

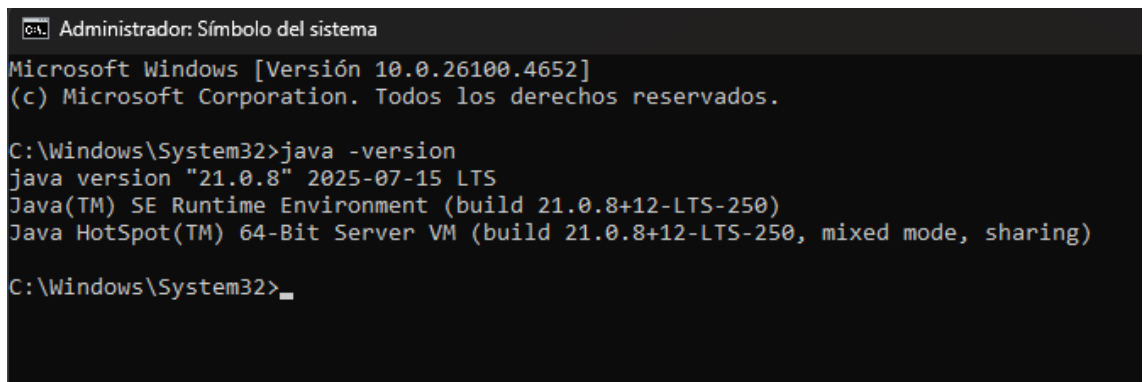
PROGRAMACIÓN II

Trabajo Práctico 1: Introducción a Java

Nombre: Luciano Wittmund.

