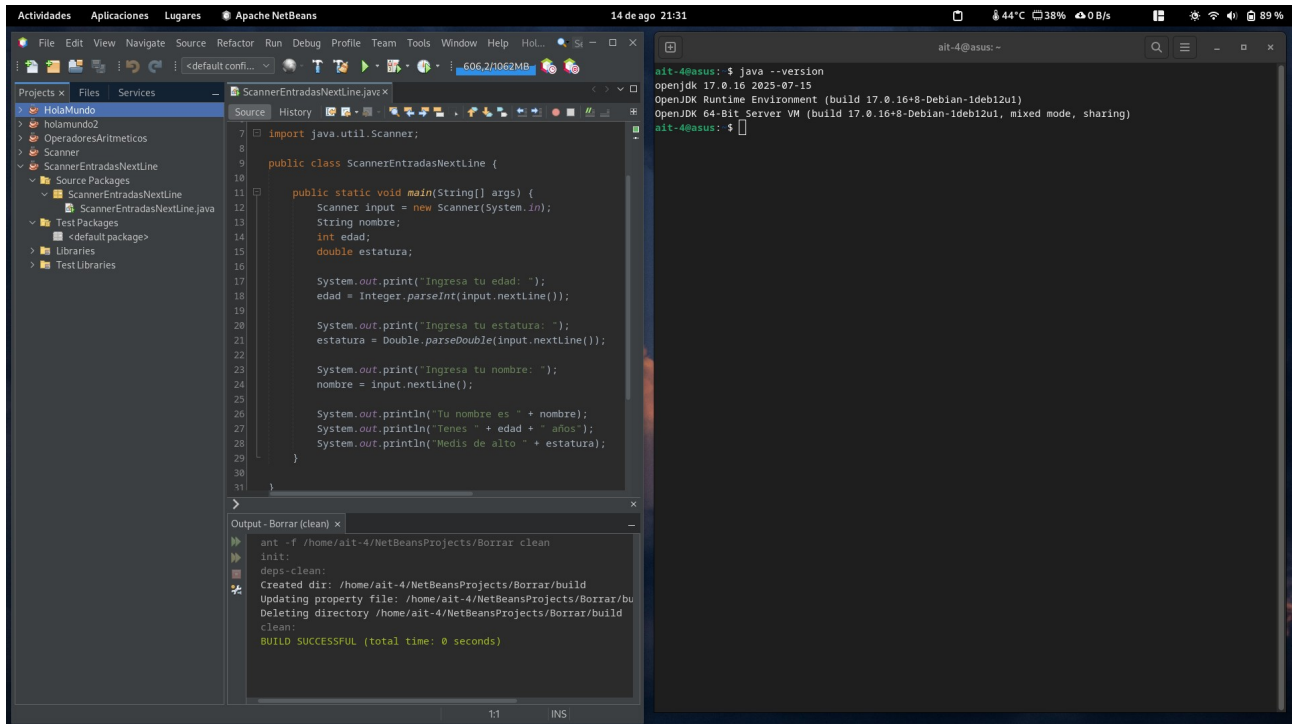


## PROGRAMACIÓN II

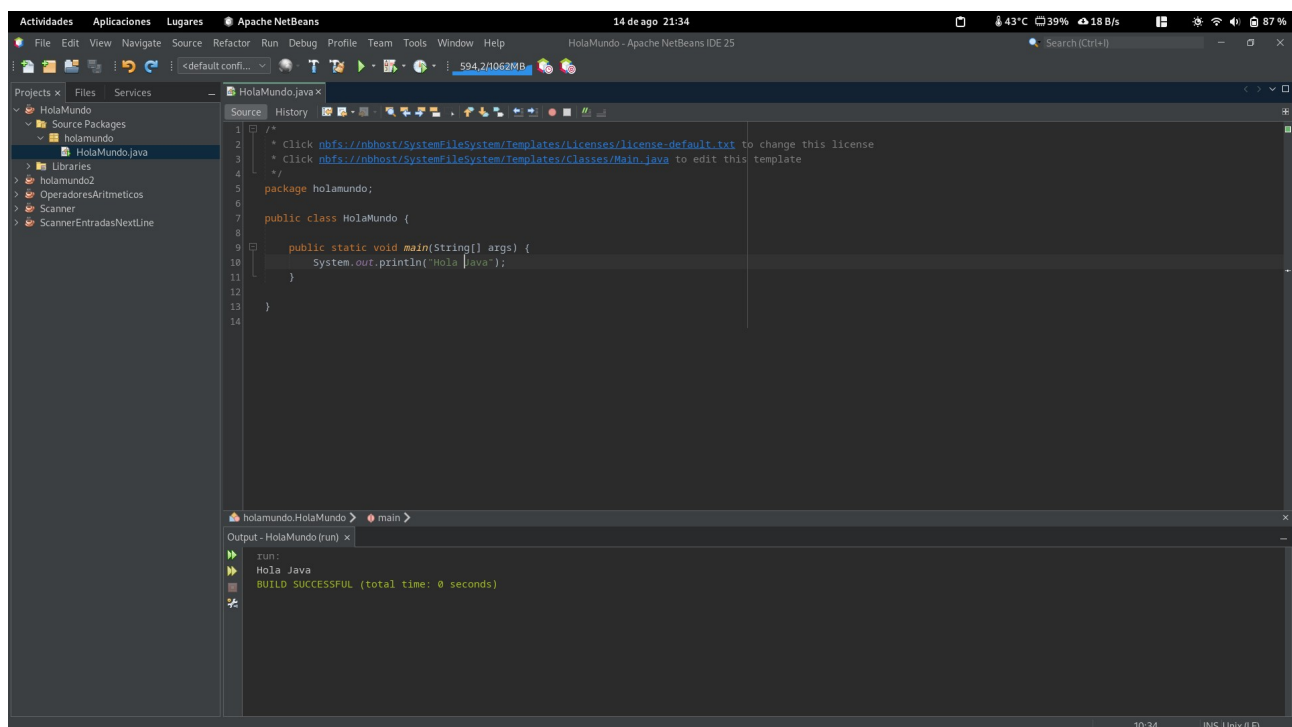
### Trabajo Práctico 1: Introducción