

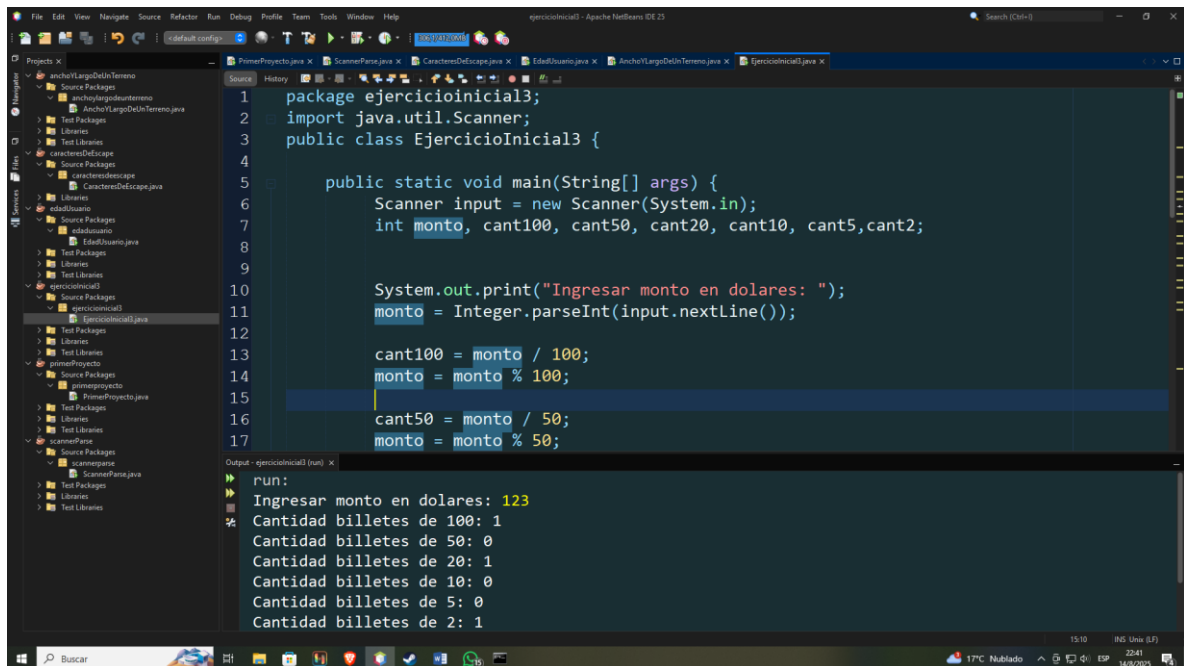
## 1. Verificar que tienes instalado Java JDK y NetBeans

- a) Confirma que tienes Java JDK instalado ejecutando el siguiente comando en la terminal: **java -versión**

R:

