

## PROGRAMACIÓN II

### Trabajo Práctico 1: Introducción a Java

#### OBJETIVO GENERAL

Aplicar los conocimientos adquiridos sobre la instalación y configuración del entorno de desarrollo, manipulación de datos, operadores matemáticos y depuración de código en Java, mediante ejercicios prácticos introductorios.

#### MARCO TEÓRICO

Concepto	Aplicación en el proyecto
Instalación y entorno	Almacenan el conjunto de países
Variables y tipos de datos	Representan los datos de cada país (nombre, población, superficie, etc.)
Entrada y salida	Separan las operaciones: carga, búsqueda, estadísticas, ordenamientos
Operadores aritméticos	Aplican filtros y validaciones según criterios
Caracteres especiales	Permite ordenar países por población, nombre, superficie, etc.
Expresiones e instrucciones	Permiten obtener indicadores clave del dataset
Tipos de datos y conversiones	Lectura del dataset desde un archivo CSV
Debugging y errores comunes	Identificación y corrección de errores de compilación.
Pruebas de escritorio	Análisis paso a paso de ejecución de código.

## CASO PRÁCTICO

El trabajo consiste en resolver una serie de ejercicios introductorios en Java que permitan:

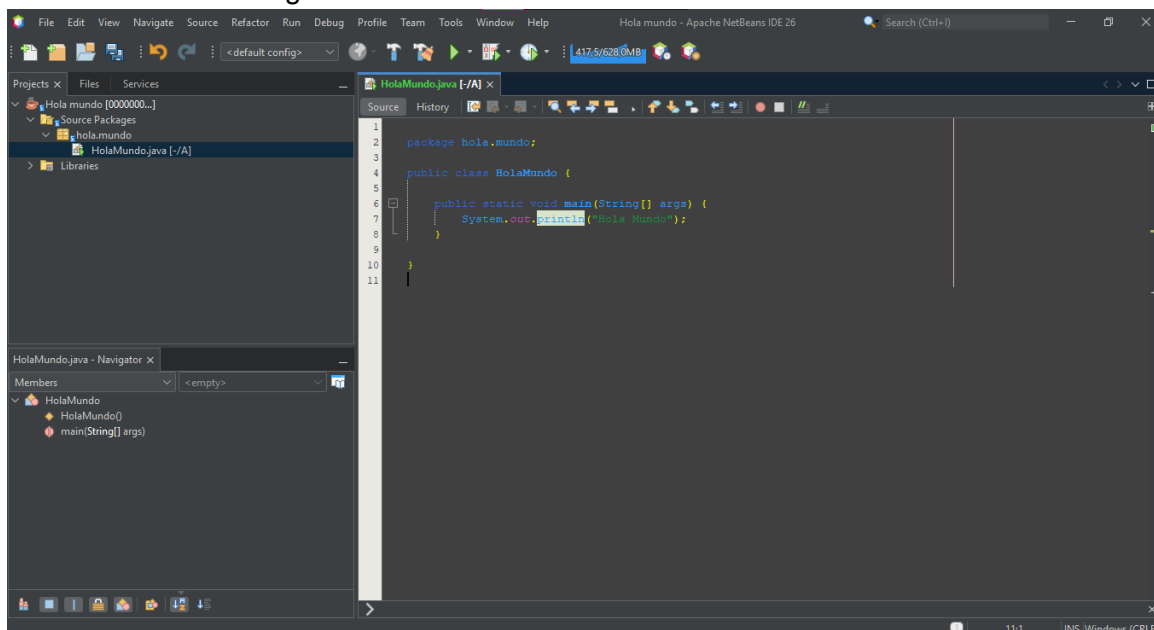
- Configurar correctamente el entorno de desarrollo (Java JDK y NetBeans).
- Crear programas básicos que imprimen mensajes en consola.
- Declarar variables de distintos tipos y manipular sus valores.
- Leer datos ingresados por el usuario usando [Scanner](#).
- Realizar operaciones aritméticas básicas.
- Aplicar caracteres de escape para dar formato a la salida.
- Analizar diferencias entre expresiones e instrucciones.
- Detectar y corregir errores simples en el código.
- Comprender el comportamiento del lenguaje mediante pruebas de escritorio.

