

# 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

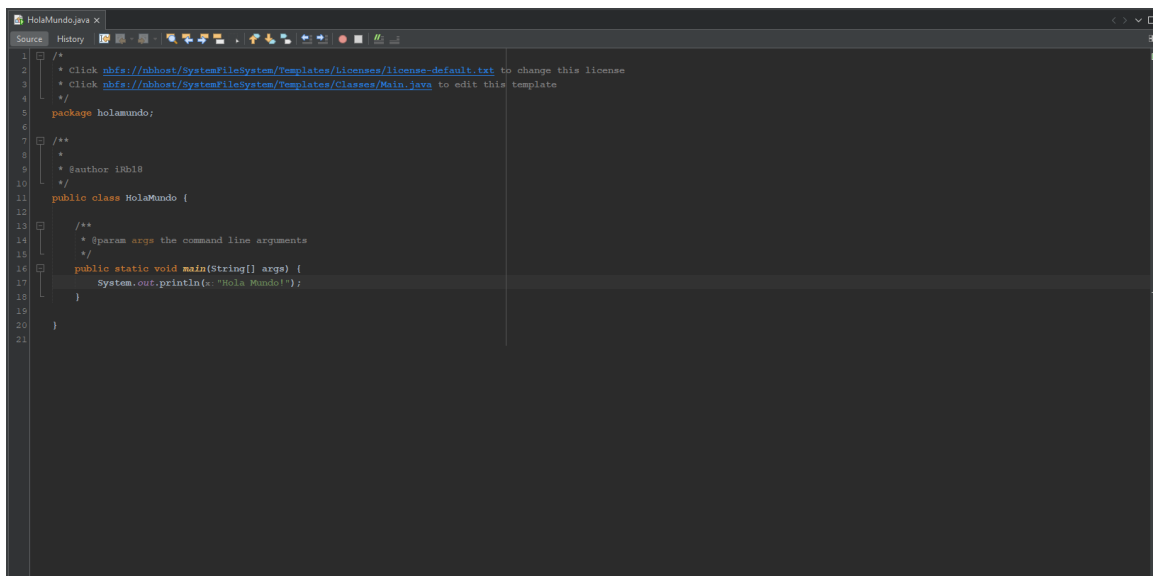
El trabajo consiste en resolver una serie de ejercicios introductorios en Java que permitan:

- 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**

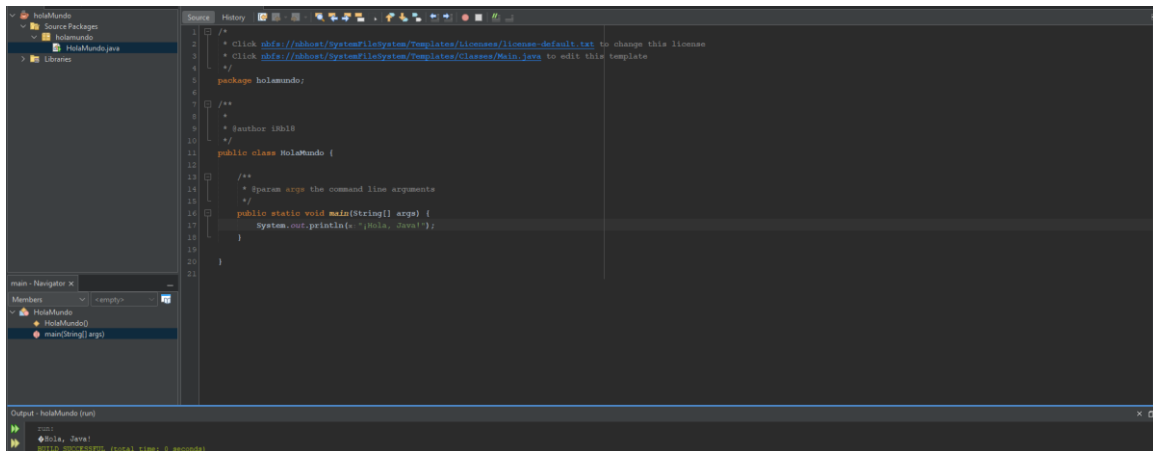
```
C:\Users\iRb18>java -version
java version "21.0.8" 2025-07-15 LTS
Java(TM) SE Runtime Environment (build 21.0.8+12-LTS-250)
Java HotSpot(TM) 64-Bit Server VM (build 21.0.8+12-LTS-250, mixed mode, sharing)
```

