

TP 1: Java

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 -version`

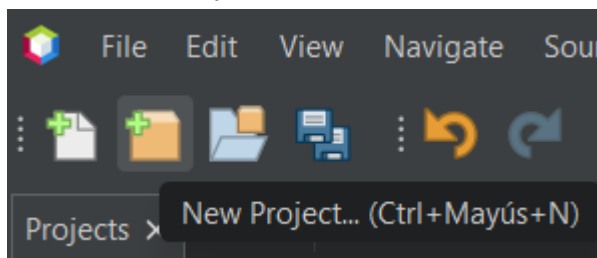
```
Windows PowerShell
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

Instale la versión más reciente de PowerShell para obtener nuevas características y mejoras. https://aka.ms/PSWindows

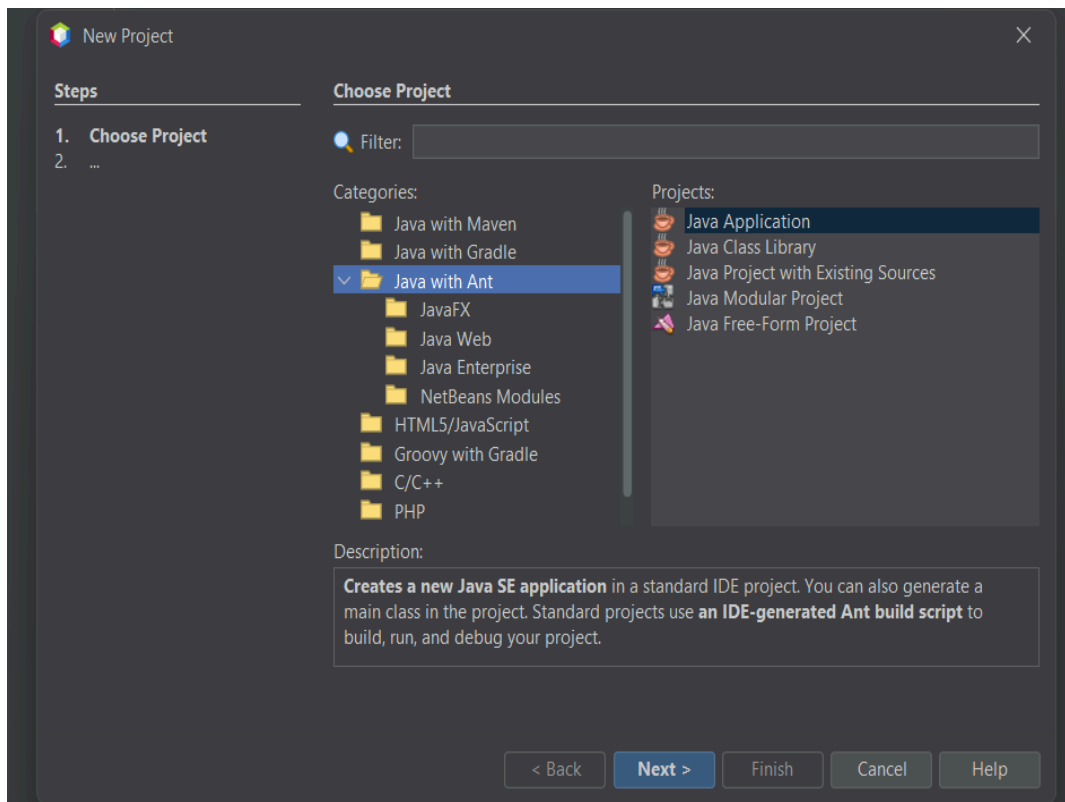
PS C:\Users\herre> java -version
java version "24.0.2" 2025-07-15
Java(TM) SE Runtime Environment (build 24.0.2+12-54)
Java HotSpot(TM) 64-Bit Server VM (build 24.0.2+12-54, mixed mode, sharing)
PS C:\Users\herre>
```

b. Abre NetBeans, crea un nuevo proyecto y configura el modo oscuro.