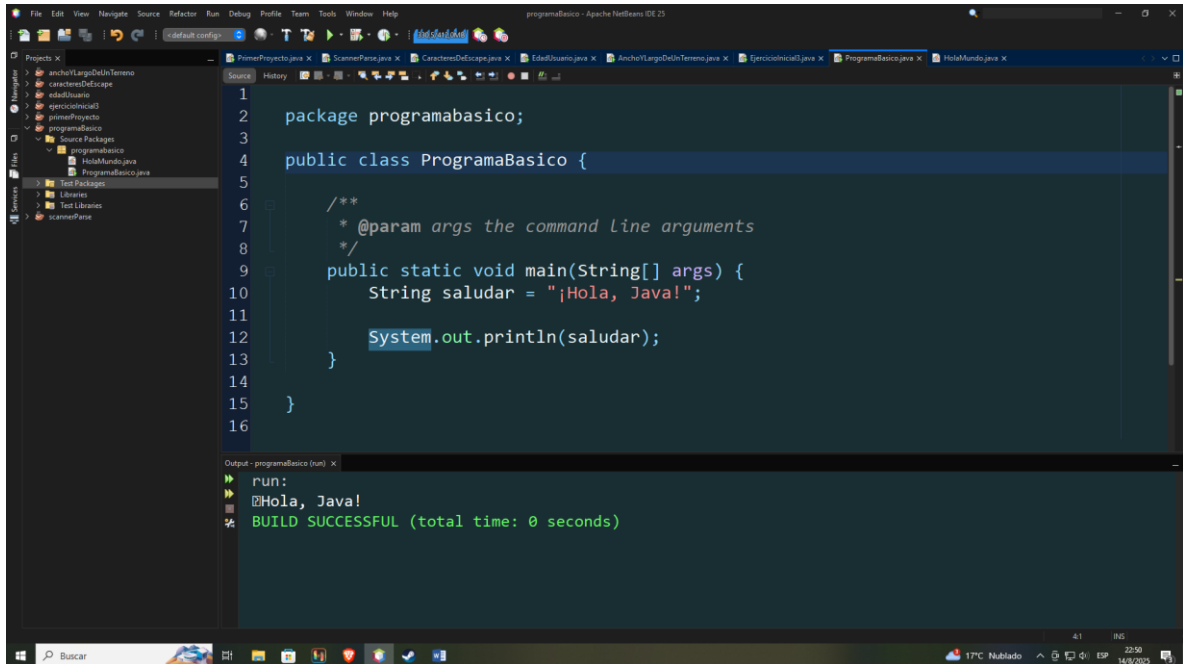
```
C:\Users\Willy>java -version
java version "24.0.2" 2025-07-15
Java(TM) SE Runtime Environment (build 24.0.2+12-54)
Java HotSpot(TM) 64-Bit Server VM (build 24.0.2+12-54, mixed mode, sharing)
```

- 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.

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



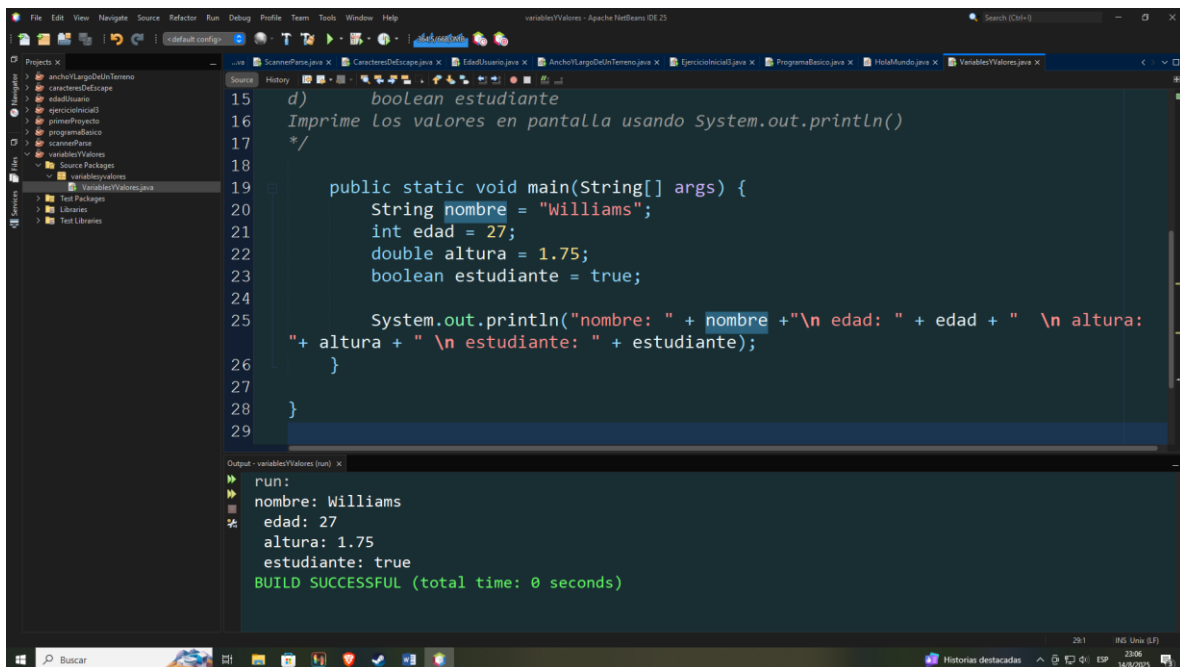
```
1 package programabasico;
2
3
4 public class ProgramaBasico {
5
6     /**
7      * @param args the command line arguments
8      */
9     public static void main(String[] args) {
10         String saludar = "¡Hola, Java!";
11
12         System.out.println(saludar);
13     }
14 }
15
16
```