1. Verificar que tienes instalado Java JDK y NetBeans
  - a. Confirma que tienes Java JDK instalado ejecutando el siguiente comando en la terminal: `java -versión`

```
C:\Users\PC ONE>java -version
java version "21.0.7" 2025-04-15 LTS
Java(TM) SE Runtime Environment (build 21.0.7+8-LTS-245)
Java HotSpot(TM) 64-Bit Server VM (build 21.0.7+8-LTS-245, mixed mode, sharing)

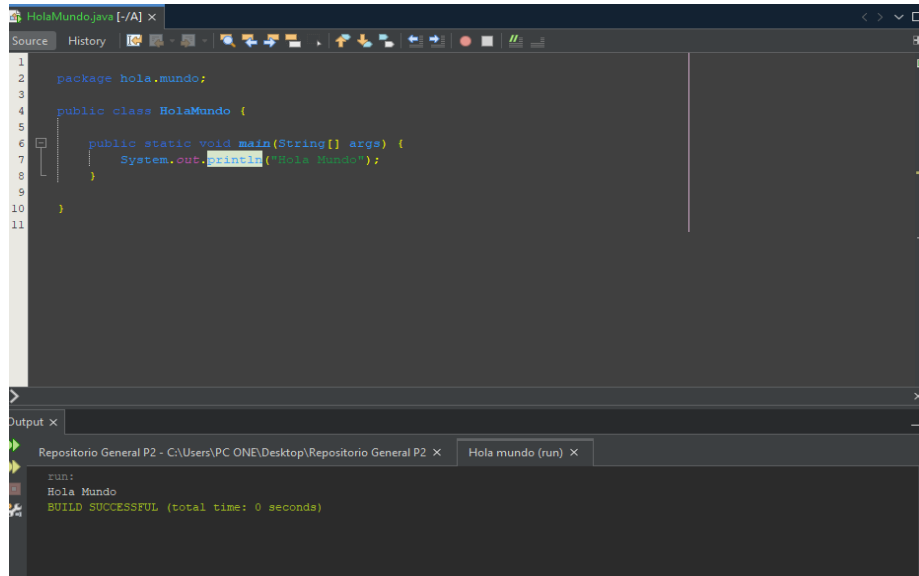
C:\Users\PC ONE>
```

- b. Abre NetBeans, crea un nuevo proyecto y configura el modo oscuro.
- c. Toma una captura de pantalla del entorno configurado y agrégala a tu entrega.



## 2. Escribir y ejecutar un programa básico en Java.

- Creá una clase llamada **HolaMundo**.
- Escribe un programa que imprima el mensaje: **¡Hola, Java!**
- Ejecuta el programa en NetBeans y adjunta una captura del resultado en la consola.



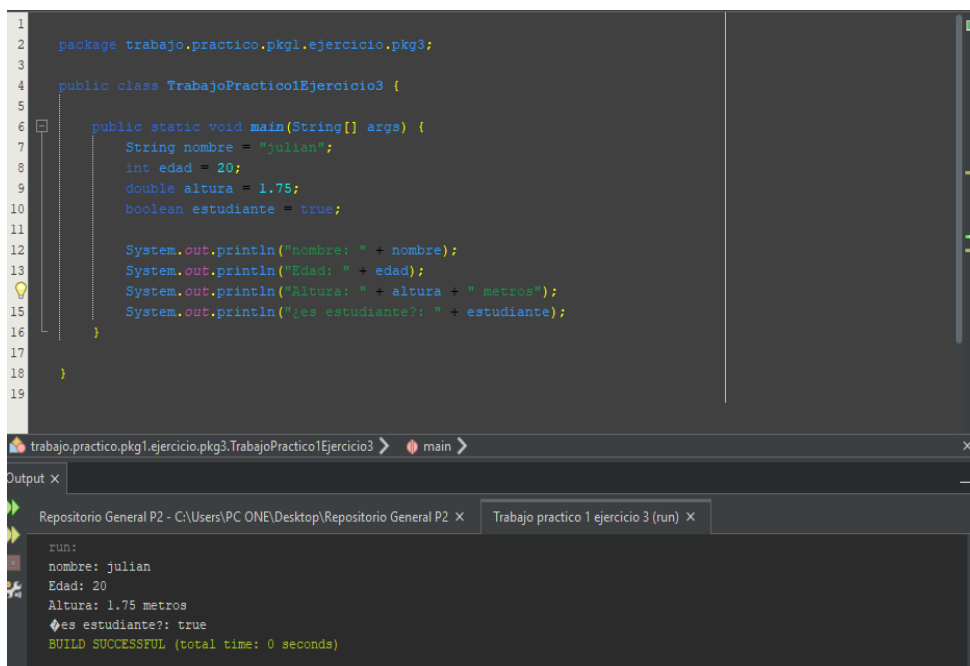
```
1 package hola.mundo;
2
3
4 public class HolaMundo {
5
6     public static void main(String[] args) {
7         System.out.println("Hola Mundo");
8     }
9
10 }
11
```

run:  
Hola Mundo  
BUILD SUCCESSFUL (total time: 0 seconds)