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

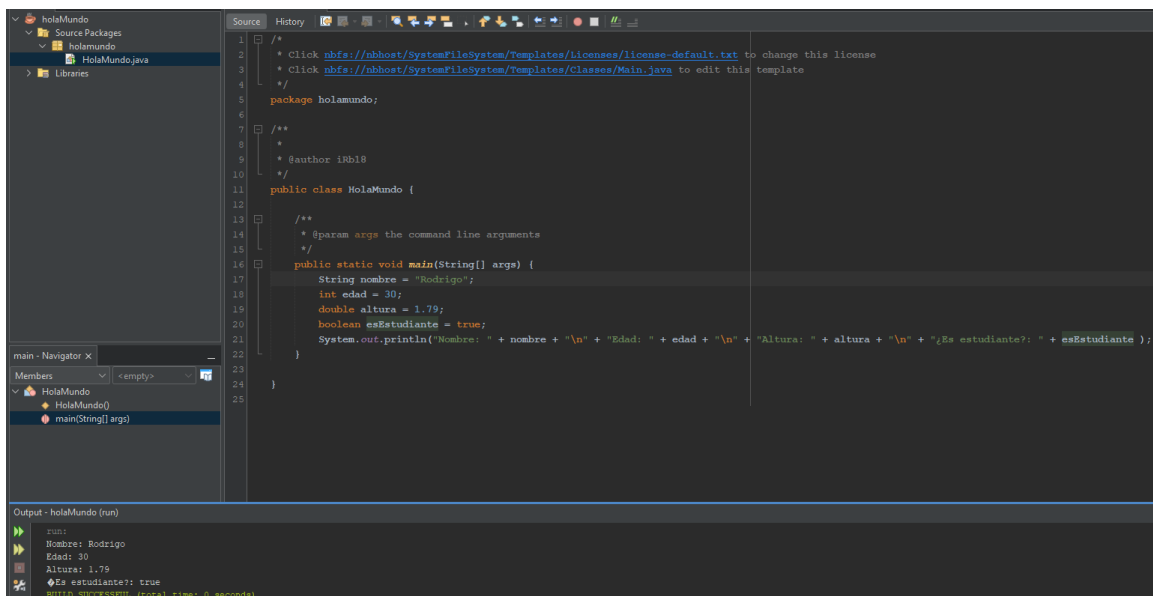
consola.



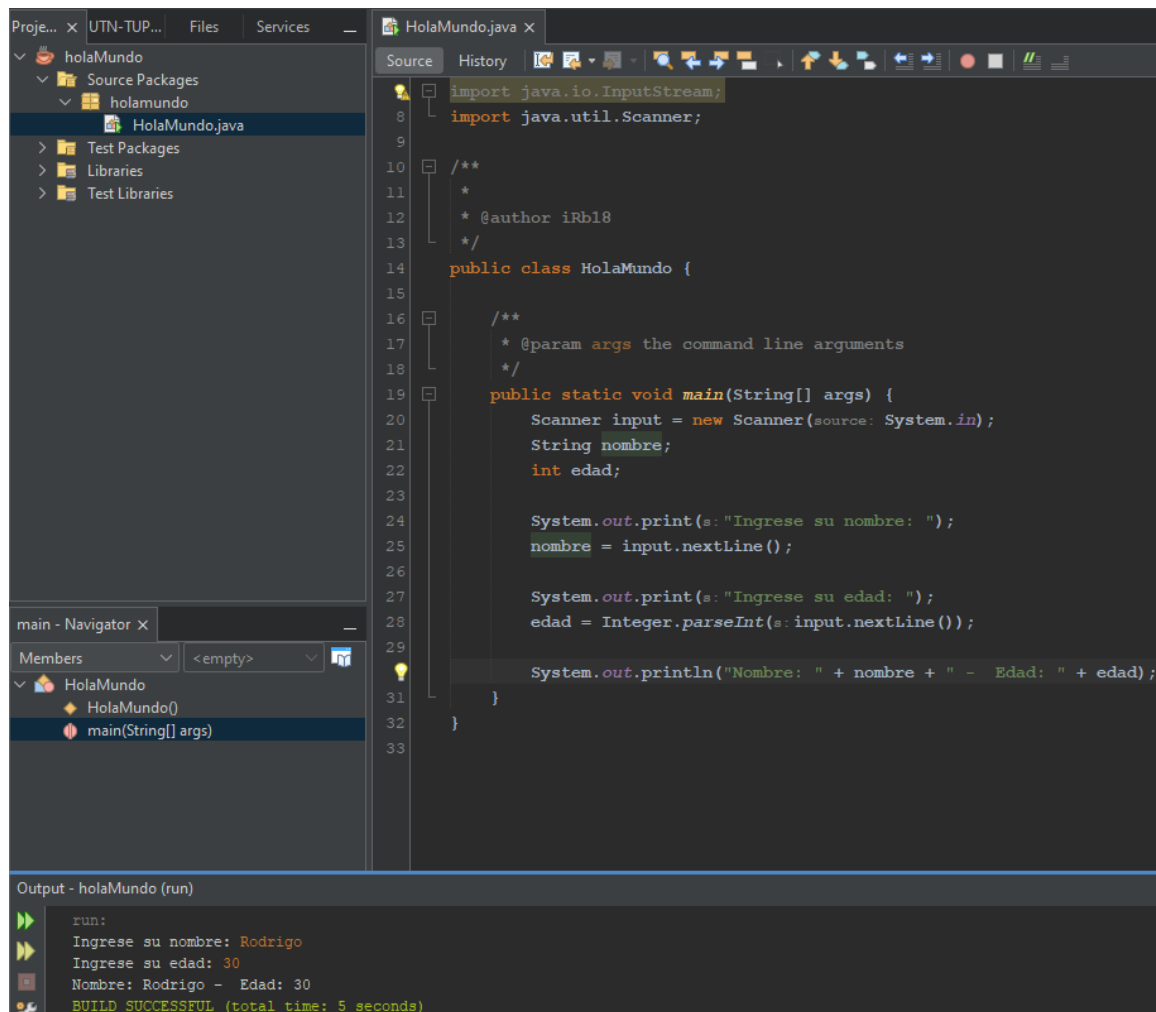
3. Crea un programa que declare las siguientes variables con valores asignados:

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

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



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.



```
8 import java.io.InputStream;
9 import java.util.Scanner;
10
11 /**
12  * @author iRb18
13  */
14 public class HolaMundo {
15
16     /**
17      * @param args the command line arguments
18      */
19     public static void main(String[] args) {
20         Scanner input = new Scanner(System.in);
21         String nombre;
22         int edad;
23
24         System.out.print(s: "Ingrese su nombre: ");
25         nombre = input.nextLine();
26
27         System.out.print(s: "Ingrese su edad: ");
28         edad = Integer.parseInt(s: input.nextLine());
29
30         System.out.println("Nombre: " + nombre + " - Edad: " + edad);
31     }
32 }
33
```

Output - holaMundo (run)

```
run:
Ingrese su nombre: Rodrigo
Ingrese su edad: 30
Nombre: Rodrigo - Edad: 30
BUILD SUCCESSFUL (total time: 5 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.

```
13  */
14  public class HolaMundo {
15
16      /**
17       * @param args the command line arguments
18       */
19      public static void main(String[] args) {
20          Scanner input = new Scanner(System.in);
21          int a, b, suma, resta, multiplicacion;
22          double division;
23
24          System.out.print("Primer número: ");
25          a = Integer.parseInt(input.nextLine());
26
27          System.out.print("Segundo número: ");
28          b = Integer.parseInt(input.nextLine());
29
30          suma = a + b;
31          resta = a - b;
32          multiplicacion = a * b;
33          division = ((double) a / b);
34
35          System.out.println("Suma: " + suma);
36          System.out.println("Resta: " + resta);
37          System.out.println("Multiplicación: " + multiplicacion);
38          System.out.println("División: " + division);
39      }
40  }
```

Output - holaMundo (run)