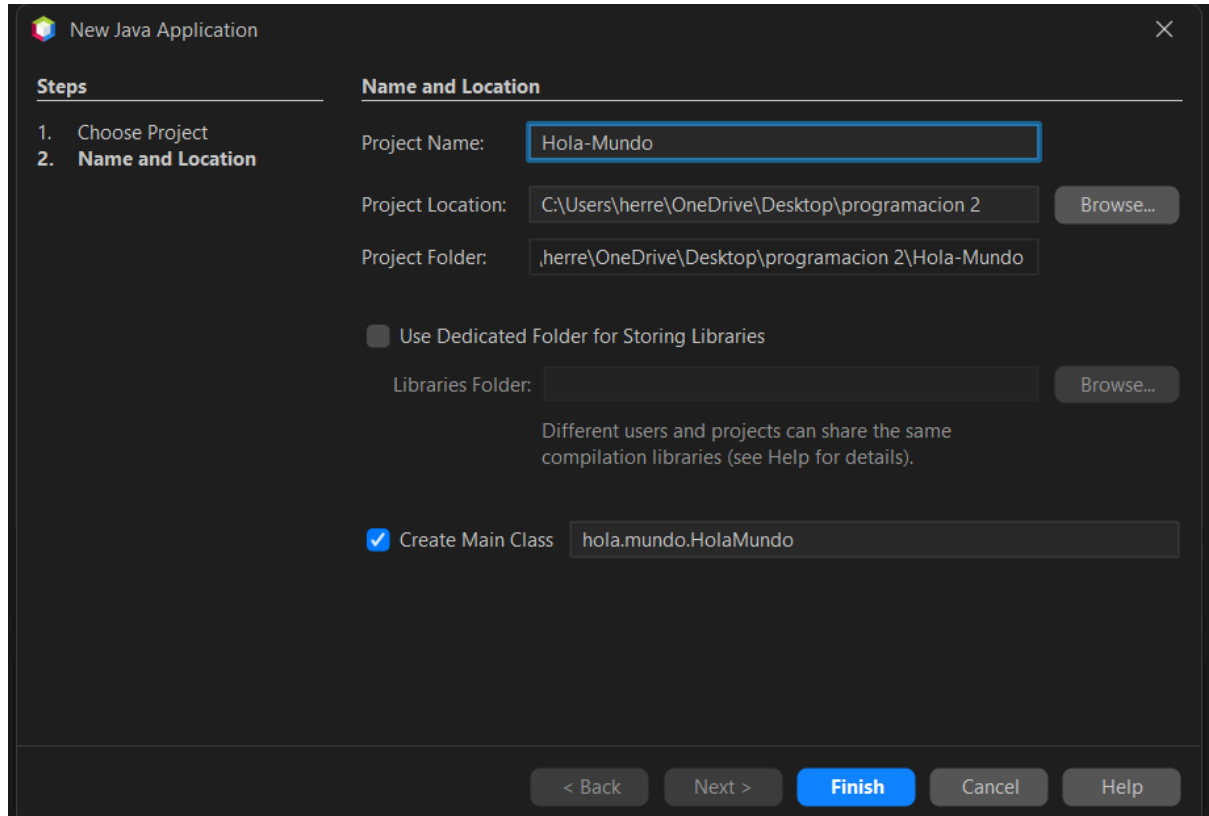
1. Creamos un proyecto



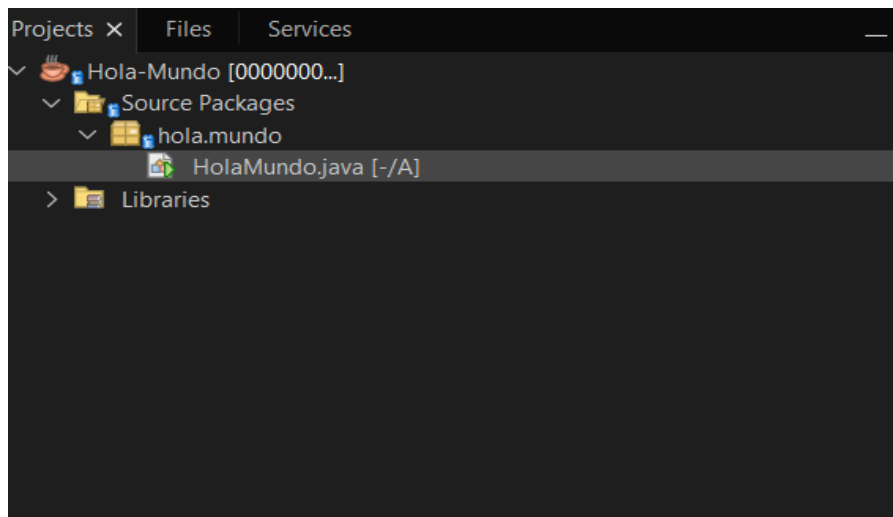
2. Seleccionamos "Java with Ant" y luego "Java Application". Seguido a esto oprimimos en "Next"



3. Le asignamos un nombre al proyecto y colocamos finish.

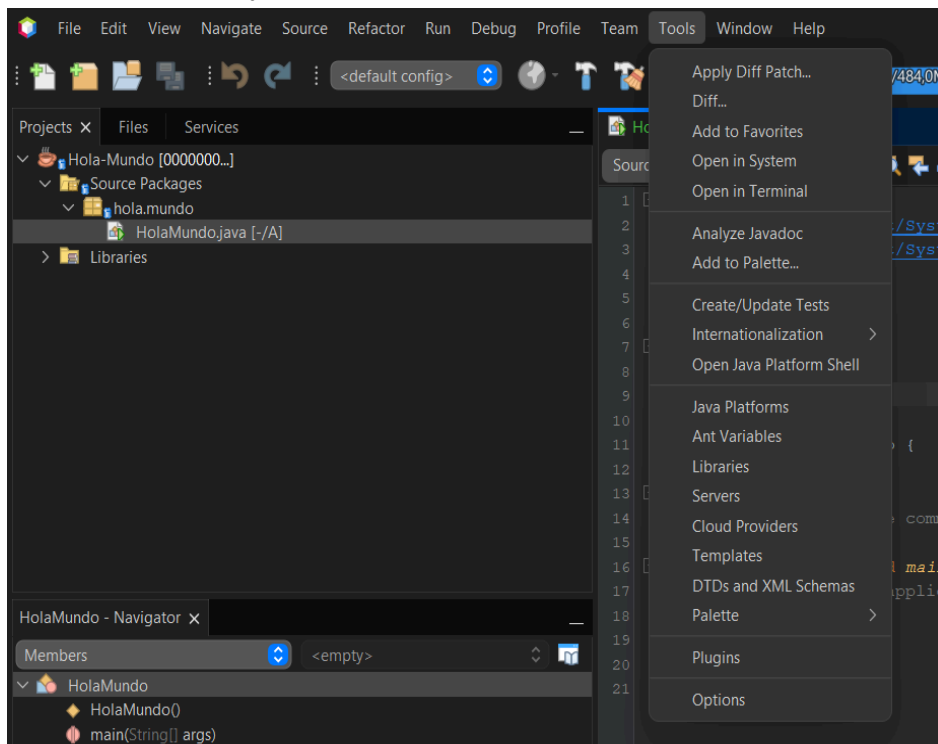


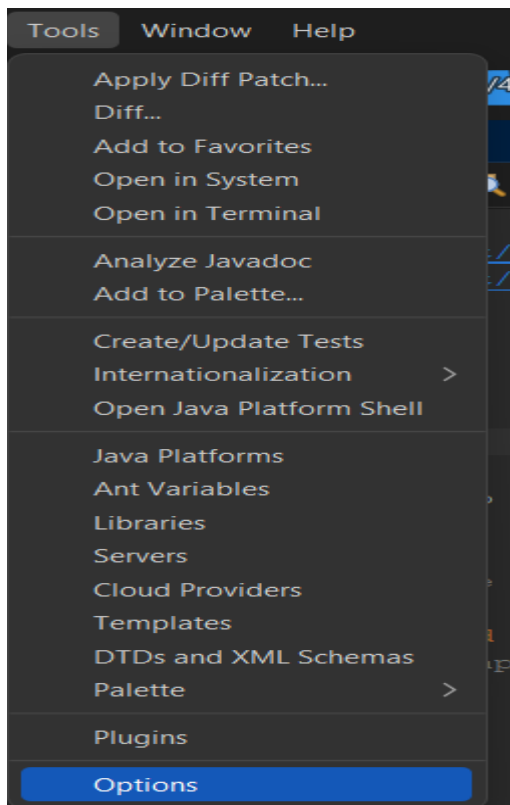
4. Del lado izquierdo de la página podemos ver que el proyecto está creado.