1. Verificar que tienes instalado Java JDK y NetBeans

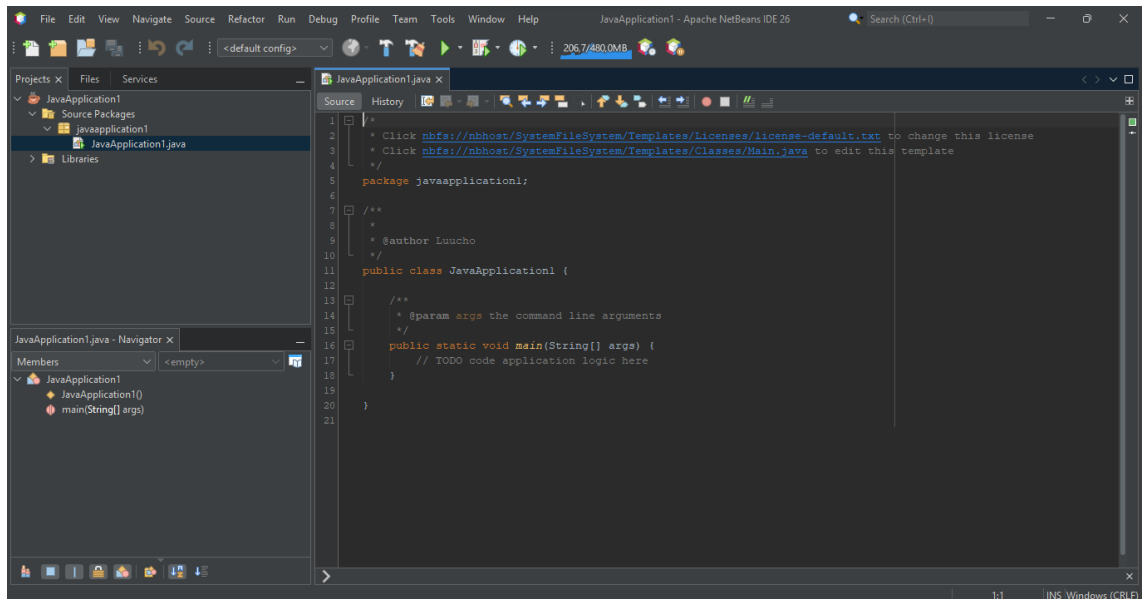
- Confirma que tienes Java JDK instalado ejecutando el siguiente comando en la terminal: `java -version`
- Abre NetBeans, crea un nuevo proyecto y configura el modo oscuro.
- Toma una captura de pantalla del entorno configurado y agrégala a tu entrega.



```
Administrador: Símbolo del sistema
Microsoft Windows [Versión 10.0.26100.4652]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Windows\System32>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)

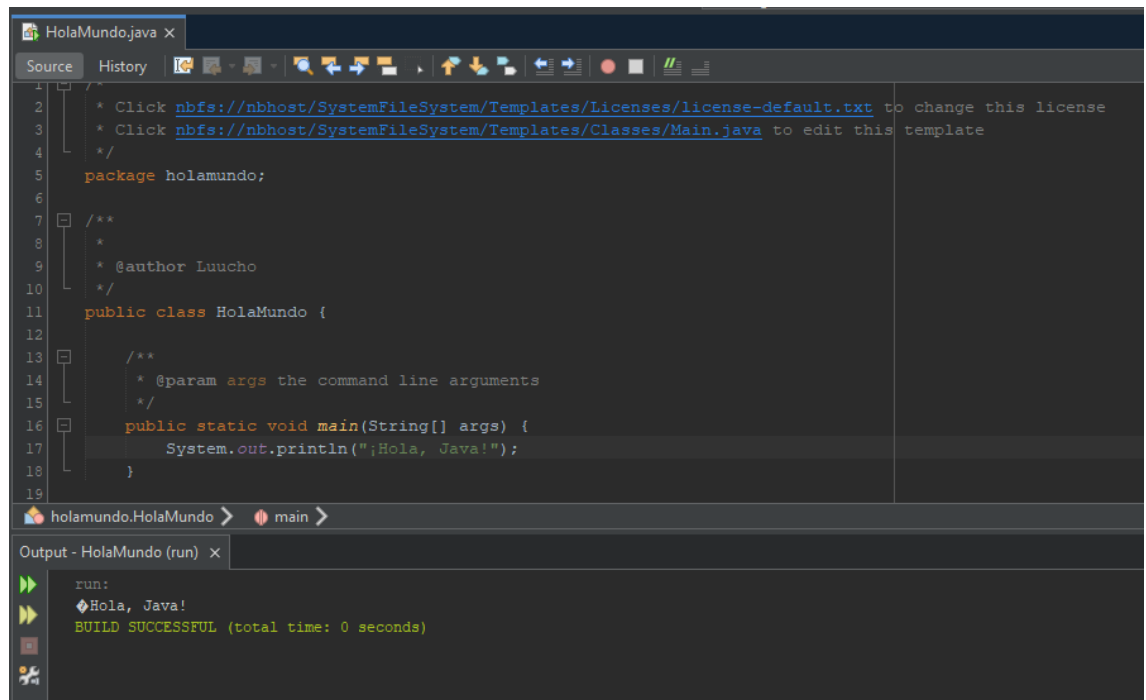
C:\Windows\System32>_
```



```
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help JavaApplication1 - Apache NetBeans IDE 26 Search (Ctrl-I)
<default config> 206.7/480.0MB
Projects X Files Services
JavaApplication1
  Source Packages
    javaapplication1
      JavaApplication1.java
Libraries
JavaApplication1.java - Navigator X
Members
  JavaApplication1
    JavaApplication1()
    main(String[] args)
Source History
1 /**
2  * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
3  * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Main.java to edit this template
4  */
5 package javaapplication1;
6
7 /**
8  *
9  * @author Luucho
10 */
11 public class JavaApplication1 {
12
13     /**
14      * @param args the command line arguments
15      */
16     public static void main(String[] args) {
17         // TODO code application logic here
18     }
19
20 }
21
```

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.