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()

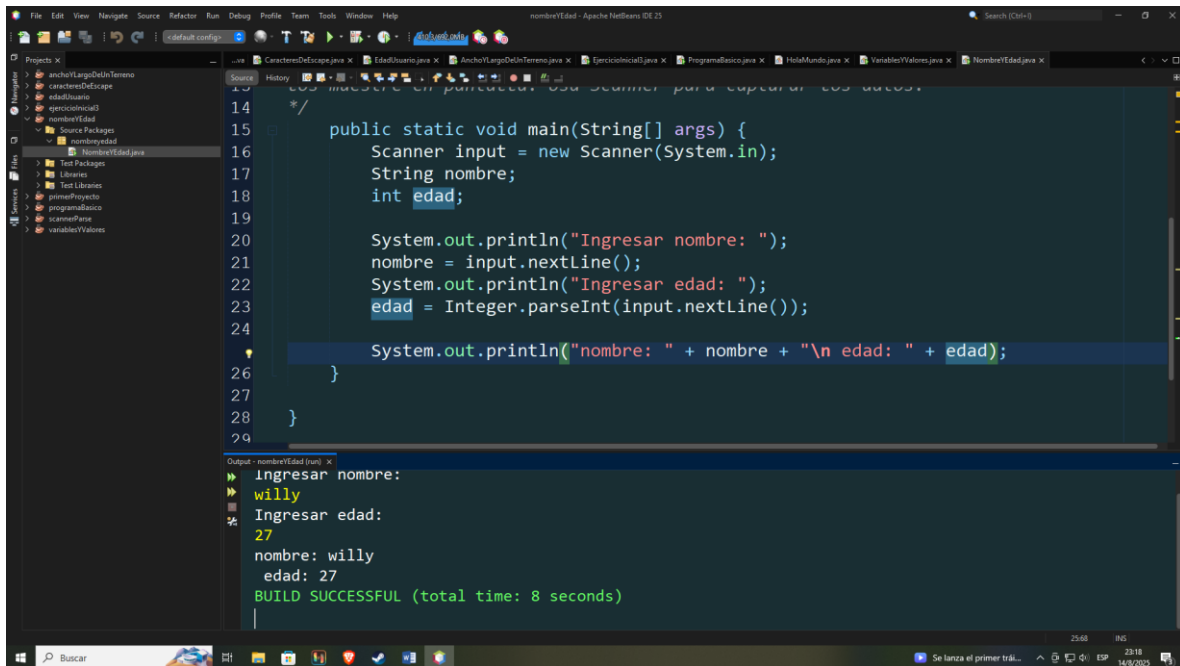


```
15 d) boolean estudiante
16 Imprime los valores en pantalla usando System.out.println()
17 */
18
19 public static void main(String[] args) {
20     String nombre = "Williams";
21     int edad = 27;
22     double altura = 1.75;
23     boolean estudiante = true;
24
25     System.out.println("nombre: " + nombre + "\n edad: " + edad + " \n altura:
26     "+ altura + " \n estudiante: " + estudiante);
27 }
28 }
29
```