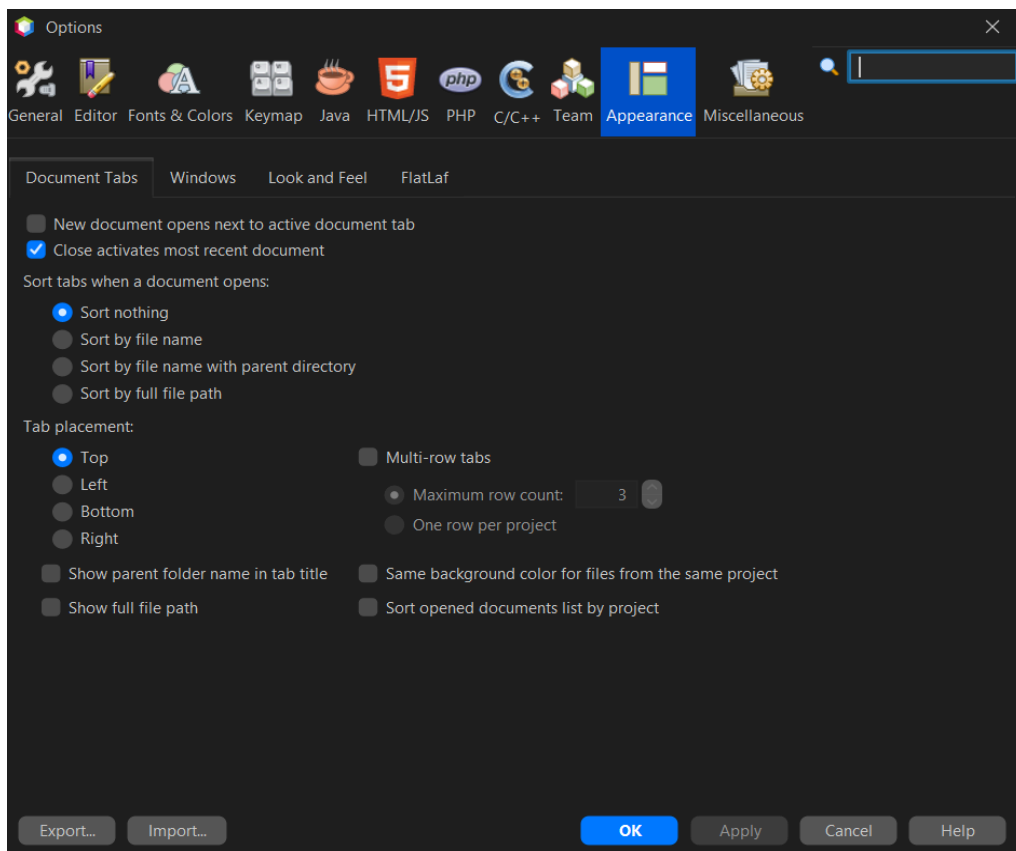
CONFIGURAR EN MODO OSCURO:

1. Oprimimos “Tools” y hacemos clic en “options”

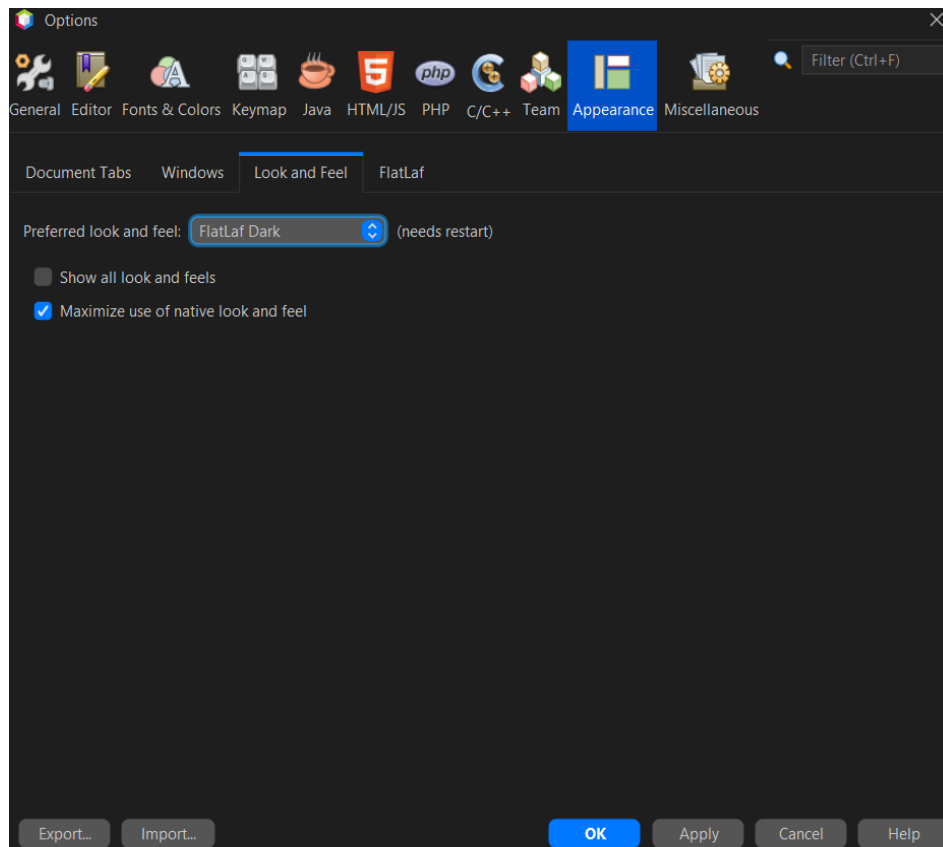




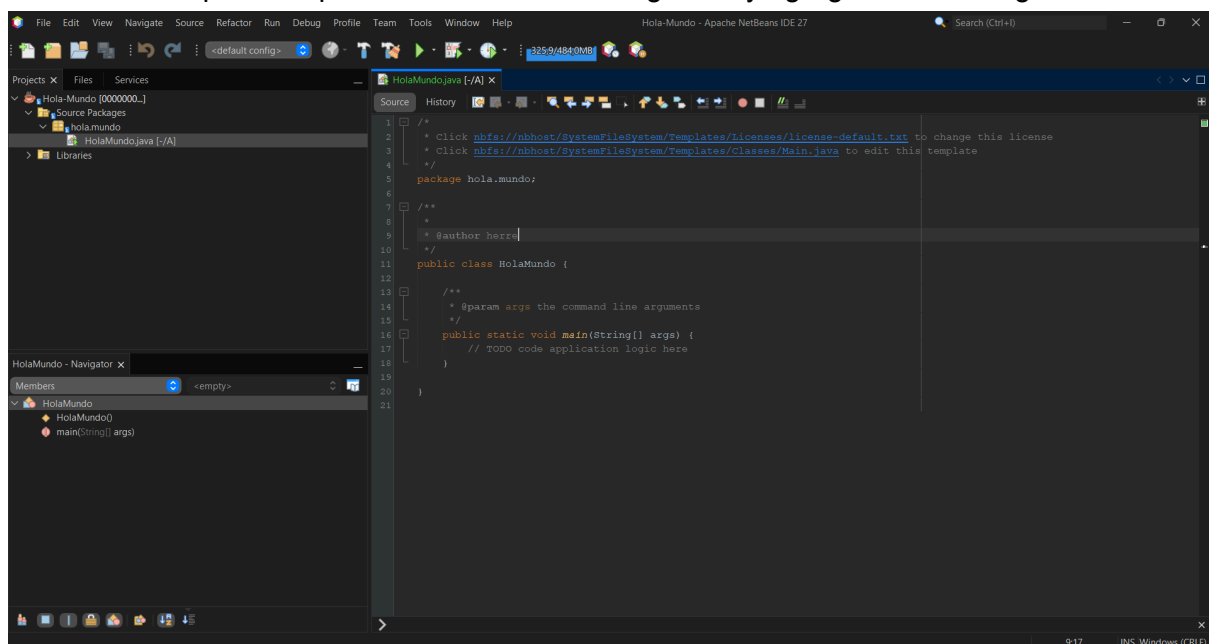
2. Ahora seleccionamos en “Appearance”



