

TP01 - Introducción a Java

 NetBeans  Java 21.0.8.LTS  license MIT  Repository  GitHub

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	Permiten 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	Ánálisis paso a paso de ejecución de código

Caso práctico

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

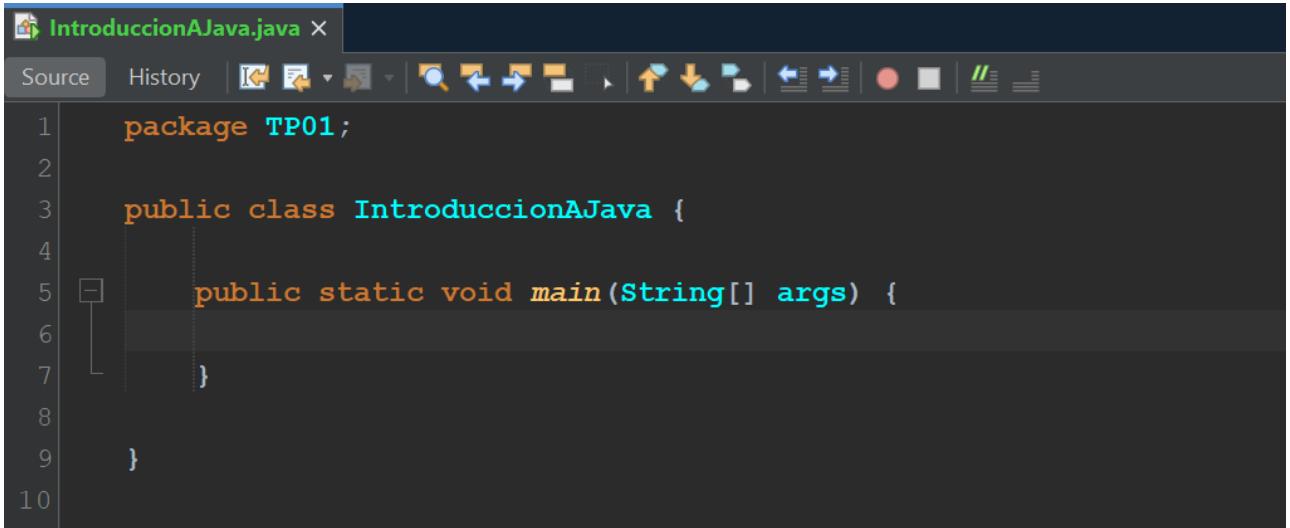
- Configurar correctamente el entorno de desarrollo (Java JDK y NetBeans).
- Crear programas básicos que imprimen mensajes en consola.
- Declarar variables de distintos tipos y manipular sus valores.
- Leer datos ingresados por el usuario usando Scanner.
- Realizar operaciones aritméticas básicas.
- 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 instalación del entorno

- Confirmar instalación de Java con `java -version`:

```
lahoz@cristian-lahoz MINGW64 ~
$ 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)
```

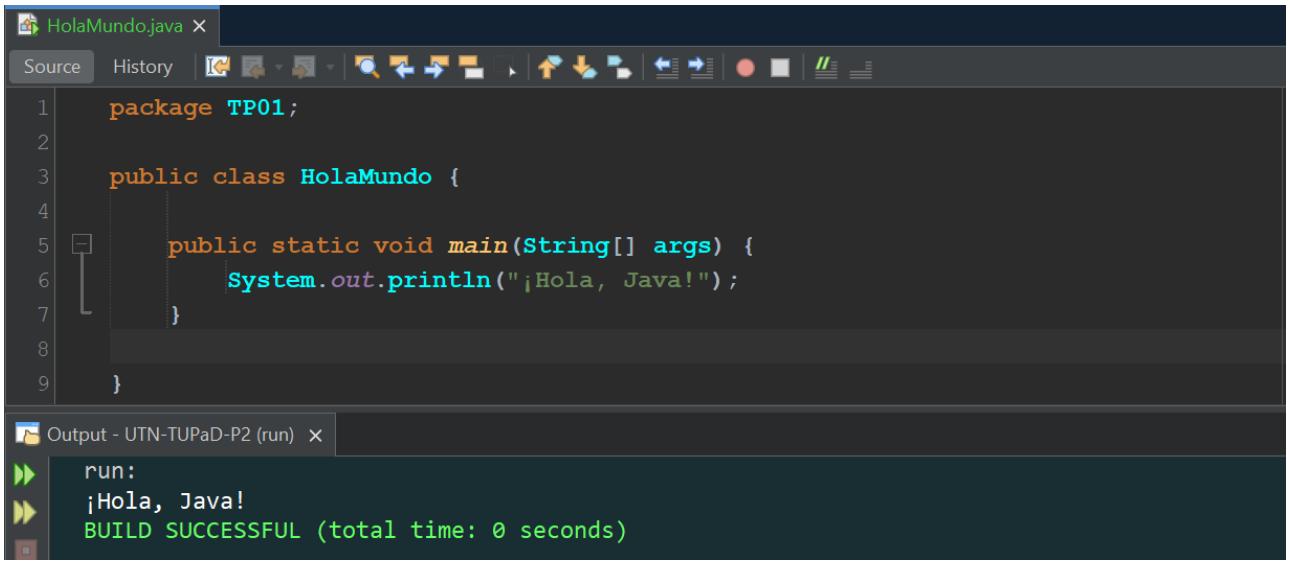
- Abrir NetBeans, crear nuevo proyecto y activar modo oscuro.
- Captura de pantalla del entorno configurado:



```
1 package TP01;
2
3 public class IntroduccionAJava {
4
5     public static void main(String[] args) {
6
7     }
8
9 }
10
```

