

# 1. Primitive Data Types

```
import java.util.Scanner;

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

        System.out.print("Enter Age: ");
        int age = scanner.nextInt();
        System.out.print("Enter Height: ");
        double height = scanner.nextDouble();

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

        System.out.println("Age: " + age);
        System.out.println("Height: " + height);
        System.out.println("Weight: " + weight);

        scanner.close();
    }
}
```

## 2 Variables

```
import java.util.Scanner;

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

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

        int id = scanner.nextInt();

        scanner.nextLine(); // Consume newline left-over

        System.out.print("Enter Name: ");
```

```
String name = scanner.nextLine();

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

double marks = scanner.nextDouble();

scanner.nextLine();


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

char grade = scanner.nextLine().charAt(0);

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

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

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

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


scanner.close();

}

}
```

### 3 Operators

```
import java.util.Scanner;

public class OperatorsDemo {

    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);

        System.out.print("Enter Number1: ");

        int num1 = scanner.nextInt();

        System.out.print("Enter Number2: ");

        int num2 = scanner.nextInt();
```

```

// Arithmetic Operations

int sum = num1 + num2;

System.out.println("Addition: " + sum);


// Relational Operations (Ternary operator used for simplicity)

int greaterNumber = (num1 > num2) ? num1 : num2;

System.out.println("Greater number: " + greaterNumber);


// Logical Operations

boolean bothPositive = (num1 > 0) && (num2 > 0);

System.out.println("Are both positive? " + bothPositive);


scanner.close();

}

}

```

## 4. String Concatenation

```

import java.util.Scanner;

public class StringConcatenation {

    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);

        System.out.print("Enter First Name: ");

        String firstName = scanner.nextLine();

        System.out.print("Enter Last Name: ");

        String lastName = scanner.nextLine();
    }
}

```

```

        String greeting = "Hello, " + firstName + " " + lastName + "! Welcome to the system.";

        System.out.println(greeting);

        scanner.close();

    }

}

```

## 5. StringBuilder

```

import java.util.Scanner;

public class StringBuilderReverse {

    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);

        System.out.print("Input: ");

        String inputSentence = scanner.nextLine();

        StringBuilder sb = new StringBuilder(inputSentence);

        sb.reverse();

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

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

        scanner.close();

    }

}

```

## 6. String API

```

import java.util.Scanner;

public class CharacterCounter {

    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);

        System.out.print("String: ");
    }
}

```

```

String inputString = scanner.nextLine();

System.out.print("Character: ");

char targetChar = scanner.nextLine().charAt(0);

int count = 0;

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

    if (inputString.charAt(i) == targetChar) {

        count++;

    }

}

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

scanner.close();

}

}

```

## 7. Date, Time, and Numeric Objects

```

import java.time.LocalDate;

import java.time.format.DateTimeFormatter;

import java.text.NumberFormat;

import java.util.Locale;

import java.util.Scanner;

public class DateTimeAndCurrency {

    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);

        // Display current date formatted as DD-MM-YYYY
    }

}

```

```

LocalDate currentDate = LocalDate.now();

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

String formattedDate = currentDate.format(dateFormatter);

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


// Show formatted currency value for Indian Rupee (INR)

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

double amount = scanner.nextDouble();


NumberFormat currencyFormatter = NumberFormat.getCurrencyInstance(Locale.of("en",
"IN"));

String formattedAmount = currencyFormatter.format(amount);

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

scanner.close();

}

}

```

## 8. Flow Control

```

import java.util.Scanner;

public class NumberCheck {

    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);

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

        int number = scanner.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.");  
    }  
    scanner.close();  
}  
}
```

## 9. Conditions

```
import java.util.Scanner;  
  
public class GradeCalculator {  
    public static void main(String[] args) {  
        Scanner scanner = new Scanner(System.in);  
  
        System.out.print("Enter Marks: ");  
  
        int marks = scanner.nextInt();  
  
        char grade;  
  
        if (marks >= 90) {  
            grade = 'A';  
        } else if (marks >= 80) {  
            grade = 'B';  
        } else if (marks >= 70) {  
            grade = 'C';  
        } else if (marks >= 60) {  
            grade = 'D';  
        }  
    }  
}
```

```

    } else {
        grade = 'F';
    }

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

    scanner.close();
}
}