```
run:
Primer número: 5
Segundo número: 3
Suma: 8
Resta: 2
Multiplicación: 15
División: 1.6666666666666667
BUILD SUCCESSFUL (total time: 3 seconds)
```

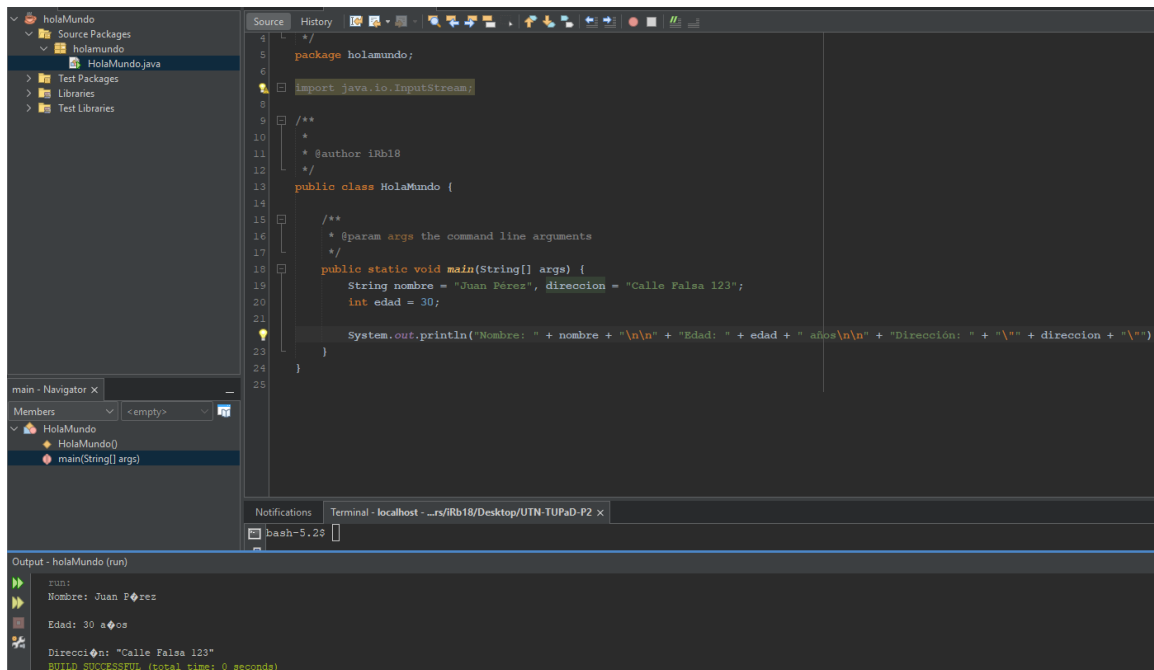
6. Escribe un programa que muestre el siguiente mensaje en consola:

**Nombre: Juan Pérez**

**Edad: 30 años**

**Dirección: "Calle Falsa 123"**

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



```
1  /*
2  3  package holamundo;
4
5  import java.io.InputStream;
6
7
8  /**
9   *
10  * @author iRb18
11  */
12
13  public class HolaMundo {
14
15      /**
16       * @param args the command line arguments
17       */
18      public static void main(String[] args) {
19          String nombre = "Juan Pérez", direccion = "Calle Falsa 123";
20          int edad = 30;
21
22          System.out.println("Nombre: " + nombre + "\n" + "Edad: " + edad + " años\n" + "Dirección: " + "\"" + direccion + "\"");
23      }
24  }
25
```

Output - HolaMundo (run)