2. Programar "Hola Mundo"

- Crear clase `HolaMundo`.
- Imprimir en consola: `¡Hola, Java!`
- Captura de ejecución:



```
1 package TP01;
2
3 public class HolaMundo {
4
5     public static void main(String[] args) {
6         System.out.println("¡Hola, Java!");
7     }
8
9 }
```

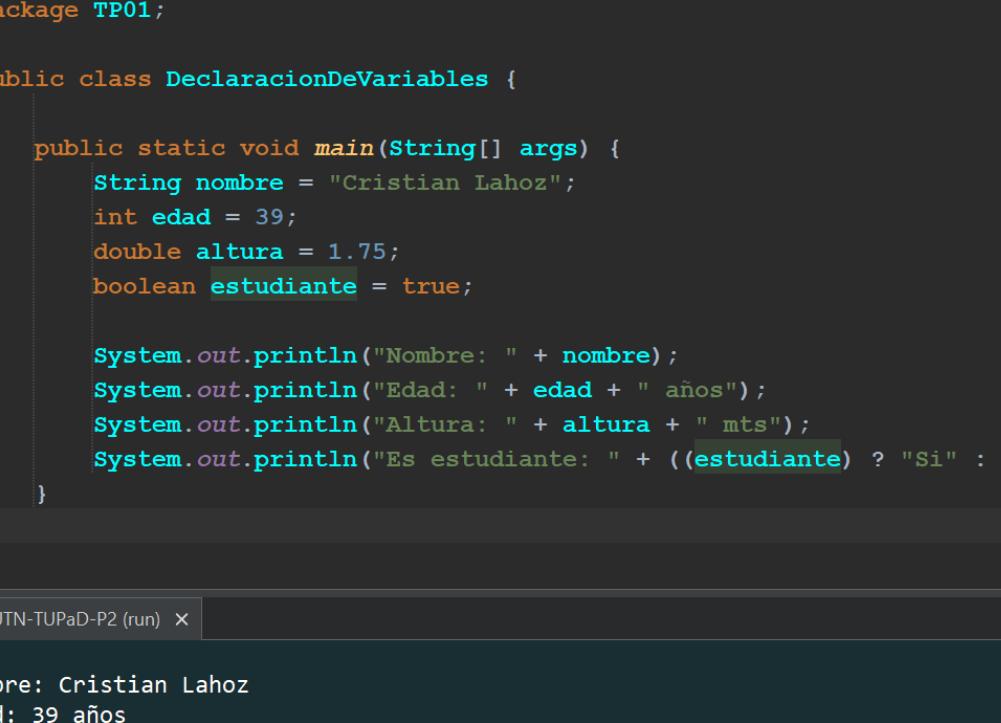
Output - UTN-TUPaD-P2 (run) x

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

3. Declaración de variables

- Declarar y asignar valores:
 - `String nombre`
 - `int edad`
 - `double altura`
 - `boolean estudiante`

- Imprimir los valores en pantalla usando `System.out.println()`:



The screenshot shows an IDE interface with two tabs: 'DeclaracionDeVariables.java' and 'Output - UTN-TUPaD-P2 (run)'. The Java code in the top tab defines a class 'DeclaracionDeVariables' with a main method. It declares four variables: 'nombre' (String), 'edad' (int), 'altura' (double), and 'estudiante' (boolean). It then prints these variables using System.out.println. The output tab shows the execution of the code, displaying the printed values: 'Nombre: Cristian Lahoz', 'Edad: 39 años', 'Altura: 1.75 mts', and 'Es estudiante: Si'. The status bar at the bottom indicates a 'BUILD SUCCESSFUL (total time: 0 seconds)'.

```
package TP01;

public class DeclaracionDeVariables {
    public static void main(String[] args) {
        String nombre = "Cristian Lahoz";
        int edad = 39;
        double altura = 1.75;
        boolean estudiante = true;

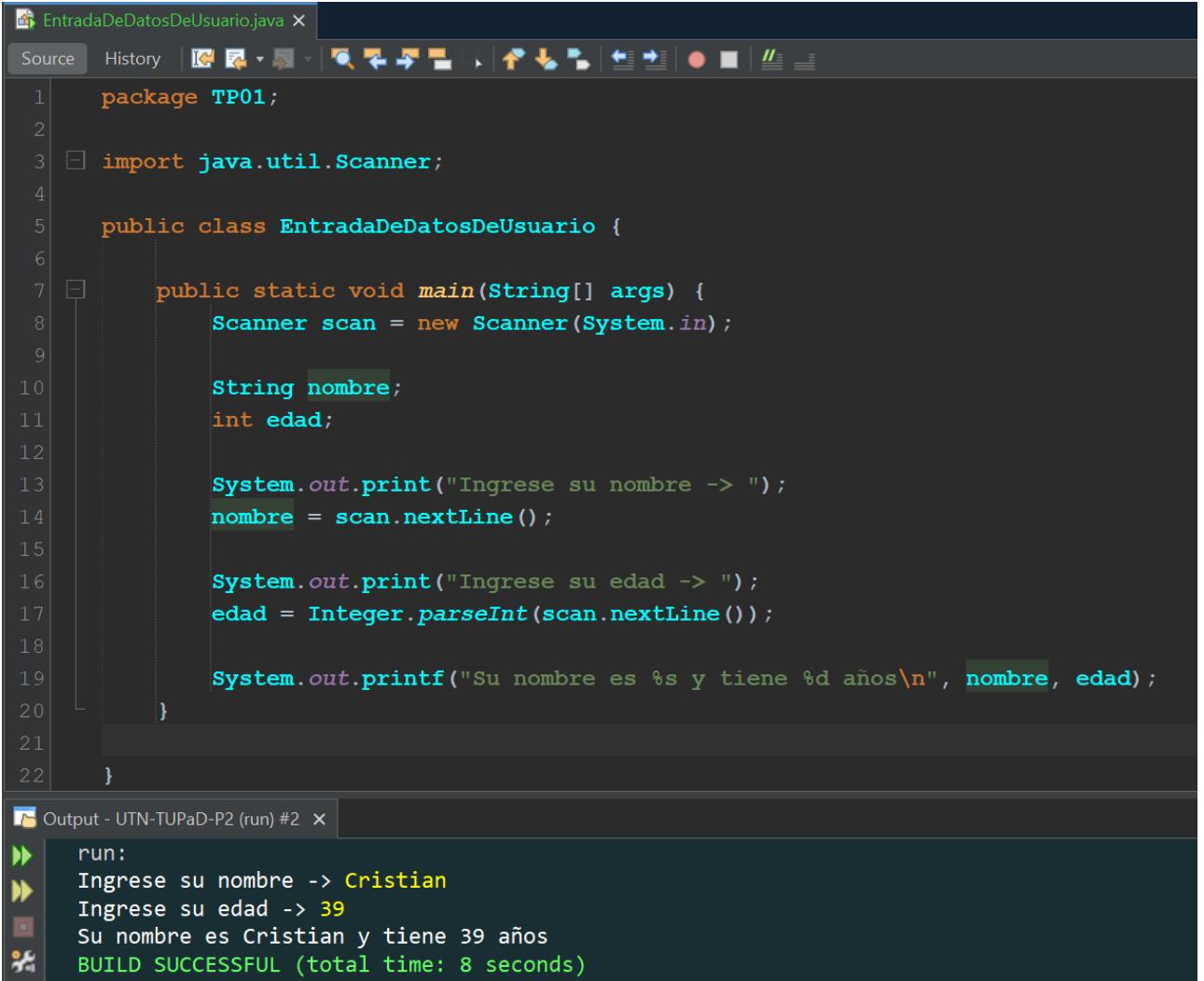
        System.out.println("Nombre: " + nombre);
        System.out.println("Edad: " + edad + " años");
        System.out.println("Altura: " + altura + " mts");
        System.out.println("Es estudiante: " + ((estudiante) ? "Si" : "No"));
    }
}
```