## 3. Crea un programa que declare las siguientes variables con valores asignados:

- String nombre
- int edad
- double altura
- boolean estudiante

Imprime los valores en pantalla usando **System.out.println()**.



```
1 package trabajo.practico.pkg1.ejercicio.pkg3;
2
3
4 public class TrabajoPractico1Ejercicio3 {
5
6     public static void main(String[] args) {
7         String nombre = "julian";
8         int edad = 20;
9         double altura = 1.75;
10        boolean estudiante = true;
11
12        System.out.println("nombre: " + nombre);
13        System.out.println("Edad: " + edad);
14        System.out.println("Altura: " + altura + " metros");
15        System.out.println("¿es estudiante?: " + estudiante);
16    }
17
18 }
19
```

run:  
nombre: julian  
Edad: 20  
Altura: 1.75 metros  
¿es estudiante?: true  
BUILD SUCCESSFUL (total time: 0 seconds)

4. Escribe un programa que solicite al usuario ingresar su nombre y edad, y luego los muestre en pantalla. Usa **Scanner** para capturar los datos.

```
3 import java.util.Scanner;
4 import java.io.PrintStream;
5 import java.io.UnsupportedEncodingException;
6
7 public class TrabajoPractico1Ejercicio4 {
8
9     public static void main(String[] args) throws UnsupportedEncodingException {
10
11         System.setOut(new PrintStream(System.out, true, "UTF-8"));
12
13         Scanner scanner = new Scanner(System.in, "UTF-8");
14
15         System.out.print("Ingresa tu nombre: ");
16         String nombre = scanner.nextLine();
17
18         System.out.print("Ingresa tu edad: ");
19         int edad = scanner.nextInt();
20
21         System.out.println("Tu nombre es: " + nombre);
22         System.out.println("Tu edad es: " + edad + " años");
23
24         scanner.close();
25     }
26 }
```

Output X

Repositorio General P2 - C:\Users\PC ONE\Desktop\Repositorio General P2 X Trabajo practico 1 ejercicio 4 (run) X

run:

Ingresa tu nombre: a

Ingresa tu edad: 2

Tu nombre es: a

Tu edad es: 2 años

BUILD SUCCESSFUL (total time: 3 seconds)

5. Escribe un programa que solicite dos números enteros y realice las siguientes operaciones:
- Suma
  - Resta
  - Multipliación
  - División

Muestra los resultados en la consola.

Repositorio General P2 - C:\Users\PC ONE\Desktop\Repositorio General P2 X Trabajo practico 1 ejercicio 5 (run) X

run:

Ingresa el primer número: 1

Ingresa el segundo número: 2

Resultados:

Suma: 3

Resta: -1

Multipliación: 2

División: 0.5

BUILD SUCCESSFUL (total time: 6 seconds)

6. Escribe un programa que muestre el siguiente mensaje en consola:

**Nombre: Juan Pérez**

**Edad: 30 años**

**Dirección: "Calle Falsa 123"**

Usa caracteres de escape (`\n`, `\"`) en `System.out.println()`.

```
package trabajopracticoejercicio6;

import java.io.PrintStream;
import java.io.UnsupportedEncodingException;

public class TrabajoPracticoEjercicio6 {

    public static void main(String[] args) throws UnsupportedEncodingException {
        // Forzar UTF-8 en la consola
        System.setOut(new PrintStream(System.out, true, "UTF-8"));

        System.out.println("Nombre: Juan Pérez Edad: 30 años\nDirección: \"Calle Falsa 123\"");
    }
}
```

run:  
Nombre: Juan Pérez Edad: 30 años  
Dirección: "Calle Falsa 123"  
BUILD SUCCESSFUL (total time: 0 seconds)

7. Analiza el siguiente código y responde: ¿Cuáles son expresiones y cuáles son instrucciones? Explica la diferencia en un breve párrafo.

`int x = 10;` // Línea 1

`x = x + 5;` // Línea 2

`System.out.println(x);` // Línea 3

`int x = 10;` → Instrucción

Es una declaración de variable con asignación.  
Contiene la expresión 10 (el valor que se asigna a x).