3. Luego vamos a “Look and Feel”, seleccionamos el modo “FlatLaf Dark” y presionamos apply. NetBeans nos pedirá reiniciar la página para poder mostrar los resultados.

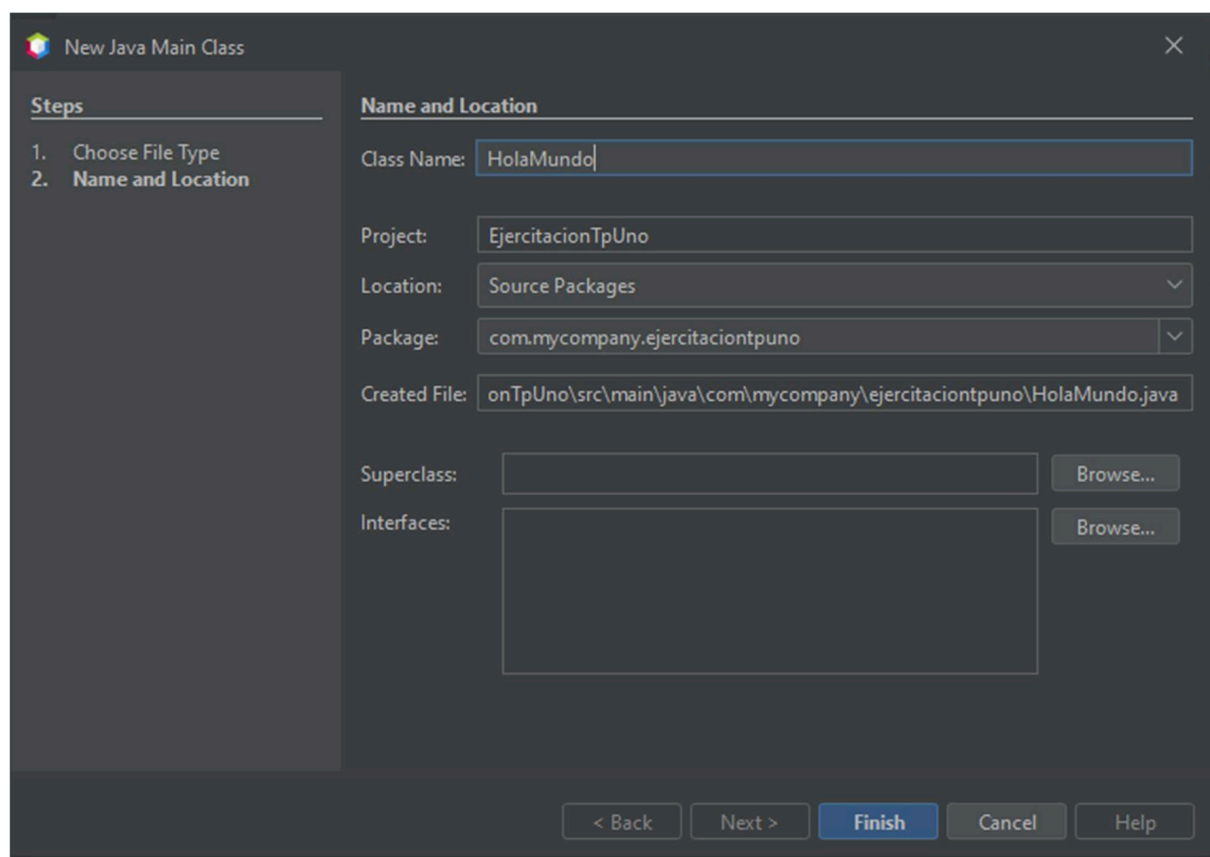
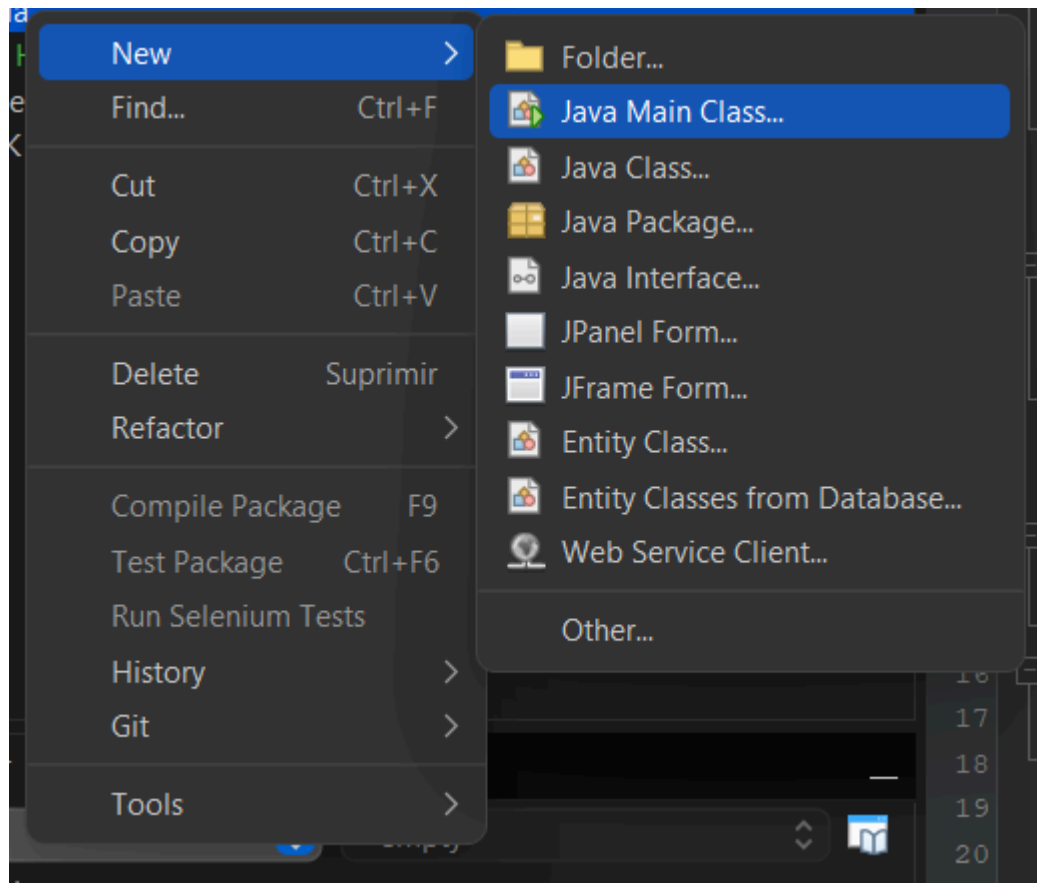