a Java

#### Casos prácticos:

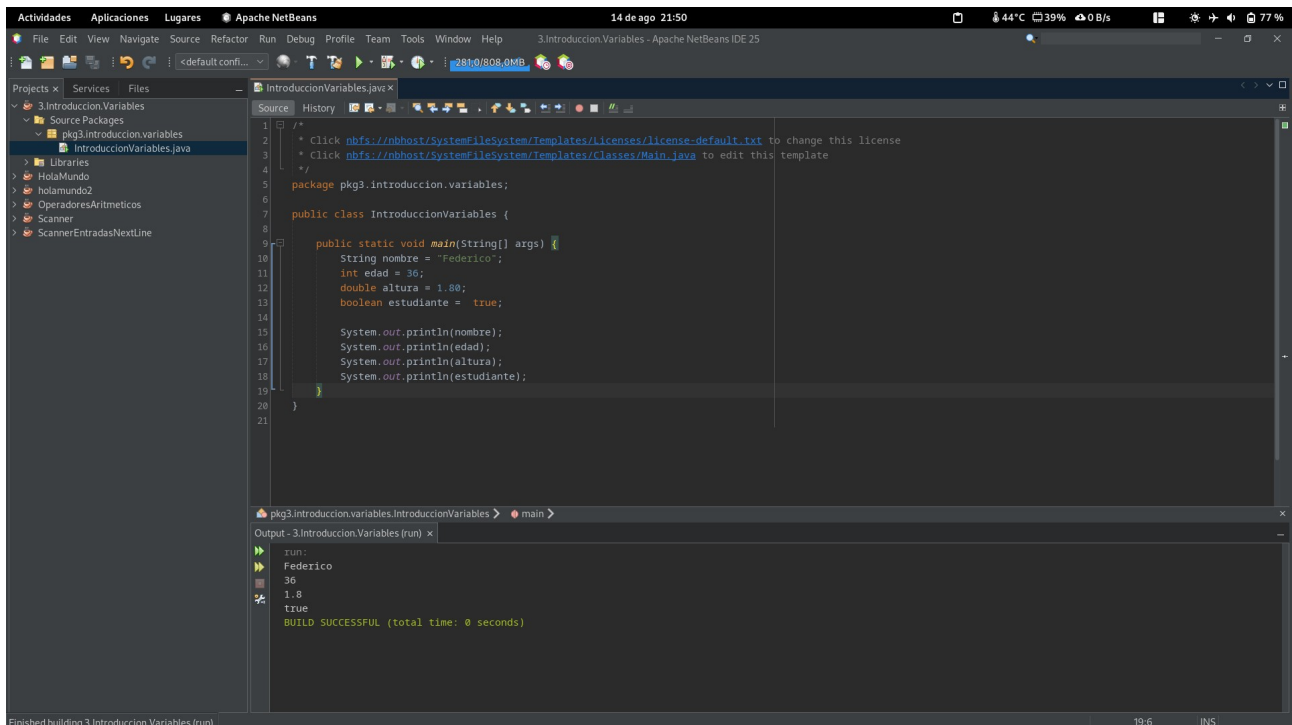
##### 1. Verificar que tienes instalado Java JDK y NetBeans.