`x = x + 5;` → Instrucción

Es una asignación.  
Dentro de esta instrucción, `x + 5` es una expresión, porque calcula un valor.

`System.out.println(x);` → Instrucción

Es una llamada a un método, que ejecuta una acción (imprimir en consola).  
La expresión dentro del paréntesis es `x`, que simplemente representa el valor de la variable.

8. Manejar conversiones de tipo y división en Java.
  - a. Escribe un programa que divida dos números enteros ingresados por el usuario.

```
package trabajopracticoejercicio8;

import java.util.Scanner;
import java.io.PrintStream;
import java.io.UnsupportedEncodingException;

public class TrabajoPractico1Ejercicio8 {

    public static void main(String[] args) throws UnsupportedEncodingException {
        System.setOut(new PrintStream(System.out, true, "UTF-8"));
        Scanner scanner = new Scanner(System.in, "UTF-8");

        System.out.print("Ingresa el primer número (int): ");
        int num1 = scanner.nextInt();

        System.out.print("Ingresa el segundo número (int): ");
        int num2 = scanner.nextInt();

        if (num2 != 0) {
            int resultado = num1 / num2; // División entera
            System.out.println("Resultado de la división entera: " + resultado);
        } else {
            System.out.println("No se puede dividir entre 0");
        }

        scanner.close();
    }
}
```

- b. Modifica el código para usar **double** en lugar de **int** y compara los resultados.

```
package trabajopracticoejercicio8;

import java.util.Scanner;
import java.io.PrintStream;
import java.io.UnsupportedEncodingException;

public class TrabajoPractico1Ejercicio8 {

    public static void main(String[] args) throws UnsupportedEncodingException {
        // Forzar UTF-8 en la consola
        System.setOut(new PrintStream(System.out, true, "UTF-8"));
        Scanner scanner = new Scanner(System.in, "UTF-8");

        System.out.print("Ingresa el primer número (double): ");
        double num1 = scanner.nextDouble();

        System.out.print("Ingresa el segundo número (double): ");
        double num2 = scanner.nextDouble();

        if (num2 != 0) {
            double resultado = num1 / num2; // División decimal
            System.out.println("Resultado de la división decimal: " + resultado);
        } else {
            System.out.println("No se puede dividir entre 0");
        }

        scanner.close();
    }
}
```

La diferencia es que int/int trunca la parte decimal del resultado mientras que double/double no lo hace dando mas precisión en la división.

9. Corrige el siguiente código para que funcione correctamente. Explica qué error tenía y cómo lo solucionaste.

```
import java.util.Scanner;

public class ErrorEjemplo {

    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);

        System.out.print("Ingresa tu nombre: ");

        String nombre = scanner.nextInt(); // ERROR

        System.out.println("Hola, " + nombre);

    }

}
```

**Corrección de código y explicación:**

```
import java.util.Scanner;

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

        System.out.print("Ingresa tu nombre: ");
        String nombre = scanner.nextLine(); // CORREGIDO: usar nextLine() para
texto

        System.out.println("Hola, " + nombre);

        scanner.close();
    }
}
```

El problema del código era que en la línea 6 usaba `scanner.nextInt()` para leer un nombre, el cual es un string y `nextInt` solo sirve para números enteros (Int). Para arreglar eso se cambio `scanner.nextInt()` por `scanner.nextLine()`.

También se cerró el scanner con **`scanner.close()`** por buena práctica .

10. Completa la tabla de prueba de escritorio para el siguiente código. ¿Cuál es el valor de **resultado** y por qué?

```
public class PruebaEscritorio {  
  
    public static void main(String[] args) {  
  
        int a = 5;  
  
        int b = 2;  
  
        int resultado = a / b;  
  
        System.out.println("Resultado: " + resultado);  
    }  
}
```

Línea	Variable	Valor	Comentario
3	a	5	Se asigna 5 a la variable a.
4	b	2	Se asigna 2 a la variable b.
5	Resultado	2	División entera: $5 / 2 = 2$ (los decimales se truncan).
6	—	—	Se imprime en consola: "Resultado: 2".

Explicación:

- a y b son enteros (int), por lo que la división  $a / b$  también es entera.
- Aunque  $5 / 2 = 2.5$ , al ser una división entre enteros, los decimales se eliminan, dejando el resultado como 2.
- Por eso, la consola muestra: Resultado: 2