Output - UTN-TUPaD-P2 (run) ×

run:
Nombre: Cristian Lahoz
Edad: 39 años
Altura: 1.75 mts
Es estudiante: Si
BUILD SUCCESSFUL (total time: 0 seconds)

4. Entrada de datos del usuario

- Solicitar nombre y edad. Mostrar por consola usando **Scanner**:



The screenshot shows a Java code editor and a terminal window. The code editor window has a tab labeled 'EntradaDeDatosDeUsuario.java'. The code itself is as follows:

```

1 package TP01;
2
3 import java.util.Scanner;
4
5 public class EntradaDeDatosDeUsuario {
6
7     public static void main(String[] args) {
8         Scanner scan = new Scanner(System.in);
9
10        String nombre;
11        int edad;
12
13        System.out.print("Ingrese su nombre -> ");
14        nombre = scan.nextLine();
15
16        System.out.print("Ingrese su edad -> ");
17        edad = Integer.parseInt(scan.nextLine());
18
19        System.out.printf("Su nombre es %s y tiene %d años\n", nombre, edad);
20    }
21
22 }

```

The terminal window below has a tab labeled 'Output - UTN-TUPaD-P2 (run) #2'. It shows the output of the program when run with the inputs 'Cristian' and '39' respectively. The output is:

```

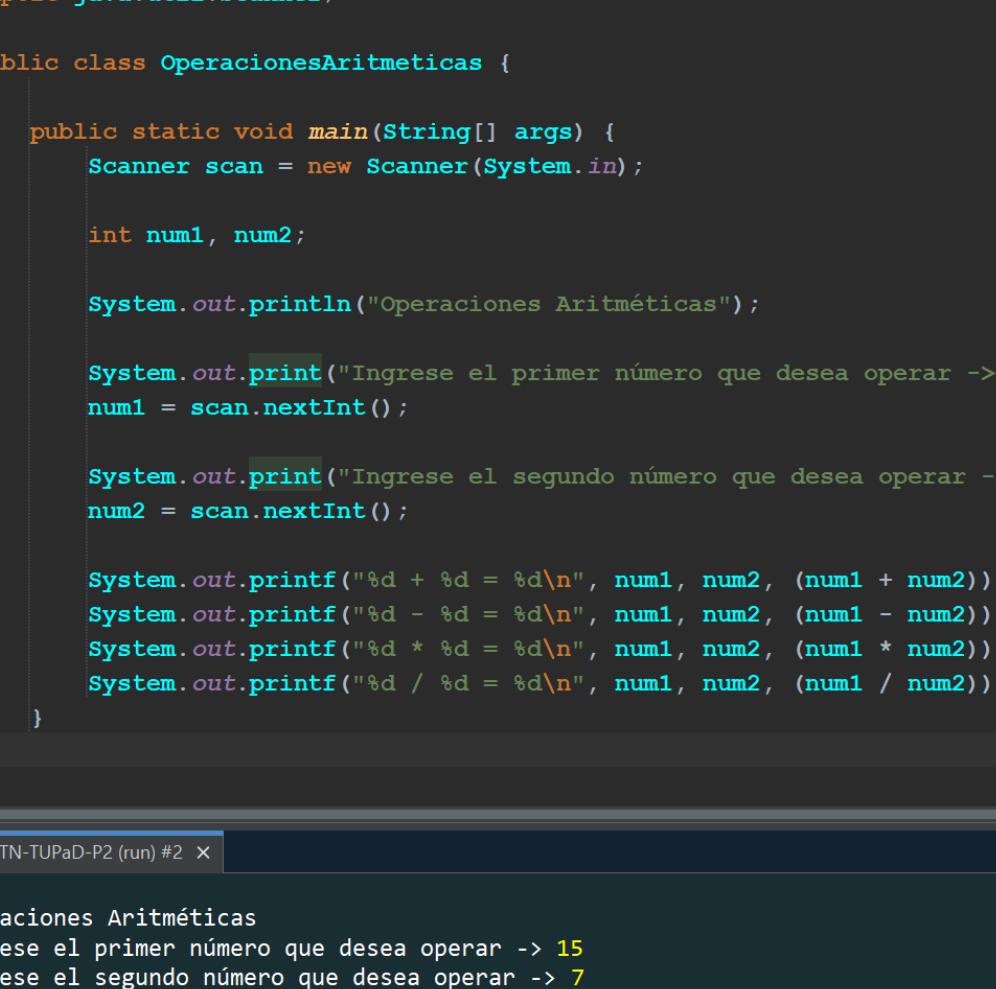
run:
Ingrese su nombre -> Cristian
Ingrese su edad -> 39
Su nombre es Cristian y tiene 39 años
BUILD SUCCESSFUL (total time: 8 seconds)