```
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`

→ Instrucción

`x = x + 5; // Línea 2`

→ Expresión

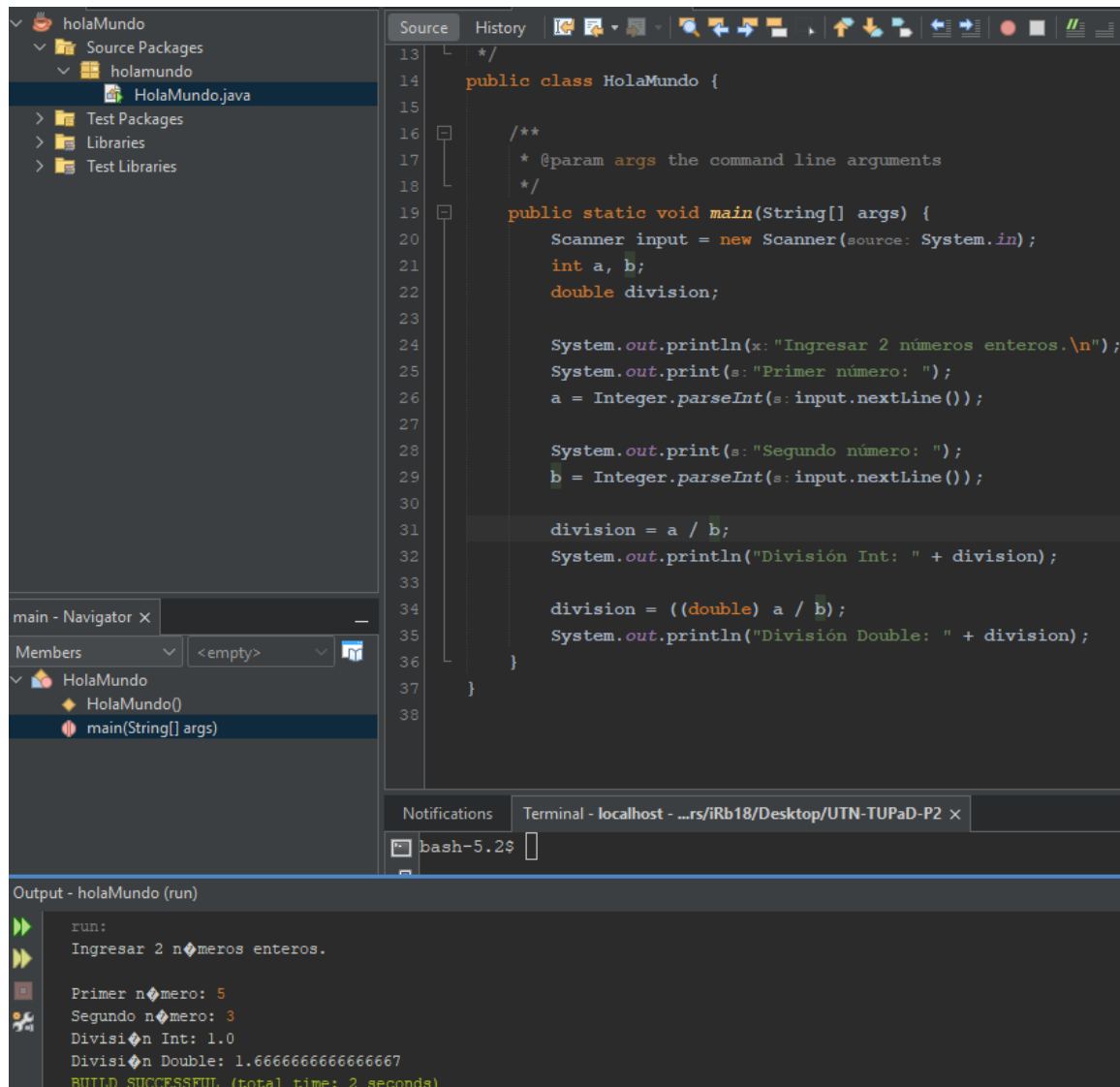
`System.out.println(x); // Línea 3`

→ Instrucción

- Una expresión es un fragmento de código que produce un valor, en cambio una instrucción es una línea de código que realiza una acción.

8. Manejar conversiones de tipo y división en Java.

- a. Escribe un programa que divida dos números enteros ingresados por el usuario.
- b. Modifica el código para usar `double` en lugar de `int` y compara los resultados.



```
13  */
14  public class HolaMundo {
15
16      /**
17       * @param args the command line arguments
18       */
19      public static void main(String[] args) {
20          Scanner input = new Scanner(System.in);
21          int a, b;
22          double division;
23
24          System.out.println(x: "Ingresar 2 números enteros.\n");
25          System.out.print(s: "Primer número: ");
26          a = Integer.parseInt(s: input.nextLine());
27
28          System.out.print(s: "Segundo número: ");
29          b = Integer.parseInt(s: input.nextLine());
30
31          division = a / b;
32          System.out.println("División Int: " + division);
33
34          division = ((double) a / b);
35          System.out.println("División Double: " + division);
36      }
37  }
38
```

Output - holaMundo (run)

```
run:
Ingresar 2 números enteros.
Primer número: 5
Segundo número: 3
División Int: 1.0
División Double: 1.6666666666666667
BUILD SUCCESSFUL (total time: 2 seconds)
```

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);

    }

}
```

- Al momento que el usuario ingresa su nombre, deberíamos de recibir un String por la declaración de dicha variable, sin embargo, estamos intentando ingresar un entero por el nextInt() que esta diseñado para leer números enteros. Esto genera una incompatibilidad de tipos.

Para solucionar este error, basta con cambiar scanner.nextLine().

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);  
  
    }  
  
}
```

- El valor de la variable resultado es 2, porque estamos dividiendo entre enteros, y esto devuelve un entero.  
Si declararíamos resultado como double, el resultado sería 2.0. Si quisiéramos obtener la parte decimal, necesitaríamos que a ó b, sean de tipo double, basta con que una de las variables lo sea.

## CONCLUSIONES ESPERADAS

- Reforzar los conceptos fundamentales del lenguaje Java.
- Familiarizarse con la estructura básica de un programa en Java.
- Aprender a depurar errores comunes.
- Comprender la importancia de las conversiones de tipo y expresiones.
- Adquirir habilidades prácticas para manipular entradas/salidas y variables.
- Aplicar el uso de herramientas como NetBeans y prácticas de depuración.