c. Toma una captura de pantalla del entorno configurado y agrégala a tu entrega.

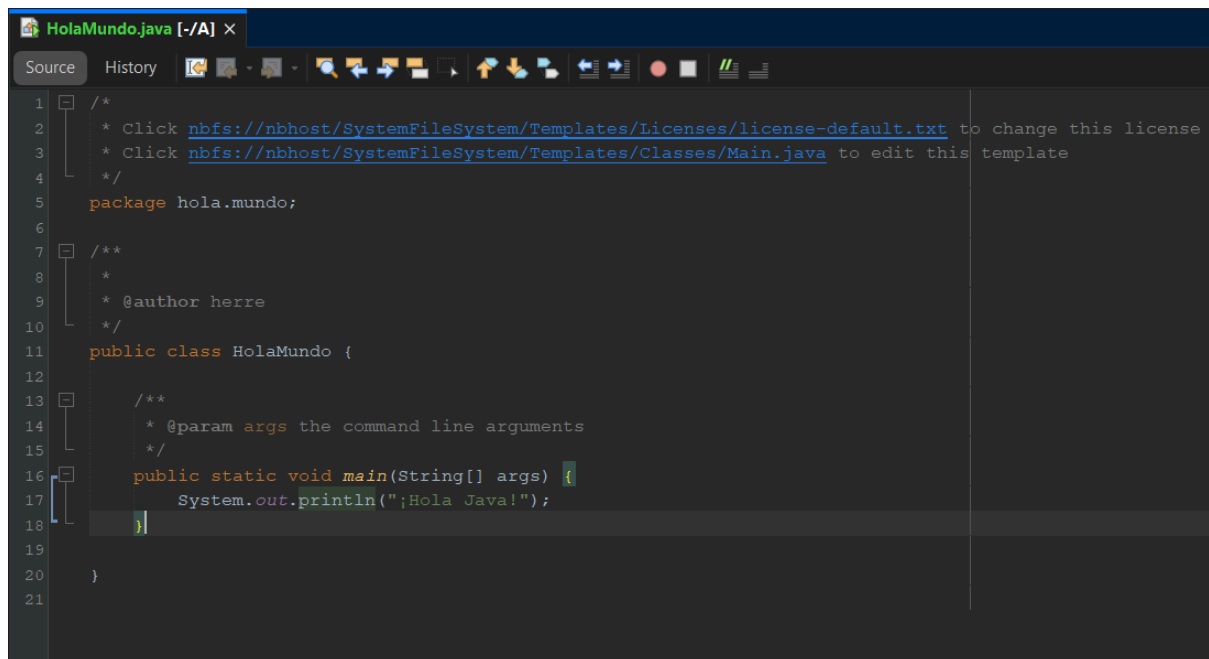


2. Escribir y ejecutar un programa básico en Java.

a. Creá una clase llamada HolaMundo.



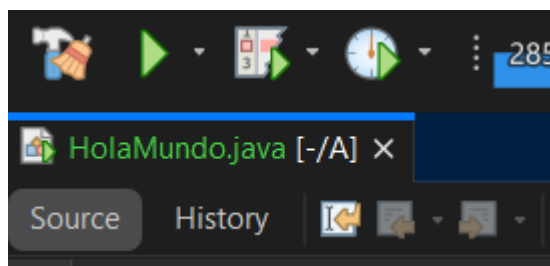
b. Escribe un programa que imprima el mensaje: ¡Hola, Java!

