

# Bienvenidos al Curso de Introducción a Java

Java es uno de los lenguajes de programación más populares y poderosos del mundo. Utilizado en una amplia variedad de aplicaciones, desde desarrollo web hasta aplicaciones móviles y sistemas embebidos, dominar Java abre un mundo de oportunidades en la programación.

En este curso, exploraremos desde los conceptos básicos hasta las técnicas avanzadas de Java. Aprenderemos cómo estructurar programas Java, trabajar con objetos y clases, manipular datos, y resolver problemas utilizando algoritmos eficientes. Además, nos sumergiremos en temas fundamentales como la programación orientada a objetos, manejo de excepciones, y desarrollo de aplicaciones interactivas.

Ya sea que seas un principiante absoluto en la programación o un desarrollador experimentado buscando expandir sus habilidades, este curso te proporcionará las bases sólidas y las habilidades prácticas necesarias para desarrollar aplicaciones robustas en Java.

¡Prepárate para embarcarte en un emocionante viaje de aprendizaje con Java!

EJERCICIOS:

## Conceptos Básicos de Java

Ejercicio 1: Hola Mundo:

Crea un programa que imprima "Hola Mundo" en la consola. Objetivos: Familiarizarse con la estructura básica de un programa Java.

Variables y Tipos de Datos:

Declara variables de diferentes tipos (int, double, boolean, char, String) y asígnales valores. Imprime los valores en la consola. Objetivos: Entender la declaración y el uso de diferentes tipos de datos

## Control de Flujo

Estructura Condicional:

Crea un programa que pida al usuario un número y determine si es positivo, negativo o cero. Objetivos: Practicar el uso de if, else if, y else.

Estructura de Repetición:

Crea un programa que imprima los números del 1 al 10 utilizando un bucle for. Objetivos: Familiarizarse con la estructura de los bucles for.

Bucle while:

Crea un programa que imprima los números del 10 al 1 usando un bucle while. Objetivos: Comprender el uso de bucles while.

## Arrays y Matrices

Arrays Unidimensionales:

Declara un array de enteros con 5 elementos, asígnale valores, e imprímelos en la consola. Objetivos: Trabajar con arrays y entender cómo acceder a sus elementos.

Matrices:

Declara una matriz (array bidimensional) de enteros 3x3, asígnale valores e imprime la matriz. Objetivos: Aprender a manejar matrices en Java.

## Métodos y Funciones

Crear un Método:

Crea un método que tome dos números enteros como parámetros y devuelva su suma. Objetivos: Entender la creación y el uso de métodos.

Sobrecarga de Métodos:

Crea métodos sobrecargados para sumar dos, tres y cuatro números. Objetivos: Aprender sobre la sobrecarga de métodos.

## 5. Programación Orientada a Objetos (POO)

Clases y Objetos:

Define una clase Coche con atributos como marca, modelo y año. Crea un objeto de la clase Coche e imprime sus atributos. Objetivos: Entender la creación de clases y objetos.

Herencia:

Crea una clase Vehículo con atributos básicos y una clase Moto que herede de Vehículo. Añade un atributo específico para Moto y muestra su uso. Objetivos: Practicar la herencia en Java.

## Excepciones y Manejo de Errores

Manejo de Excepciones:

Crea un programa que intente dividir un número por cero y maneje la excepción generada. Objetivos: Comprender cómo manejar excepciones en Java.

## Colecciones

Listas:

Crea una ArrayList de cadenas, agrega algunos nombres, e imprime la lista. Objetivos: Aprender a usar las colecciones en Java.

## Entrada/Salida

Leer Entrada del Usuario:

Crea un programa que pida al usuario su nombre y lo salude con su nombre. Objetivos: Practicar la lectura de la entrada del usuario desde la consola.

## **Proyectos**

Proyecto Final: Crea una pequeña aplicación que permita gestionar una lista de tareas. Debe poder agregar, eliminar y mostrar tareas. Objetivos: Integrar todos los conceptos aprendidos en un proyecto sencillo.