The screenshot shows the NetBeans IDE with a file named `HolaMundo.java` open. The code is as follows:

```
1  /*  
2   * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license  
3   * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Main.java to edit this template  
4   */  
5  package holamundo;  
6  
7  /**  
8   * @author Luucho  
9   */  
10 public class HolaMundo {  
11  
12     /**  
13      * @param args the command line arguments  
14      */  
15     public static void main(String[] args) {  
16         System.out.println("¡Hola, Java!");  
17     }  
18 }  
19
```

Below the code editor, the `Output - HolaMundo (run)` window shows the execution results:

```
run:  
♦ ¡Hola, Java!  
BUILD SUCCESSFUL (total time: 0 seconds)
```

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

- a. String nombre
- b. int edad
- c. double altura
- d. boolean estudiante

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

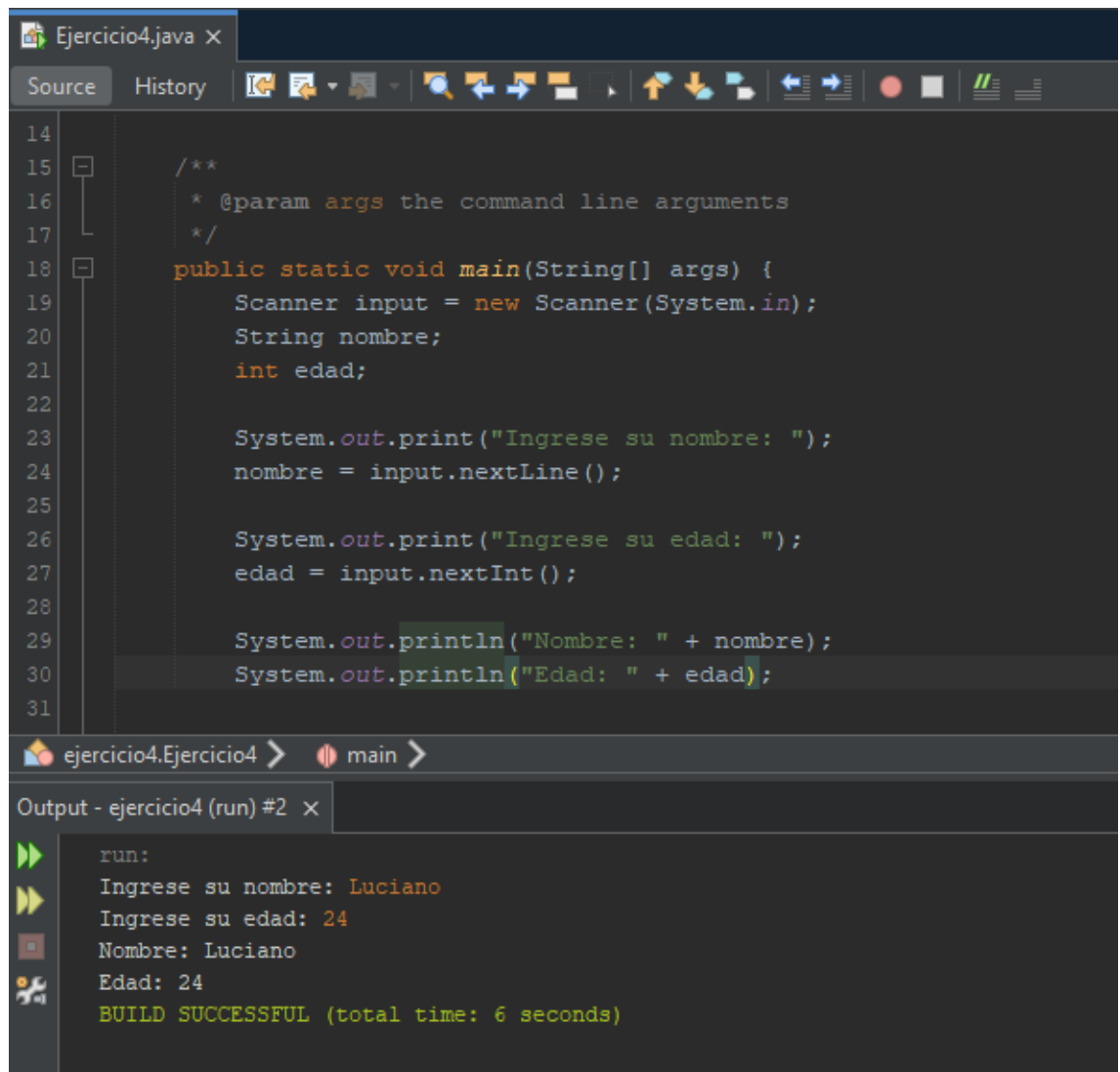
```
10  */
11  public class Ejercicio3 {
12
13      /**
14       * @param args the command line arguments
15       */
16      public static void main(String[] args) {
17          String nombre = "Luciano";
18          int edad = 24;
19          double altura = 1.80;
20          boolean estudiante = true;
21
22          System.out.println("Nombre: " + nombre);
23          System.out.println("Edad: " + edad + " Años");
24          System.out.println("Altura: " + altura + " mts.");
25          System.out.println("Estudiante: " + estudiante);
26      }
27
28  }
```