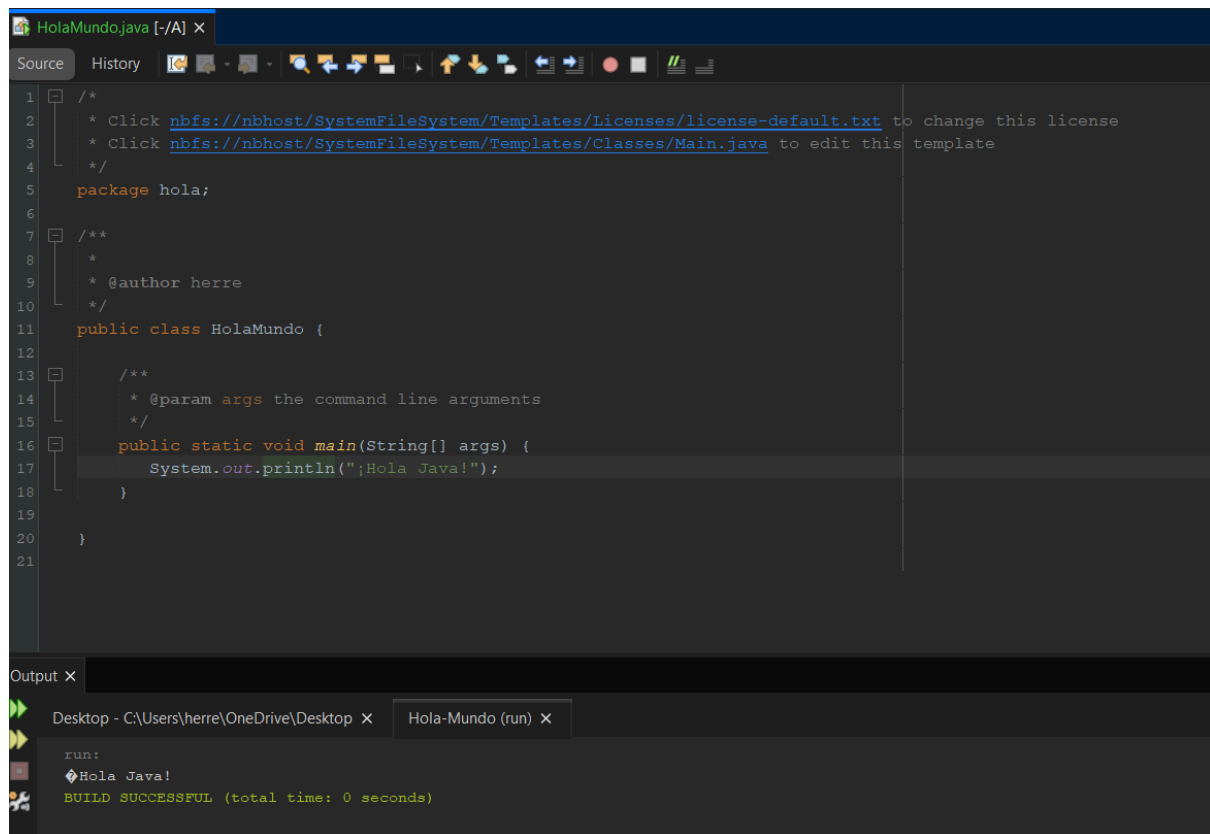


```
1  /*
2   * Click nbfs:///nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
3   * Click nbfs:///nbhost/SystemFileSystem/Templates/Classes/Main.java to edit this template
4   */
5   package hola.mundo;
6
7   /**
8    *
9    * @author herre
10   */
11   public class HolaMundo {
12
13       /**
14        * @param args the command line arguments
15        */
16       public static void main(String[] args) {
17           System.out.println("¡Hola Java!");
18       }
19
20   }
```

c. Ejecuta el programa en NetBeans y adjunta una captura del resultado en la consola.

Primero presionamos en el triángulo verde que se encuentra arriba a la izquierda y luego se adjunta el resultado en la parte inferior.





```
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 hola;
6
7  /**
8   *
9   * @author herre
10  */
11  public class HolaMundo {
12
13      /**
14       * @param args the command line arguments
15       */
16      public static void main(String[] args) {
17          System.out.println("¡Hola Java!");
18      }
19
20  }
21
```

Output

Desktop - C:\Users\herre\OneDrive\Desktop X Hola-Mundo (run) X

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().

1. Nombramos las variables nombre, edad, altura, estudiante y ejecutamos.


```
6  /**
7  *
8  * @author herre
9  */
10 public class Punto3 {
11
12     /**
13     * @param args the command line arguments
14     */
15     public static void main(String[] args) {
16         String nombre= "Jazmin Herrera";
17         int edad= 18;
18         double altura= 1.64;
19         boolean estudiante = true;
20
21         System.out.println(nombre);
22         System.out.println(edad);
23         System.out.println(altura);
24         System.out.println(estudiante);
25
26
27
28     }
29 }
```

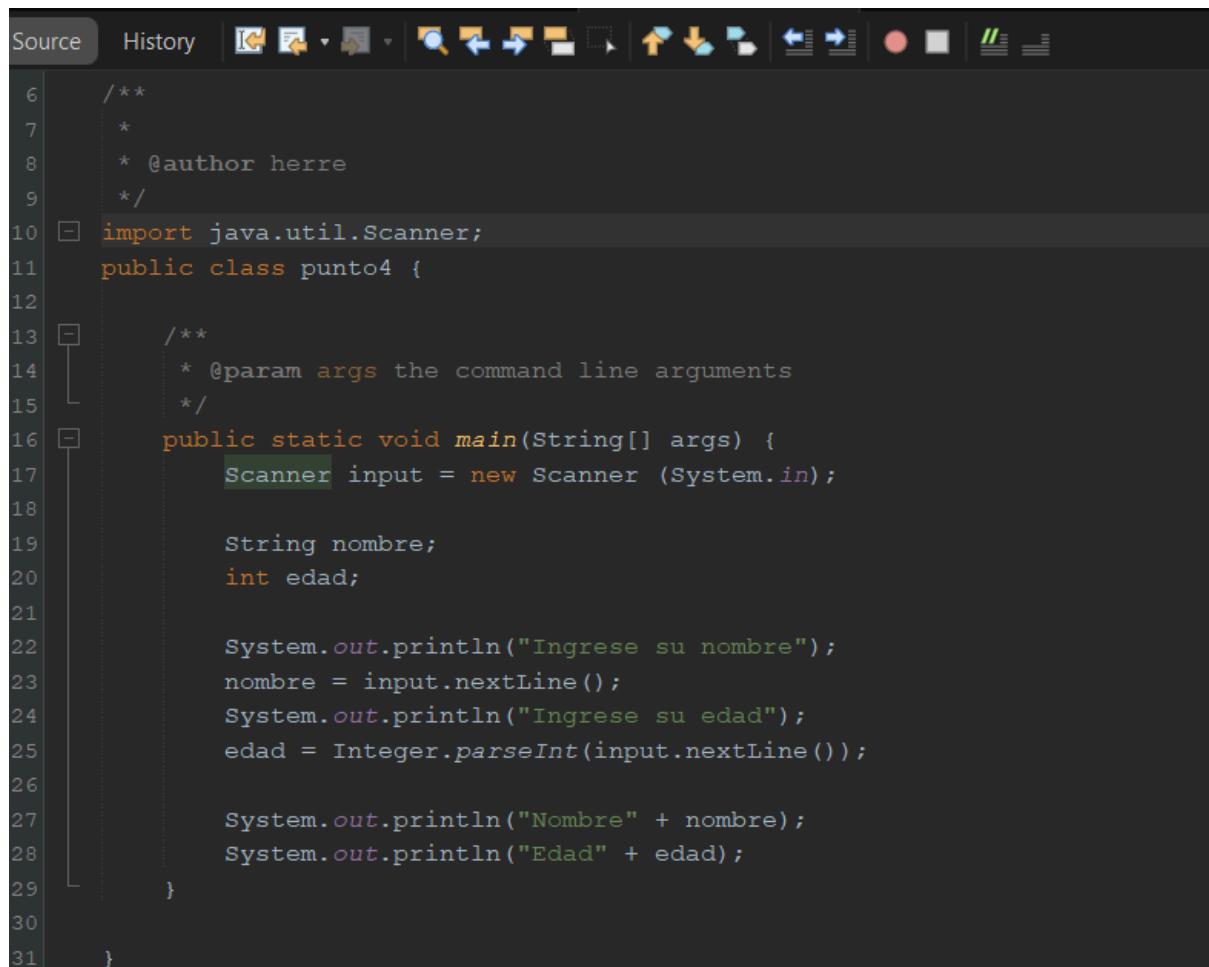
Output X

Desktop - C:\Users\herre\OneDrive\Desktop X Hola-Mundo (run) X

run:
Jazmin Herrera
18
1.64
true
BUILD SUCCESSFUL (total time: 0 seconds)