Output - variablesValores (run)

```
run:
nombre: Williams
edad: 27
altura: 1.75
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.



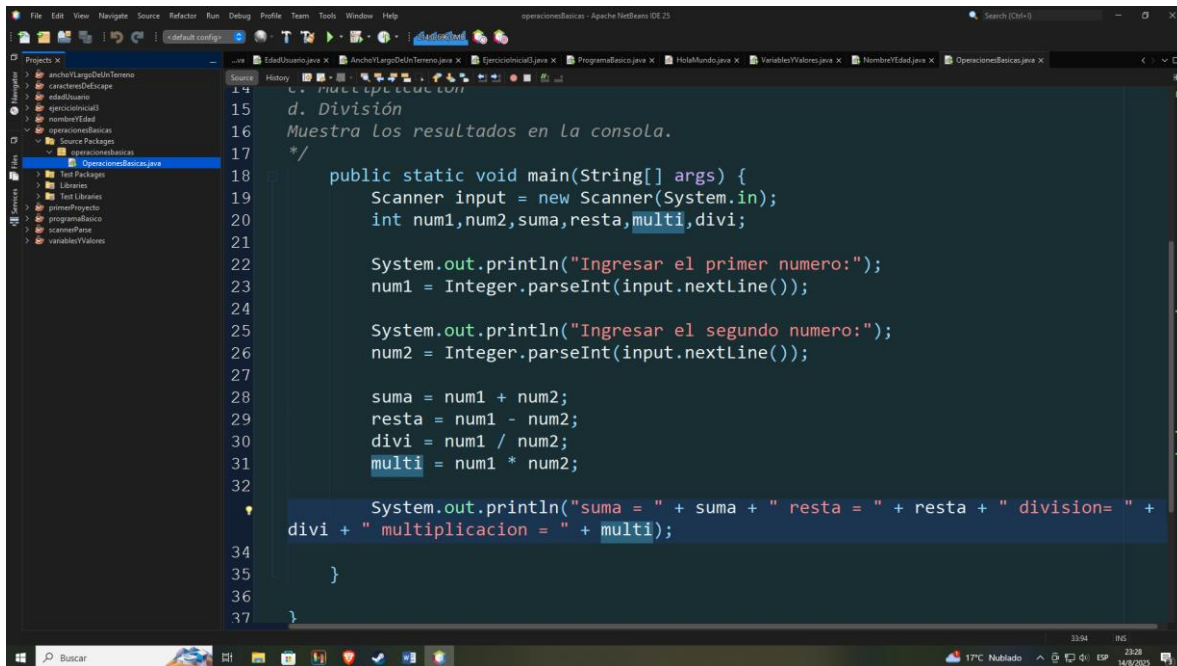
```
14 //
15
16 public static void main(String[] args) {
17     Scanner input = new Scanner(System.in);
18     String nombre;
19     int edad;
20
21     System.out.println("Ingresar nombre: ");
22     nombre = input.nextLine();
23     System.out.println("Ingresar edad: ");
24     edad = Integer.parseInt(input.nextLine());
25
26     System.out.println("nombre: " + nombre + "\n edad: " + edad);
27 }
28
29 }
```

Output - nombreYEdad (run)

```
>> Ingresar nombre:
willy
Ingresar edad:
27
nombre: willy
edad: 27
BUILD SUCCESSFUL (total time: 8 seconds)
```

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

- 1) Suma
- 2) Resta
- 3) Multiplicación
- 4) División Muestra los resultados en la consola.

A screenshot of an IDE window titled 'operacionesBasicas - Apache NetBeans IDE 2.5'. The code is in a file named 'OperacionesBasicas.java'. It contains a Java program that prompts the user for two integers and performs four operations: addition, subtraction, multiplication, and division. The results are printed to the console. The code is as follows:

```
15  d. División
16  Muestra los resultados en la consola.
17  */
18  public static void main(String[] args) {
19      Scanner input = new Scanner(System.in);
20      int num1,num2,suma,resta,multi,divi;
21
22      System.out.println("Ingresar el primer numero:");
23      num1 = Integer.parseInt(input.nextLine());
24
25      System.out.println("Ingresar el segundo numero:");
26      num2 = Integer.parseInt(input.nextLine());
27
28      suma = num1 + num2;
29      resta = num1 - num2;
30      divi = num1 / num2;
31      multi = num1 * num2;
32
33      System.out.println("suma = " + suma + " resta = " + resta + " division= " +
34      divi + " multiplicacion = " + multi);
35  }
36
37  }
```

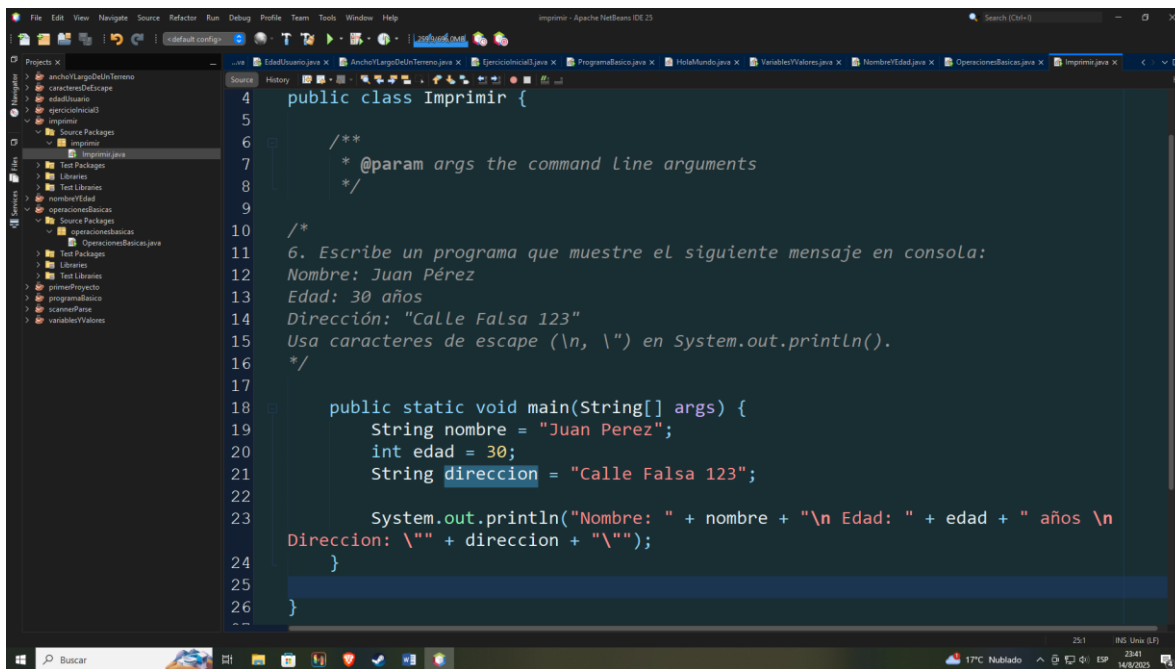
6. Escribe un programa que muestre el siguiente mensaje en consola:

**Nombre: Juan Pérez**

**Edad: 30 años**

**Dirección: "Calle Falsa 123"**

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



```
4 public class Imprimir {
5
6     /**
7      * @param args the command line arguments
8      */
9
10    /**
11     * 6. Escribe un programa que muestre el siguiente mensaje en consola:
12     * Nombre: Juan Pérez
13     * Edad: 30 años
14     * Dirección: "Calle Falsa 123"
15     * Usa caracteres de escape (\n, \") en System.out.println().
16     */
17
18    public static void main(String[] args) {
19        String nombre = "Juan Perez";
20        int edad = 30;
21        String direccion = "Calle Falsa 123";
22
23        System.out.println("Nombre: " + nombre + "\n Edad: " + edad + " años \n
24        Direccion: \"" + direccion + "\"");
25    }
26 }
```

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
```

R:

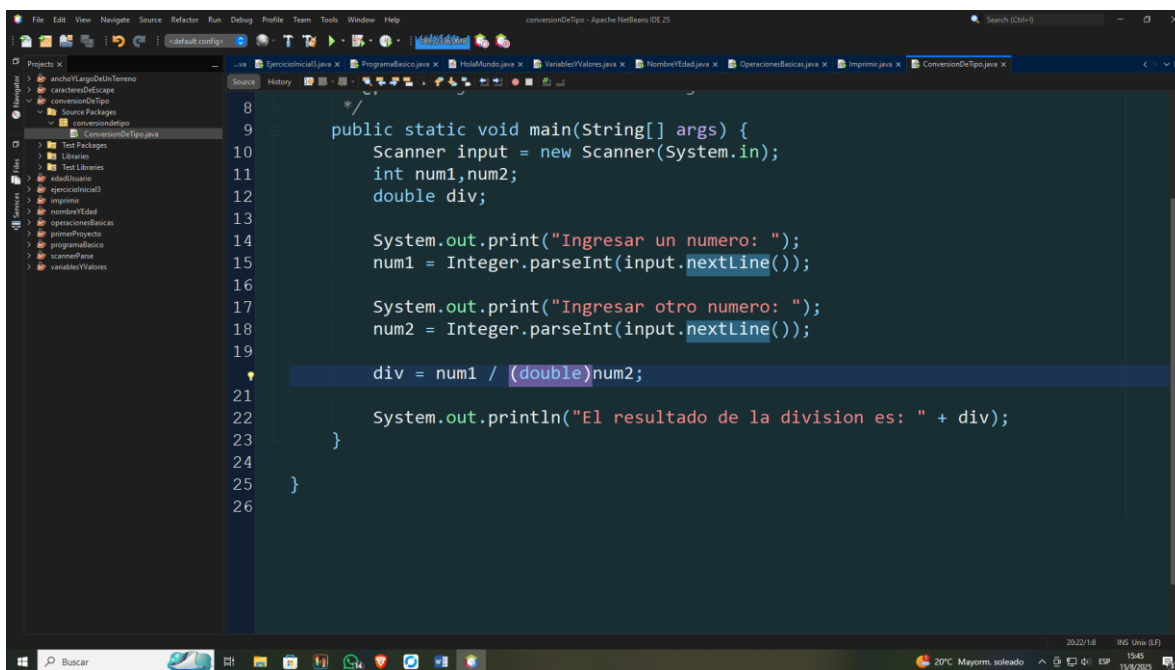
Línea 1 (int x = 10;): Instrucción (declaración y asignación).

Línea 2 (x = x + 5;): Instrucción que contiene la expresión x + 5.

Línea 3 (System.out.println(x);): Instrucción que contiene la expresión x.

8. Manejar conversiones de tipo y división en Java.

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

A screenshot of an IDE window titled 'conversionDeTipo - Apache NetBeans IDE 21'. The code editor shows a Java program with the following content:

```
8  /*
9  */
10 public static void main(String[] args) {
11     Scanner input = new Scanner(System.in);
12     int num1,num2;
13     double div;
14
15     System.out.print("Ingresar un numero: ");
16     num1 = Integer.parseInt(input.nextLine());
17
18     System.out.print("Ingresar otro numero: ");
19     num2 = Integer.parseInt(input.nextLine());
20
21     div = num1 / (double)num2;
22
23     System.out.println("El resultado de la division es: " + div);
24 }
25
26 }
```

The code is color-coded: comments are grey, keywords are blue, literals are red, and identifiers are black. The IDE interface includes a menu bar, a toolbar, a project explorer on the left, and a status bar at the bottom showing '20°C Mayorm. soleado' and '15/9/2023'.

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);

    }

}
```

**R:**

El error estaba en que se usaba `scanner.nextInt()` para leer el nombre.

Cambiando a `scanner.nextLine()` el programa puede leer texto completo sin problemas.

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	a	b	división	Salida por pantalla
1	<sin_definir>	<sin_definir>	<sin_definir>	-
2	<sin_definir>	<sin_definir>	<sin_definir>	-
3	5	<sin_definir>	<sin_definir>	-
4	5	2	2	-
5	5	2	2	Resultado: 2