```

5. Operaciones aritméticas

- Solicitar dos enteros y realizar:
 - Suma
 - Resta
 - Multiplicación
 - División

- Mostrar resultado en consola:



The screenshot shows a Java IDE interface with the following details:

- Source Tab:** The current tab is "Source". The code is a Java program named "OperacionesAritmeticas". It imports the "java.util.Scanner" class and defines a main method that performs arithmetic operations on two integers entered by the user. The output of the program is displayed in the "Output" tab.
- Output Tab:** The current tab is "Output - UTN-TUPaD-P2 (run) #2". It shows the following execution:

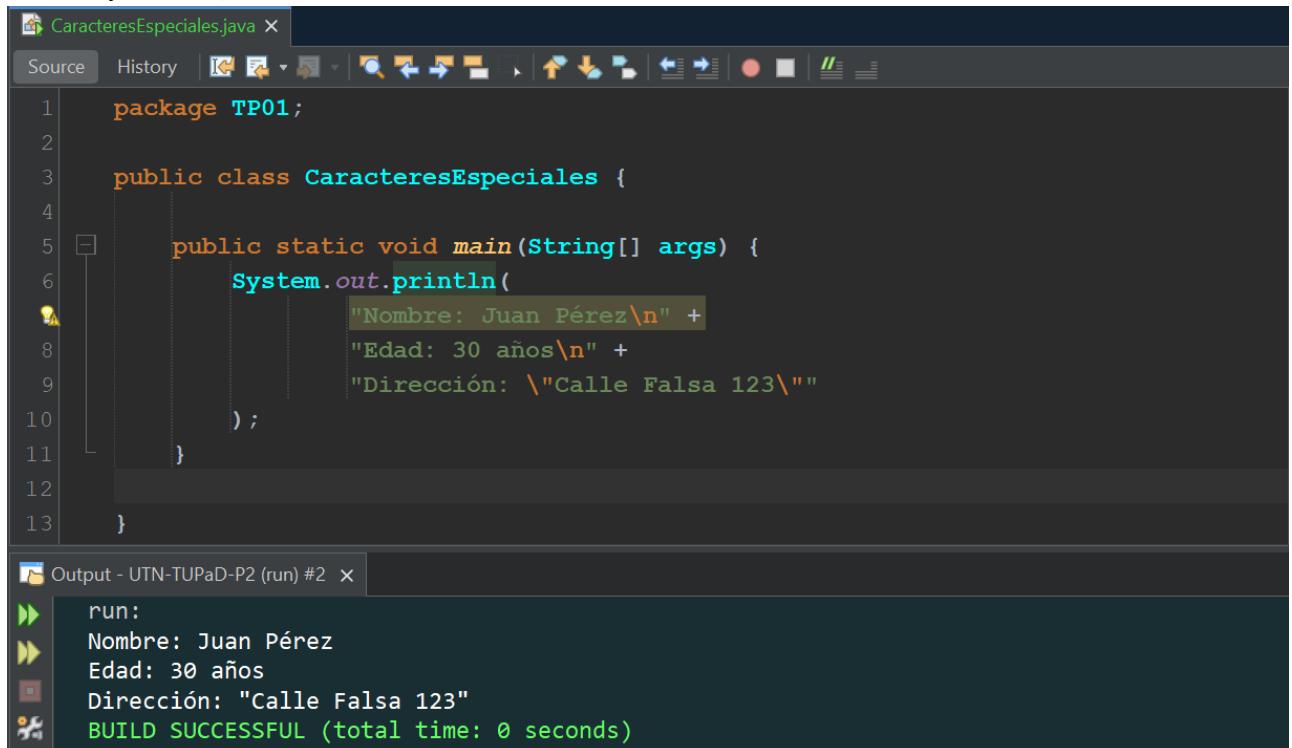
 - run:
 - Operaciones Aritméticas
 - Ingrese el primer número que desea operar -> 15
 - Ingrese el segundo número que desea operar -> 7
 - 15 + 7 = 22
 - 15 - 7 = 8
 - 15 * 7 = 105
 - 15 / 7 = 2
 - BUILD SUCCESSFUL (total time: 5 seconds)