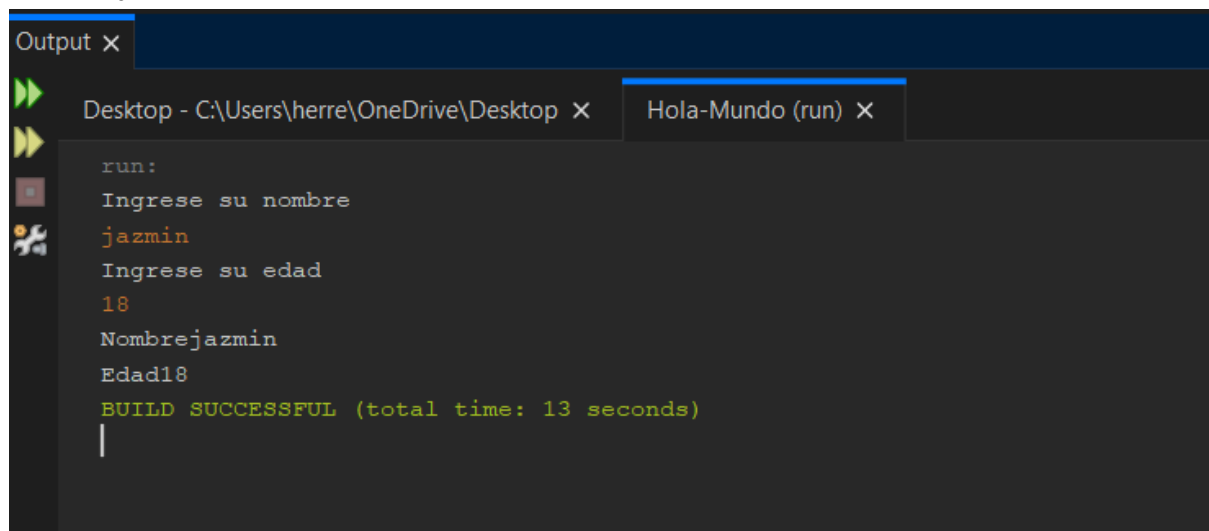
4. Escribe un programa que solicite al usuario ingresar su nombre y edad, y luego los muestre en pantalla. Usa Scanner para capturar los datos.

1. Declaramos las variables.



```
6  /**
7  *
8  * @author herre
9  */
10 import java.util.Scanner;
11 public class punto4 {
12
13     /**
14     * @param args the command line arguments
15     */
16     public static void main(String[] args) {
17         Scanner input = new Scanner (System.in);
18
19         String nombre;
20         int edad;
21
22         System.out.println("Ingrese su nombre");
23         nombre = input.nextLine();
24         System.out.println("Ingrese su edad");
25         edad = Integer.parseInt(input.nextLine());
26
27         System.out.println("Nombre" + nombre);
28         System.out.println("Edad" + edad);
29     }
30
31 }
```

2. Ejecutamos



```
run:
Ingrese su nombre
jazmin
Ingrese su edad
18
Nombrejazmin
Edad18
BUILD SUCCESSFUL (total time: 13 seconds)
```

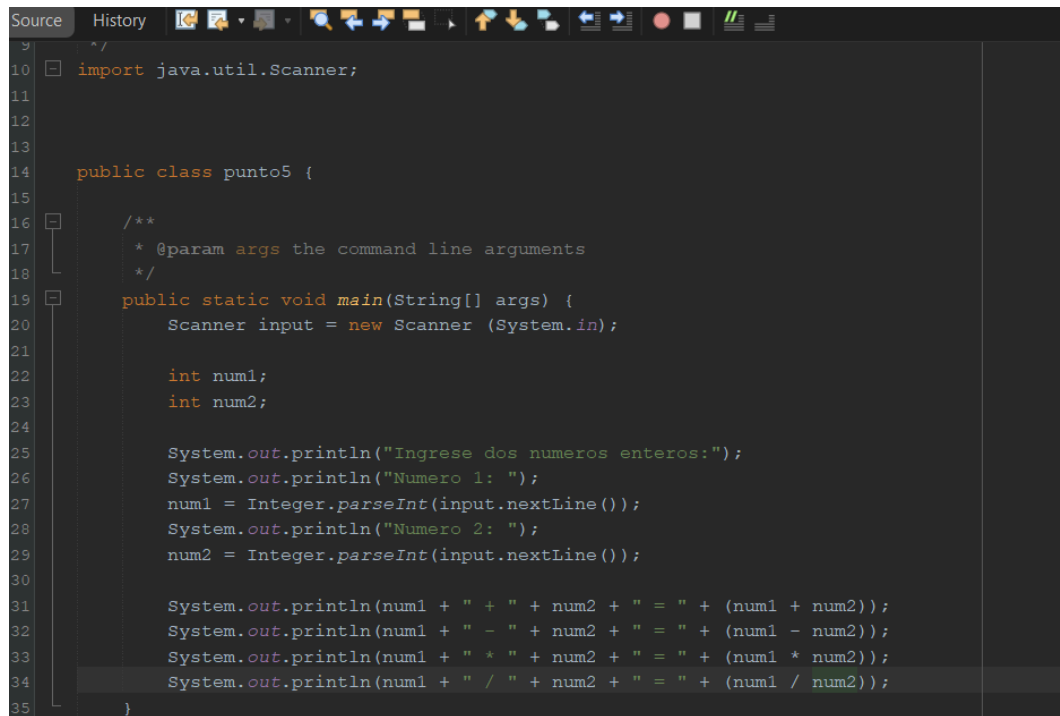
5. Escribe un programa que solicite dos números enteros y realice las siguientes operaciones:

a. Suma

b. Resta

c. Multiplicación

d. División Muestra los resultados en la consola.



```
Source History
9  ^/
10  import java.util.Scanner;
11
12
13
14  public class punto5 {
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
22          int num1;
23          int num2;
24
25          System.out.println("Ingrese dos numeros enteros:");
26          System.out.println("Numero 1: ");
27          num1 = Integer.parseInt(input.nextLine());
28          System.out.println("Numero 2: ");
29          num2 = Integer.parseInt(input.nextLine());
30
31          System.out.println(num1 + " + " + num2 + " = " + (num1 + num2));
32          System.out.println(num1 + " - " + num2 + " = " + (num1 - num2));
33          System.out.println(num1 + " * " + num2 + " = " + (num1 * num2));
34          System.out.println(num1 + " / " + num2 + " = " + (num1 / num2));
35      }
```