**Estos ejercicios te proporcionarán una base sólida en Java.**

## RESPUESTAS:

### Conceptos Básicos de Java:

#### Hola Mundo:

```
public class HolaMundo {  
    public static void main(String[]  
args) {  
        System.out.println("Hola  
Mundo");  
    }  
}
```

#### Variables y Tipos de Datos:

```
public class VariablesTipos {  
    public static void main(String[]  
args) {  
        int numero = 10;  
        double decimal = 20.5;  
        boolean esVerdadero = true;  
        char caracter = 'A';  
        String texto = "Hola";  
  
        System.out.println("Entero:  
" + numero);  
        System.out.println("Decimal:  
" + decimal);  
  
        System.out.println("Booleano: " +  
esVerdadero);  
  
        System.out.println("Caracter: " +  
caracter);  
        System.out.println("Texto: "  
+ texto);  
    }  
}
```

## CONTROL DE FLUJO:

### Estructura Condicional:

```
import java.util.Scanner;

public class Condicional {
    public static void main(String[]
args) {
        Scanner scanner = new
Scanner(System.in);
        System.out.print("Ingresa un
número: ");
        int numero =
scanner.nextInt();

        if (numero > 0) {
            System.out.println("El
número es positivo");
        } else if (numero < 0) {
            System.out.println("El
número es negativo");
        } else {
            System.out.println("El
número es cero");
        }
    }
}
```

## Estructura de Repetición

```
public class BucleFor {  
    public static void main(String[]  
args) {  
        for (int i = 1; i <= 10; i+  
+) {  
            System.out.println(i);  
        }  
    }  
}
```

### Bucle While

```
public class BucleWhile {  
    public static void main(String[]  
args) {  
        int i = 10;  
        while (i > 0) {  
            System.out.println(i);  
            i--;  
        }  
    }  
}
```

## ARRAYS Y MATRICES

### Arrays Unidimensionales

```
public class Arrays {  
    public static void main(String[]  
args) {  
        int[] numeros = {1, 2, 3, 4,  
5};  
        for (int i = 0; i <  
numeros.length; i++) {  
            System.out.println(numeros[i]);  
        }  
    }  
}
```

## Matrices

```
public class Matrices {  
    public static void main(String[]  
args) {  
        int[][] matriz = {  
            {1, 2, 3},  
            {4, 5, 6},  
            {7, 8, 9}  
        };  
  
        for (int i = 0; i <  
matriz.length; i++) {  
            for (int j = 0; j <  
matriz[i].length; j++) {  
  
                System.out.print(matriz[i][j] + "  
");  
            }  
            System.out.println();  
        }  
    }  
}
```

## METODOS Y FUNCIONES

### Crear un Método

```
public class Metodos {  
    public static void main(String[]  
args) {  
        int suma = sumar(5, 10);  
        System.out.println("La suma  
es: " + suma);  
    }  
  
    public static int sumar(int a,  
int b) {  
        return a + b;  
    }  
}
```

## Sobrecarga de Métodos

```
public class Sobrecarga {  
    public static void main(String[]  
args) {  
        System.out.println(sumar(1,  
2));  
        System.out.println(sumar(1,  
2, 3));  
        System.out.println(sumar(1,  
2, 3, 4));  
    }  
  
    public static int sumar(int a,  
int b) {  
        return a + b;  
    }  
  
    public static int sumar(int a,  
int b, int c) {  
        return a + b + c;  
    }  
  
    public static int sumar(int a,  
int b, int c, int d) {  
        return a + b + c + d;  
    }  
}
```



# PROGRAMACION ORIENTADA A OBJETOS (POO)

## Clases y Objetivos

```
class Coche {
    String marca;
    String modelo;
    int año;

    Coche(String marca, String
modelo, int año) {
        this.marca = marca;
        this.modelo = modelo;
        this.año = año;
    }

    void mostrarInfo() {
        System.out.println("Marca: "
+ marca);
        System.out.println("Modelo:
" + modelo);
        System.out.println("Año: " +
año);
    }
}

public class Main {
    public static void main(String[]
args) {
        Coche miCoche = new
Coche("Toyota", "Corolla", 2020);
        miCoche.mostrarInfo();
    }
}
```