```

## 10. Switch

```

import java.util.Scanner;

public class SimpleCalculator {

    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);

        System.out.print("Enter Number1: ");

        double num1 = scanner.nextDouble();

        System.out.print("Enter Number2: ");

        double num2 = scanner.nextDouble();

        scanner.nextLine(); // Consume newline

        System.out.print("Enter Operation (+, -, *, /): ");

        char operation = scanner.nextLine().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 is not allowed.");

        }

        break;
    default:

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

        break;
}

scanner.close();
}
```

```
}
```

## 11. Loops and Branching

```
import java.util.Scanner;

public class EvenNumbers {

    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);

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

        int n = scanner.nextInt();

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

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

        }

        System.out.println();

        scanner.close();

    }

}
```

## 12. Arrays

```
import java.util.Scanner

public class ArrayAverage {

    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++) {

    System.out.print("Number " + (i + 1) + ": ");

    numbers[i] = scanner.nextInt();

    sum += numbers[i];

}

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

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

scanner.close();

}

}
```

## 13. Enum

```
import java.util.Scanner;

public class DayOfWeekEnum {

    public enum Day {

        MONDAY,

        TUESDAY,

        WEDNESDAY,

        THURSDAY,

        FRIDAY,

        SATURDAY,

        SUNDAY

    }

}
```

```
public static void main(String[] args) {

    Scanner scanner = new Scanner(System.in);

    System.out.print("Enter Day (e.g., MONDAY): ");

    String dayInput = scanner.nextLine().toUpperCase();

    try {

        Day day = Day.valueOf(dayInput);

        switch (day) {

            case MONDAY:

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

                break;

            case TUESDAY:

            case WEDNESDAY:

            case THURSDAY:

                System.out.println("Mid-week!");

                break;

            case FRIDAY:

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

                break;

            case SATURDAY:

            case SUNDAY:

                System.out.println("It's the weekend!");

                break;

        }

    } catch (IllegalArgumentException e) {
```

```

        System.out.println("Invalid day entered. Please use MONDAY, TUESDAY, etc.");
    }
    scanner.close();
}
}

```

## 14. OOPs Concepts

```

import java.util.Scanner;

class Student {
    String name;
    int marks;
    // Constructor
    public Student(String name, int marks) {
        this.name = name;
        this.marks = marks;
    }
    public void displayStudentData() {
        System.out.println("Student Name: " + name);
        System.out.println("Marks: " + marks);
    }
}

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

```

```
System.out.print("Enter Name: ");

String studentName = scanner.nextLine();

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

int studentMarks = scanner.nextInt();

Student student1 = new Student(studentName, studentMarks); // Create an object

student1.displayStudentData(); // Display its data

scanner.close();

}

}
```

## 15. Inheritance

```
import java.util.Scanner;

class Employee {

    String name;

    double salary;

    public Employee(String name, double salary) {

        this.name = name;

        this.salary = salary;

    }

    public void displayEmployeeInfo() {

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

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

    }

}
```

```
class Manager extends Employee {  
    String department;  
    public Manager(String name, double salary, String department) {  
        super(name, salary); // Call superclass constructor  
        this.department = department;  
    }  
    @Override  
    public void displayEmployeeInfo() { // Overriding for Manager  
        super.displayEmployeeInfo(); // Call superclass method  
        System.out.println("Department: " + department);  
    }  
}
```

```
public class InheritanceDemo {  
    public static void main(String[] args) {  
        Scanner scanner = new Scanner(System.in);  
        System.out.print("Enter Name: ");  
        String managerName = scanner.nextLine();  
        System.out.print("Enter Salary: ");  
        double managerSalary = scanner.nextDouble();  
        scanner.nextLine(); // Consume newline  
        System.out.print("Enter Department: ");  
        String managerDepartment = scanner.nextLine();  
        Manager manager = new Manager(managerName, managerSalary, managerDepartment);  
        manager.displayEmployeeInfo();  
    }  
}
```

```
scanner.close();
```

```
}
```

```
}
```