ejercicio3.Ejercicio3 > main >

Output - Ejercicio3 (run) x

```
run:
Nombre: Luciano
Edad: 24 Años
Altura: 1.8 mts.
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.



The screenshot shows an IDE window titled 'Ejercicio4.java'. The code is as follows:

```
14
15 /**
16  * @param args the command line arguments
17  */
18 public static void main(String[] args) {
19     Scanner input = new Scanner(System.in);
20     String nombre;
21     int edad;
22
23     System.out.print("Ingrese su nombre: ");
24     nombre = input.nextLine();
25
26     System.out.print("Ingrese su edad: ");
27     edad = input.nextInt();
28
29     System.out.println("Nombre: " + nombre);
30     System.out.println("Edad: " + edad);
31 }
```

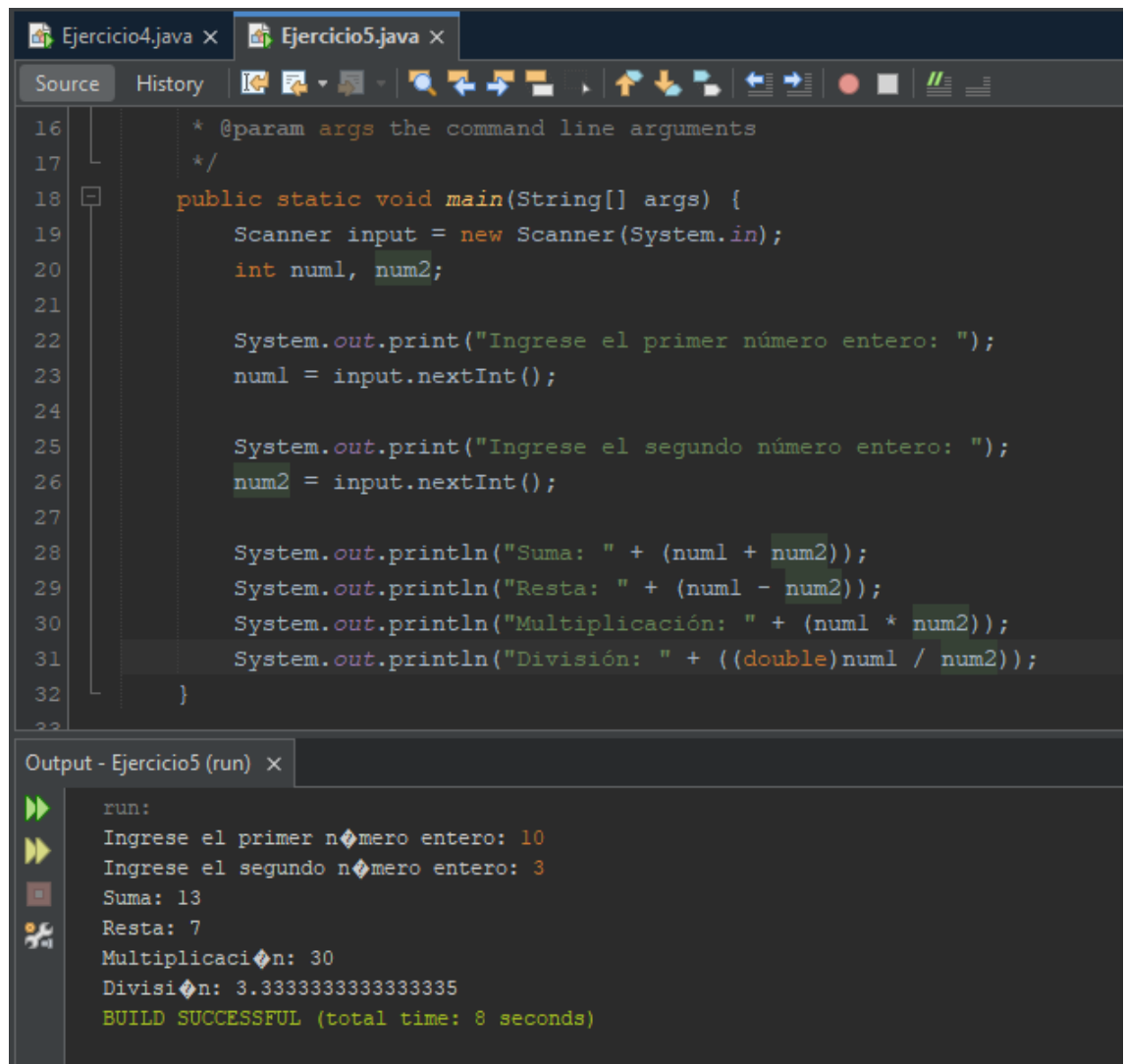
Below the code editor, the 'Output - ejercicio4 (run) #2' window shows the execution results:

```
run:
Ingrese su nombre: Luciano
Ingrese su edad: 24
Nombre: Luciano
Edad: 24
BUILD SUCCESSFUL (total time: 6 seconds)
```

5. Escribe un programa que solicite dos números enteros y realice las siguientes operaciones:

- a. Suma
- b. Resta
- c. Multiplicación
- d. División

Muestra los resultados en la consola



The screenshot shows an IDE with two tabs: 'Ejercicio4.java' and 'Ejercicio5.java'. The 'Ejercicio5.java' tab is active, displaying a Java program that takes two integers as input and performs addition, subtraction, multiplication, and division. The code is as follows:

```
16      * @param args the command line arguments
17      */
18      public static void main(String[] args) {
19          Scanner input = new Scanner(System.in);
20          int num1, num2;
21
22          System.out.print("Ingrese el primer número entero: ");
23          num1 = input.nextInt();
24
25          System.out.print("Ingrese el segundo número entero: ");
26          num2 = input.nextInt();
27
28          System.out.println("Suma: " + (num1 + num2));
29          System.out.println("Resta: " + (num1 - num2));
30          System.out.println("Multiplicación: " + (num1 * num2));
31          System.out.println("División: " + ((double)num1 / num2));
32      }
```

Below the code editor, the 'Output - Ejercicio5 (run)' window shows the execution results:

```
run:
Ingrese el primer número entero: 10
Ingrese el segundo número entero: 3
Suma: 13
Resta: 7
Multiplicación: 30
División: 3.3333333333333335
BUILD SUCCESSFUL (total time: 8 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()`.

```
3  * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Main.java to edit this template
4  */
5  package ejercicio6;
6
7  /**
8   *
9   * @author Luucho
10  */
11  public class Ejercicio6 {
12
13      /**
14       * @param args the command line arguments
15       */
16      public static void main(String[] args) {
17
18          System.out.println("Nombre: Juan Pérez\nEdad: 30 años\nDirección: \"Calle Falsa 123\"");
19
20      }
21
22  }
```

