Theory:** Repeats code fixed number of times.

🔹 **Example:**

import java.util.Scanner;

public class Data {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.println("Enter n Value");

int n = sc.nextInt();

for (int i=0; i<=10; i++)

{

System.out.println(n + "X" + i + " = "+(n\*i));

}

sc.close();

}

}

➔ while & do-while loops

**Theory:** Repeats while condition is true.

**while loop:**

🔹 **Example:**

public class WhileLoopDemo {

public static void main(String[] args) {

int i = 1;

while (i <= 5) {

System.out.println("Number: " + i);

i++;

}

}

}

**do-while loop:**

public class DoWhileDemo {

public static void main(String[] args) {

int i = 1;

do {

System.out.println("Number: " + i);

i++;

} while (i <= 5);

}

}

**5) Functions**

**Theory:** Functions help reuse code; they take input and return output.

✅ 1. **No Return, No Parameters**Bottom of Form

import java.util.Scanner;

public class Type1 {

public static void add() {

Scanner sc = new Scanner(System.in);

System.out.print("Enter first number: ");

int a = sc.nextInt(); // Read first number

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

int b = sc.nextInt(); // Read second number

int sum = a + b;

System.out.println("Sum (No return, No parameters): " + sum);

sc.close();

}

public static void main(String[] args) {

add(); // Call the method

}

}

✅ 2. **Return, No Parameters**

import java.util.Scanner;

public class Type2 {

public static int add() {

Scanner sc = new Scanner(System.in);

System.out.print("Enter first number: ");

int a = sc.nextInt();

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

int b = sc.nextInt();

int sum = a + b;

sc.close();

return sum;

}

public static void main(String[] args) {

int result = add();

System.out.println("Sum (Return, No parameters): " + result);

}

}

✅ 3. **No Return, With Parameters**

public class Type3 {

public static void add(int a, int b) {

int sum = a + b;

System.out.println("Sum (No return, With parameters): " + sum);

}

public static void main(String[] args) {

java.util.Scanner sc = new java.util.Scanner(System.in);

System.out.print("Enter first number: ");

int num1 = sc.nextInt();

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

int num2 = sc.nextInt();

add(num1, num2); // Pass input to method

sc.close();

}

}

✅ 4. **Return, With Parameters**

public class Type4 {

public static int add(int a, int b) {

return a + b;

}

public static void main(String[] args) {

java.util.Scanner sc = new java.util.Scanner(System.in);

System.out.print("Enter first number: ");

int num1 = sc.nextInt();

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

int num2 = sc.nextInt();

int result = add(num1, num2);

System.out.println("Sum (Return, With parameters): " + result);

sc.close();

}

}

**6) Arrays**

**Theory**: Arrays store multiple values of the same type in a single variable.

import java.util.Scanner;

public class ArrayInput {

public static void main(String[] args) {

int[] numbers = new int[5];

Scanner sc = new Scanner(System.in);

System.out.println("Enter 5 numbers:");

for (int i = 0; i < 5; i++) {

numbers[i] = sc.nextInt(); // Taking input

}

System.out.println("\nYou entered:");

for (int i = 0; i < 5; i++) {

System.out.print(numbers[i] + " ");

}

sc.close();

}

}