# 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

public class PrimitiveDataTypes {

public static void main(String[] args) {

int age = 25;

float height = 5.9f;

double weight = 68.5;

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

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

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

}

}

# 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

public class StudentInfo {

public static void main(String[] args) {

int id = 101;

String name = "Arun";

double marks = 89.5;

char grade = 'A';

System.out.println("Student ID: " + id);

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

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

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

}

}

# 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

public class Operators {

public static void main(String[] args) {

int n1 = 10;

int n2 = 20;

System.out.println("Addition: " + (n1 + n2));

System.out.println("Greater number: " + Math.max(n1, n2));

System.out.println("Are both positive? " + (n1 > 0 && n2 > 0));

}

1. 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.

public class Greeting {

public static void main(String[] args) {

String firstName = "Ravi";

String lastName = "Kumar";

System.out.println("Hello, " + firstName + " " + lastName + "! Welcome to the system.");

}

}

# 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

public class ReverseString {

public static void main(String[] args) {

String input = "Hello Java Learners";

StringBuilder sb = new StringBuilder(input);

System.out.println("Original: " + input);

System.out.println("Reversed: " + sb.reverse());

}

}

1. 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.

public class CharacterCount {

public static void main(String[] args) {

String text = "banana";

char ch = 'a';

int count = 0;

for (char c : text.toCharArray()) {

if (c == ch) count++;

}

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

}

}

# 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

import java.text.NumberFormat;

import java.time.LocalDate;

import java.time.format.DateTimeFormatter;

import java.util.Locale;

public class DateAndCurrency {

public static void main(String[] args) {

LocalDate today = LocalDate.now();

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

System.out.println("Current Date: " + today.format(formatter));

double amount = 12345.678;

NumberFormat currency = NumberFormat.getCurrencyInstance(new Locale("en", "IN"));

System.out.println("Formatted Amount: " + currency.format(amount));

}

}

# 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.

public class NumberSign {

public static void main(String[] args) {

int number = -5;

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.");

}

}

# Conditionspublic class NumberSign {

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

**Sample Input:**

Marks: 76

**Sample Output:**

Grade: B

public class GradeCheck {

public static void main(String[] args) {

int marks = 76;

if (marks >= 90) System.out.println("Grade: A");

else if (marks >= 80) System.out.println("Grade: B");

else if (marks >= 70) System.out.println("Grade: C");

else System.out.println("Grade: D");

}

}

# Switch

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

**Sample Input:**

Number1: 10

Number2: 5 Operation: \*

**Sample Output:**

Result: 50

public class Calculator {

public static void main(String[] args) {

int a = 10, b = 5;

char operation = '\*';

switch (operation) {

case '+': System.out.println("Result: " + (a + b)); break;

case '-': System.out.println("Result: " + (a - b)); break;

case '\*': System.out.println("Result: " + (a \* b)); break;

case '/': System.out.println("Result: " + (a / b)); break;

default: System.out.println("Invalid operation.");

}

}

}

# 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 \* 2; i += 2) {

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

}

}

}

# 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

public class ArrayAverage {

public static void main(String[] args) {

int[] numbers = {10, 20, 30, 40, 50};

int sum = 0;

for (int num : numbers) {

sum += num;

}

double avg = (double) sum / numbers.length;

System.out.println("Average: " + avg);

}

}

# 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 day = Day.MONDAY;

switch (day) {

case MONDAY:

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

break;

case FRIDAY:

System.out.println("Almost weekend!");

break;

case SUNDAY: System.out.println("Rest day!");

break;

default: System.out.println("Regular day.");

}

}

}

# 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

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 static voi main(String[] args) {

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

s.display();

}

}

# 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;

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 static void main(String[args) {

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

m.display();

}

}