6. Formato con caracteres especiales

- Imprimir por consola:

Nombre: Juan Pérez
Edad: 30 años
Dirección: "Calle Falsa 123"

- Usar `\n` y `\"` en `System.out.println()`:



```

1 package TP01;
2
3 public class CaracteresEspeciales {
4
5     public static void main(String[] args) {
6         System.out.println(
7             "Nombre: Juan Pérez\n" +
8             "Edad: 30 años\n" +
9             "Dirección: \"Calle Falsa 123\""
10        );
11    }
12 }
13

```

Output - UTN-TUPaD-P2 (run) #2

```

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

```

7. Expresiones e instrucciones

- Analizar este código y distinguir expresiones/instrucciones:

```

int x = 10;           // Instrucción (declaración y asignación)
x = x + 5;           // Instrucción (asignación de una expresión)
System.out.println(x); // Instrucción (llamada a método)

```

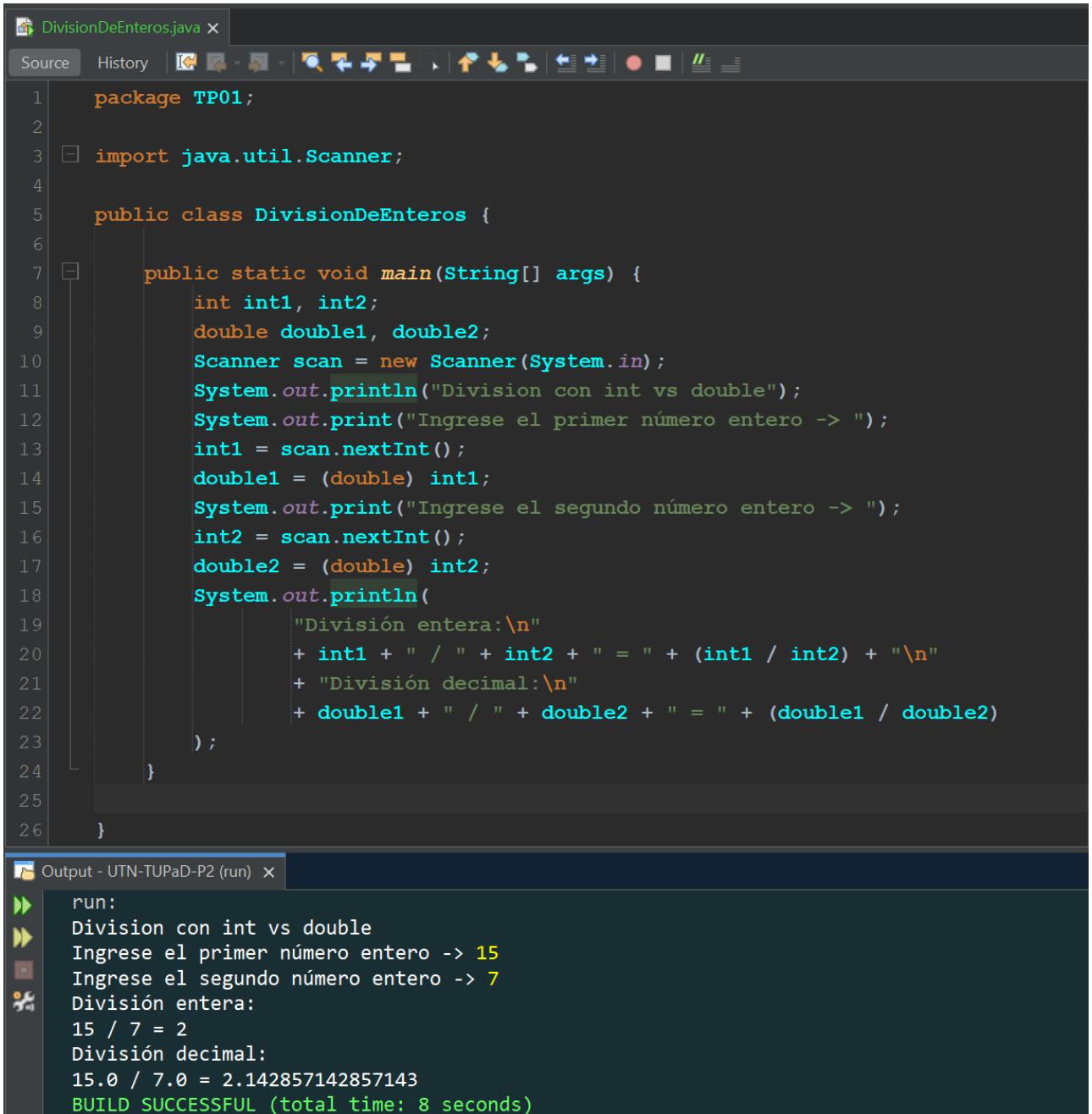
- Explicar diferencia:

- Expresión es cualquier fragmento de código que produce un valor. Puede estar sola o formar parte de una instrucción.
- Instrucción es una línea de código que realiza una acción específica, como declarar una variable, asignar un valor o llamar a una función.

8. Conversión de tipos

- Dividir dos enteros ingresados.

- Repetir usando **double**:



The screenshot shows a Java code editor with the file `DivisionDeEnteros.java` and a terminal window below it.

Code Editor (DivisionDeEnteros.java):

```

1 package TP01;
2
3 import java.util.Scanner;
4
5 public class DivisionDeEnteros {
6
7     public static void main(String[] args) {
8         int int1, int2;
9         double double1, double2;
10        Scanner scan = new Scanner(System.in);
11        System.out.println("División con int vs double");
12        System.out.print("Ingrese el primer número entero -> ");
13        int1 = scan.nextInt();
14        double1 = (double) int1;
15        System.out.print("Ingrese el segundo número entero -> ");
16        int2 = scan.nextInt();
17        double2 = (double) int2;
18        System.out.println(
19            "División entera:\n" +
20            int1 + " / " + int2 + " = " + (int1 / int2) + "\n" +
21            "División decimal:\n" +
22            double1 + " / " + double2 + " = " + (double1 / double2)
23        );
24    }
25
26 }

```

Terminal (Output - UTN-TUPaD-P2 (run)):

```

run:
Division con int vs double
Ingrese el primer número entero -> 15
Ingrese el segundo número entero -> 7
División entera:
15 / 7 = 2
División decimal:
15.0 / 7.0 = 2.142857142857143
BUILD SUCCESSFUL (total time: 8 seconds)

```

- Comparar resultados: La división de enteros da como resultado un numero entero. La división de decimales (utilizando los mismos números) da como resultado un número decimal.

9. Corrección de errores

- Código con error:

```

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);
    }
}

```

```
    }
}
```

- Explicar el error y corregirlo: `scanner.nextInt()` lee un número entero, pero se está intentando guardar en una variable de tipo `String`, lo cual es incompatible. Además, no se espera que alguien ingrese un número como nombre. Para corregirlo se reemplazó `scanner.nextInt()` por `scanner.nextLine()`, que permite leer una línea completa de texto y es adecuado para leer nombres, frases, etc.

```
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
        String nombre = scanner.nextLine();           // CORREGIDO
        System.out.println("Hola, " + nombre);
    }
}
```

10. Prueba de escritorio

- Analizar y completar ejecución del siguiente código:

```
1  public class PruebaEscritorio {
2      public static void main(String[] args) {
3          int a = 5;
4          int b = 2;
5          int resultado = a / b;
6          System.out.println("Resultado: " + resultado);
7      }
8  }
```

Línea	int a	int b	int resultado
1	<sin_definir>	<sin_definir>	<sin_definir>
2	<sin_definir>	<sin_definir>	<sin_definir>
3	5	<sin_definir>	<sin_definir>
4	5	2	<sin_definir>
5	5	2	2
6	5	2	2

Línea	int a	int b	int resultado
7	5	2	2
8	5	2	2

Conclusiones esperadas

- Reforzar conceptos fundamentales de Java.
- Familiarizarse con la estructura básica de un programa.
- Aprender a depurar errores comunes.
- Comprender conversiones de tipo y expresiones.
- Manejar entradas/salidas y variables.
- Usar NetBeans y herramientas de depuración.