

# PROGRAMACIÓN II

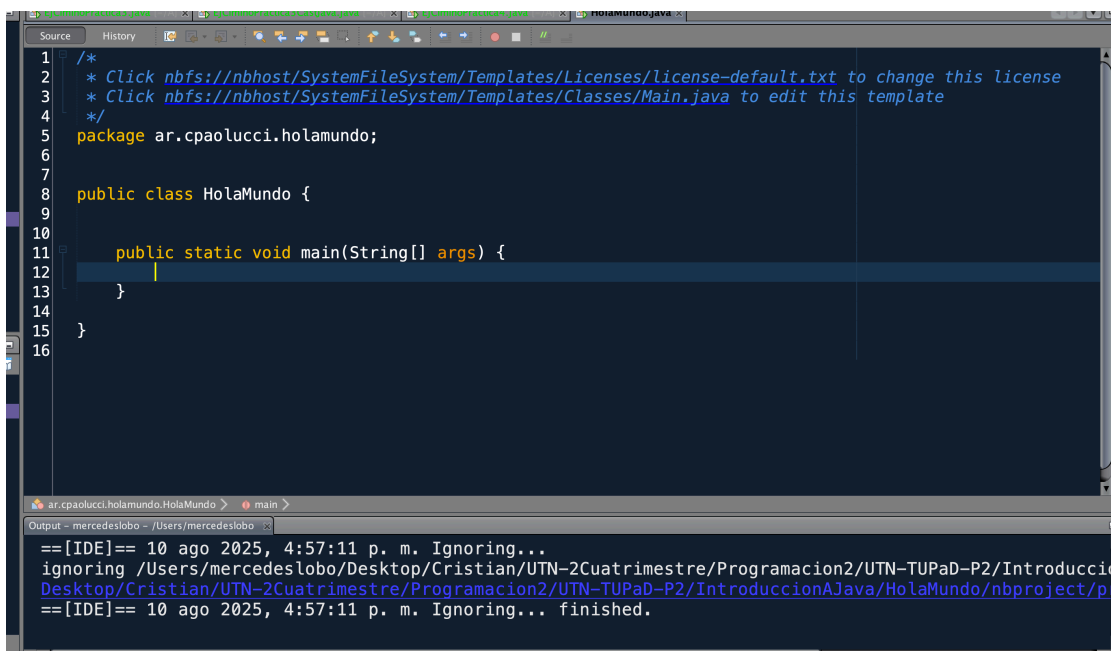
## Trabajo Práctico 1: Introducción a Java

### 1. Capturas de pantalla

a)

```
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)
mercedeslobo@MacBook-Pro-de-Mercedes: %
```

c)

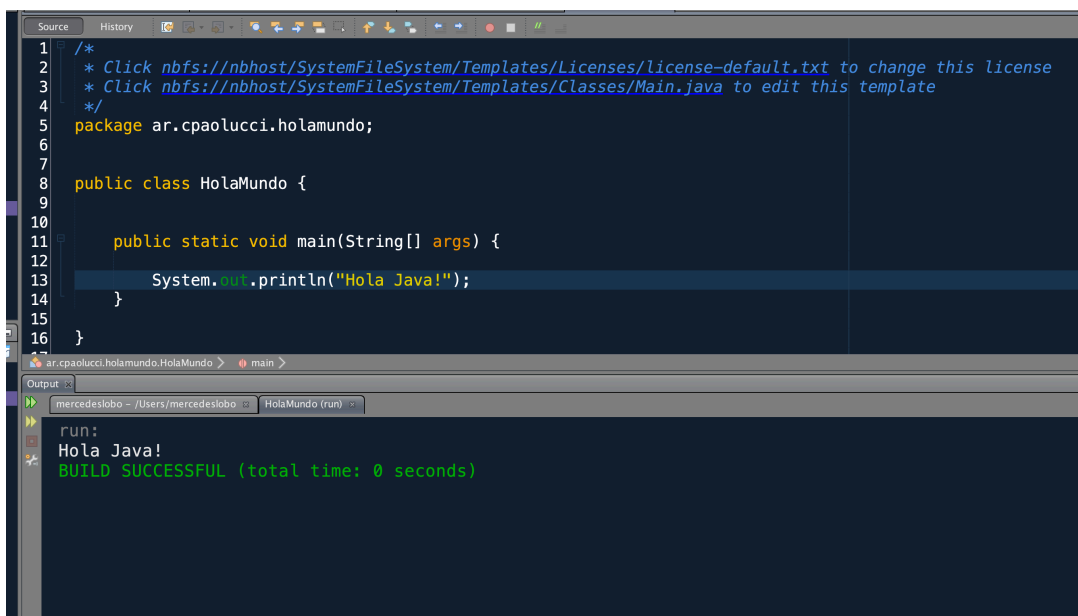


```
1  /*
2  * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
3  * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Main.java to edit this template
4  */
5  package ar.cpaolucci.holamundo;
6
7
8  public class HolaMundo {
9
10
11     public static void main(String[] args) {
12         |
13     }
14
15 }
16
```

Output - mercedeslobo - /Users/mercedeslobo

```
==[IDE]== 10 ago 2025, 4:57:11 p. m. Ignoring...
ignoring /Users/mercedeslobo/Desktop/Cristian/UTN-2Cuatrimestre/Programacion2/UTN-TUPaD-P2/Introduccion
Desktop/Cristian/UTN-2Cuatrimestre/Programacion2/UTN-TUPaD-P2/IntroduccionAJava/HolaMundo/nbproject/p
==[IDE]== 10 ago 2025, 4:57:11 p. m. Ignoring... finished.
```

### 2. Captura de pantalla



```
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 ar.cpaolucci.holamundo;
6
7
8  public class HolaMundo {
9
10
11     public static void main(String[] args) {
12         |
13         System.out.println("Hola Java!");
14     }
15
16 }
```

Output - mercedeslobo - /Users/mercedeslobo

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

## 7. Análisis:

### Línea 1: `int x = 10;`

- Es una **instrucción** de declaración y asignación.
- Contiene una **expresión**: 10, que es evaluada y asignada a x.

### Línea 2: `x = x + 5;`

- Es una **instrucción** de asignación.
- Contiene una **expresión**: `x + 5`, que se evalúa y luego se asigna a x.

### Línea 3: `System.out.println(x);`

- Es una **instrucción** que ejecuta un método (imprime un valor en la consola).
- El parámetro x es una **expresión**, ya que representa un valor que se evalúa antes de imprimirse.

### Diferencia entre expresión e instrucción:

Una **expresión** es cualquier parte del código que **produce un valor** al evaluarse, como 10, `x + 5` o x.

Una **instrucción** (o sentencia) es una línea completa que **realiza una acción**, como declarar una variable, asignar un valor o llamar a un método.

En resumen, **las expresiones están contenidas dentro de las instrucciones** y se usan para generar valores que luego se usan para ejecutar acciones.

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);
}
}
```

**String nombre = scanner.nextInt();**

El error es porque se esta intentando guardar un **número entero (int)** en una **variable de tipo String**, lo cual **no es compatible**.

Lo correcto seria:

**String nombre = scanner.nextLine();**

nextInt() devuelve un entero (int), no un String.