Output - Ejercicio6 (run) x

```
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`

Línea 1: `int x = 10;`

- **Instrucción:** Es una instrucción de declaración e inicialización de variable.
- **Expresión:** 10 es una expresión (un literal que produce un valor).

Línea 2: `x = x + 5;`

- **Instrucción:** Es una instrucción de asignación.
- **Expresión:** `x + 5` es una expresión (suma dos valores y produce uno), y `x = x + 5` también es una expresión (devuelve el nuevo valor asignado, aunque no siempre se use su resultado).

Línea 3: `System.out.println(x);`

- **Instrucción:** Es una instrucción (llamada a un método que produce un efecto: imprimir en consola).
- **Expresión:** `x` es una expresión (accede al valor de la variable).

En Java, **una expresión es cualquier fragmento de código que produce un valor**, como `x + 5` o `10`. En cambio, **una instrucción es una unidad completa de ejecución**, como una declaración, una asignación o una llamada a un método. Las instrucciones **pueden contener expresiones**, pero su propósito principal es realizar acciones, no solo calcular valores.

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.

a-

```
9      * @author Lucho
10     */
11     public class Ejercicio8 {
12
13         /**
14          * @param args the command line arguments
15          */
16         public static void main(String[] args) {
17
18             Scanner input = new Scanner(System.in);
19             int num1, num2;
20
21             System.out.print("Ingrese el primer número entero: ");
22             num1 = input.nextInt();
23
24             System.out.print("Ingrese el segundo número entero: ");
25             num2 = input.nextInt();
26
27             System.out.println(num1 + " / " + num2 + " = " + (num1 / num2));
28         }
29     }

```

Output - Ejercicio8 (run) x

```
run:
Ingrese el primer número entero: 7
Ingrese el segundo número entero: 3
7 / 3 = 2
BUILD SUCCESSFUL (total time: 8 seconds)

```

b-

```
9      * @author Luucho
10     */
11     public class Ejercicio8 {
12
13         /**
14          * @param args the command line arguments
15          */
16         public static void main(String[] args) {
17
18             Scanner input = new Scanner(System.in);
19             double num1, num2;
20
21             System.out.print("Ingrese el primer número entero: ");
22             num1 = input.nextDouble();
23
24             System.out.print("Ingrese el segundo número entero: ");
25             num2 = input.nextDouble();
26
27             System.out.println(num1 + " / " + num2 + " = " + (num1 / num2));
28         }
29     }

```

Output - Ejercicio8 (run) X

```
run:
Ingrese el primer número entero: 7
Ingrese el segundo número entero: 3
7.0 / 3.0 = 2.3333333333333335
BUILD SUCCESSFUL (total time: 11 seconds)

```

- Con int, se pierde la parte decimal.
- Con double, se conserva el valor completo con decimales.

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);

    }

}
```



```
8  /**
9  *
10 * @author Luucho
11 */
12 public class ErrorEjemplo {
13
14     /**
15     * @param args the command line arguments
16     */
17     public static void main(String[] args) {
18         Scanner scanner = new Scanner(System.in);
19
20         System.out.print("Ingresa tu nombre: ");
21
22         String nombre = scanner.nextInt();
23
24         System.out.println("Hola, " + nombre);
25     }
26
27 }
```

Output - ErrorEjemplo (run) x

```
run:
Ingresa tu nombre: Luciano
Hola, Luciano
BUILD SUCCESSFUL (total time: 3 seconds)
```

Qué error tenía y cómo se solucionó?

- El código usaba **scanner.nextInt()** para leer un **nombre**, pero **nextInt()** sirve **solo para leer números enteros**.
- Al intentar guardar el valor en una **variable String**, ocurría un error de tipos: no se puede asignar un **int** a un **String**.

Solución: reemplazamos **scanner.nextInt()** por **scanner.nextLine()**, que es el método correcto para **leer texto o cadenas de caracteres**, como un nombre.

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 de código	a	b	resultado
int a = 5;	5	—	—
int b = 2;	5	2	—
int resultado = a / b;	5	2	2
System.out.println(...)	5	2	2