## Herencia

```
class Vehiculo {  
  
    String marca;  
  
    int año;  
  
    Vehiculo(String marca, int año) {  
  
        this.marca = marca;  
  
        this.año = año;  
  
    }  
  
    void mostrarInfo() {  
  
        System.out.println("Marca: " + marca);  
  
        System.out.println("Año: " + año);  
  
    }  
}  
  
class Moto extends Vehiculo {  
  
    boolean tieneSidecar;  
  
    Moto(String marca, int año, boolean tieneSidecar) {  
  
        super(marca, año);  
  
        this.tieneSidecar = tieneSidecar;  
  
    }  
}
```

**@Override**

```
void mostrarInfo() {  
    super.mostrarInfo();  
    System.out.println("Tiene sidecar: " + tieneSidecar);  
}  
}
```

```
public class Main {  
    public static void main(String[] args) {  
        Moto miMoto = new Moto("Harley-Davidson", 2021, true);  
        miMoto.mostrarInfo();  
    }  
}
```

# EXCEPCIONES Y MANEJO DE ERRORES

## Manejo de Excepciones

```
public class ManejoExcepciones {  
    public static void main(String[]  
args) {  
        try {  
            int resultado =  
dividir(10, 0);  
  
System.out.println("Resultado: " +  
resultado);  
        } catch (ArithmeticException  
e) {  
  
System.out.println("Error: División  
por cero no permitida.");  
        }  
    }  
  
    public static int dividir(int a,  
int b) {  
        return a / b;  
    }  
}
```

## COLECCIONES

### Listas

```
import java.util.ArrayList;

public class Listas {
    public static void main(String[]
args) {
        ArrayList<String> nombres =
new ArrayList<>();
        nombres.add("Ana");
        nombres.add("Juan");
        nombres.add("Pedro");

        for (String nombre :
nombres) {

            System.out.println(nombre);
        }
    }
}
```

## ENTRADA/SALIDA

### Leer Entrada del Usuario

```
import java.util.Scanner;

public class EntradaUsuario {
    public static void main(String[]
args) {
        Scanner scanner = new
Scanner(System.in);
        System.out.print("¿Cuál es
tu nombre? ");
        String nombre =
scanner.nextLine();
        System.out.println("Hola, "
+ nombre);
    }
}
```

# **PROYECTOS**

## **Proyecto Final**

```
import java.util.ArrayList;
```

```
import java.util.Scanner;
```

```
class Tarea {
```

```
    String descripcion;
```

```
    Tarea(String descripcion) {
```

```
        this.descripcion = descripcion;
```

```
    }
```

```
    @Override
```

```
    public String toString() {
```

```
        return descripcion;
```

```
    }
```

```
}
```

```
public class ListaTareas {
```

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

```
        ArrayList<Tarea> tareas = new ArrayList<>();
```

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

```
        int opcion;
```

```
do {  
  
    System.out.println("1. Agregar tarea");  
  
    System.out.println("2. Eliminar tarea");  
  
    System.out.println("3. Mostrar tareas");  
  
    System.out.println("4. Salir");  
  
    System.out.print("Elige una opción: ");  
  
    opcion = scanner.nextInt();  
  
    scanner.nextLine(); // Consumir la nueva línea  
  
    switch (opcion) {  
  
        case 1:  
  
            System.out.print("Descripción de la tarea: ");  
  
            String descripcion = scanner.nextLine();  
  
            tareas.add(new Tarea(descripcion));  
  
            break;  
  
        case 2:  
  
            System.out.print("Número de la tarea a eliminar: ");  
  
            int indice = scanner.nextInt();  
  
            if (indice >= 0 && indice < tareas.size()) {  
  
                tareas.remove(indice);  
  
            } else {  
  
                System.out.println("Índice no válido");  
  
            }  
  
            break;
```

**case 3:**

```
System.out.println("Tareas:");
```

```
for (int i = 0; i < tareas.size(); i++) {
```

```
    System.out.println(i + ". " + tareas.get(i));
```

```
}
```

```
break;
```

**case 4:**

```
System.out.println("Saliendo...");
```

```
break;
```

**default:**

```
System.out.println("Opción no válida");
```

```
}
```

```
} while (opcion != 4);
```

```
}
```

```
}
```