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   * 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
6  /**
7   *
8   * @author herre
9   */
10 public class punto6 {
11
12     /**
13      * @param args the command line arguments
14      */
15     public static void main(String[] args) {
16         System.out.println("Nombre: Juan Perez \n" +
17                             "Edad: 30 anos \n" +
18                             "Direccion: \"Calle Falsa 123\"");
19     }
20
21 }
22
```

Output - Hola-Mundo (run) x

```
run:
Nombre: Juan Perez
Edad: 30 anos
Direccion: "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

- **Instrucción:** porque declara una variable y le asigna un valor inicial.

Línea 2

- **Instrucción:** porque asigna un valor a la variable x.

Línea 3

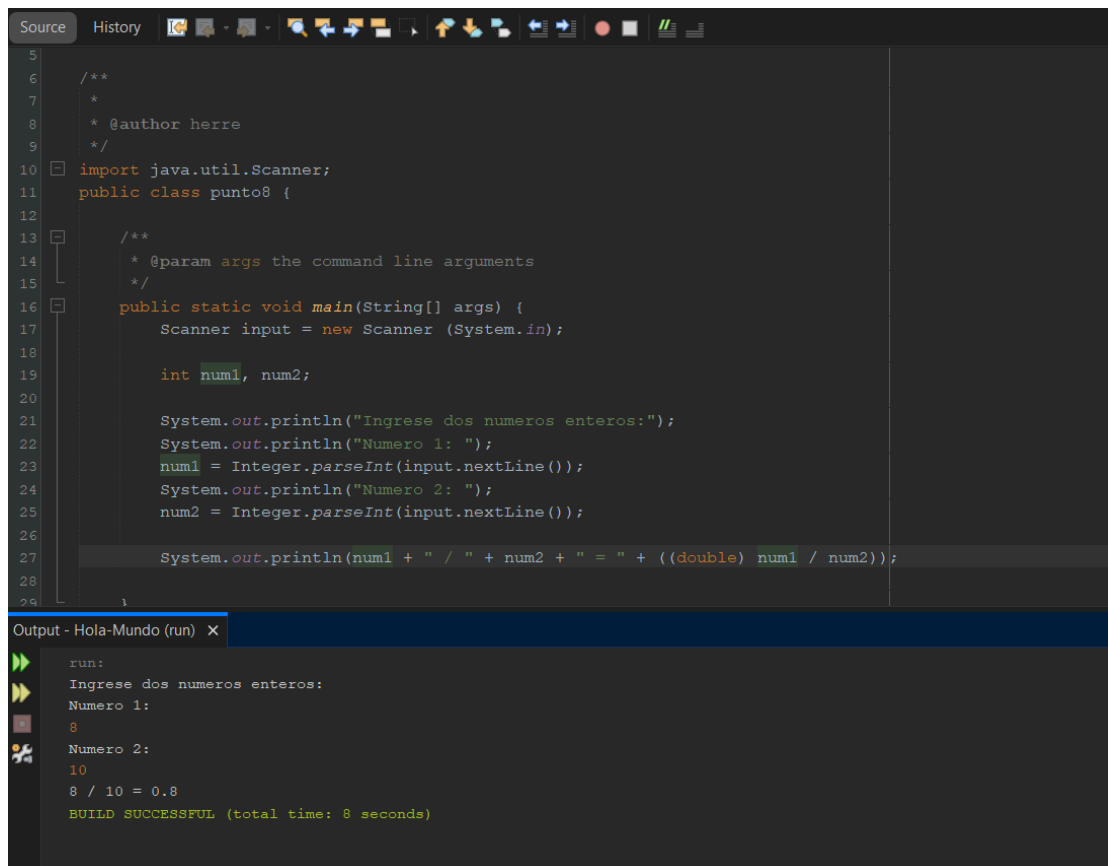
- **Instrucción:** porque ejecuta una acción (imprimir por pantalla).

La diferencia es que una expresión produce un valor y una instrucción realiza una acción en el programa.

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.



The screenshot shows an IDE with a Java source file named 'punto8.java'. The code imports 'java.util.Scanner' and defines a class 'punto8' with a 'main' method. The 'main' method prompts the user to enter two integers, reads them using 'Scanner', and then prints the result of their division as a double. The output window shows the execution results: 'Ingrese dos numeros enteros:', 'Numero 1: 8', 'Numero 2: 10', and '8 / 10 = 0.8'. The build is successful.

```
5
6  /**
7   *
8   * @author herre
9   */
10 import java.util.Scanner;
11 public class punto8 {
12
13     /**
14     * @param args the command line arguments
15     */
16     public static void main(String[] args) {
17         Scanner input = new Scanner (System.in);
18
19         int num1, num2;
20
21         System.out.println("Ingrese dos numeros enteros:");
22         System.out.println("Numero 1: ");
23         num1 = Integer.parseInt(input.nextLine());
24         System.out.println("Numero 2: ");
25         num2 = Integer.parseInt(input.nextLine());
26
27         System.out.println(num1 + " / " + num2 + " = " + ((double) num1 / num2));
28
29     }
30 }
```

Output - Hola-Mundo (run) X

```
run:
Ingrese dos numeros enteros:
Numero 1:
8
Numero 2:
10
8 / 10 = 0.8
BUILD SUCCESSFUL (total time: 8 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); } }
```

El error ocurre porque se utilizó `nextInt()` (para enteros) en lugar de `nextLine()` (para cadenas). Como `int` no se puede asignar a `String`, el compilador marca un error de tipos incompatibles.

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	Valor de a	Valor de b	Resultado
<code>int a = 5</code>	5	-	-
<code>int b = 2</code>	5	2	-
<code>int resultado = a/b</code>	5	2	2
<code>System.out.println("Resultado: " + resultado);</code>	-	-	"resultado:2"

El valor de `resultado` es 2 porque el programa realiza una división entre 5 y 2, lo que daría un número decimal 2.5. Sin embargo, como la variable está declarada como `int` (entero), el resultado que se obtiene es 2, ya que se descarta la parte decimal.