##### 2. Escribir y ejecutar un programa básico en Java.



3. Crea un programa que declare las siguientes variables con valores asignados: String nombre, int edad, double altura y boolean estudiante.



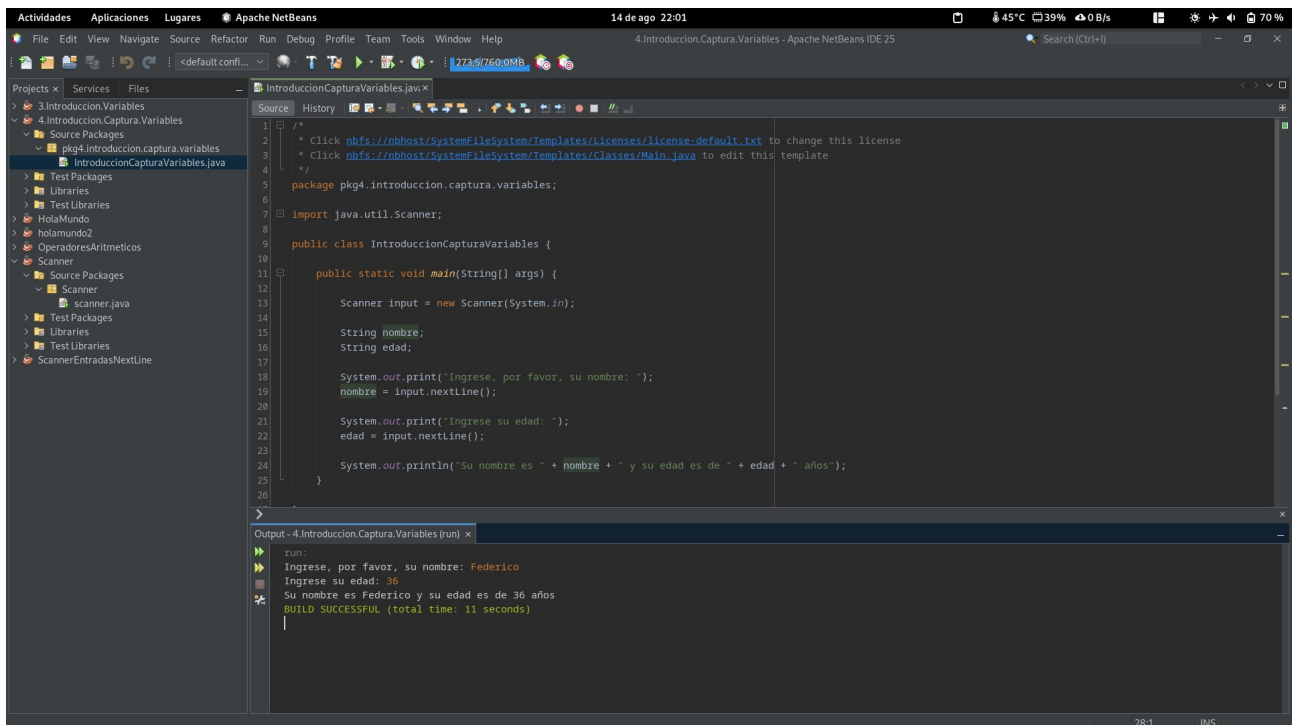
The screenshot shows the Apache NetBeans IDE interface. The main editor displays the source code for `IntroduccionVariables.java` in the `pkg3.introduccion.variables` package. The code defines a public class `IntroduccionVariables` with a `main` method that declares and assigns values to four variables: `String nombre = "Federico";`, `int edad = 36;`, `double altura = 1.8;`, and `boolean estudiante = true;`. These variables are then printed to the console using `System.out.println`. The left sidebar shows the project structure with the file `IntroduccionVariables.java` selected. The bottom output window shows the successful execution of the program, displaying the values of the variables: `Federico`, `36`, `1.8`, and `true`. The build was successful with a total time of 0 seconds.

```
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 pkg3.introduccion.variables;
6
7  public class IntroduccionVariables {
8
9      public static void main(String[] args) {
10         String nombre = "Federico";
11         int edad = 36;
12         double altura = 1.8;
13         boolean estudiante = true;
14
15         System.out.println(nombre);
16         System.out.println(edad);
17         System.out.println(altura);
18         System.out.println(estudiante);
19     }
20 }
21
```

Output - 3.Introduccion.Variables (run) x

```
run:
Federico
36
1.8
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.



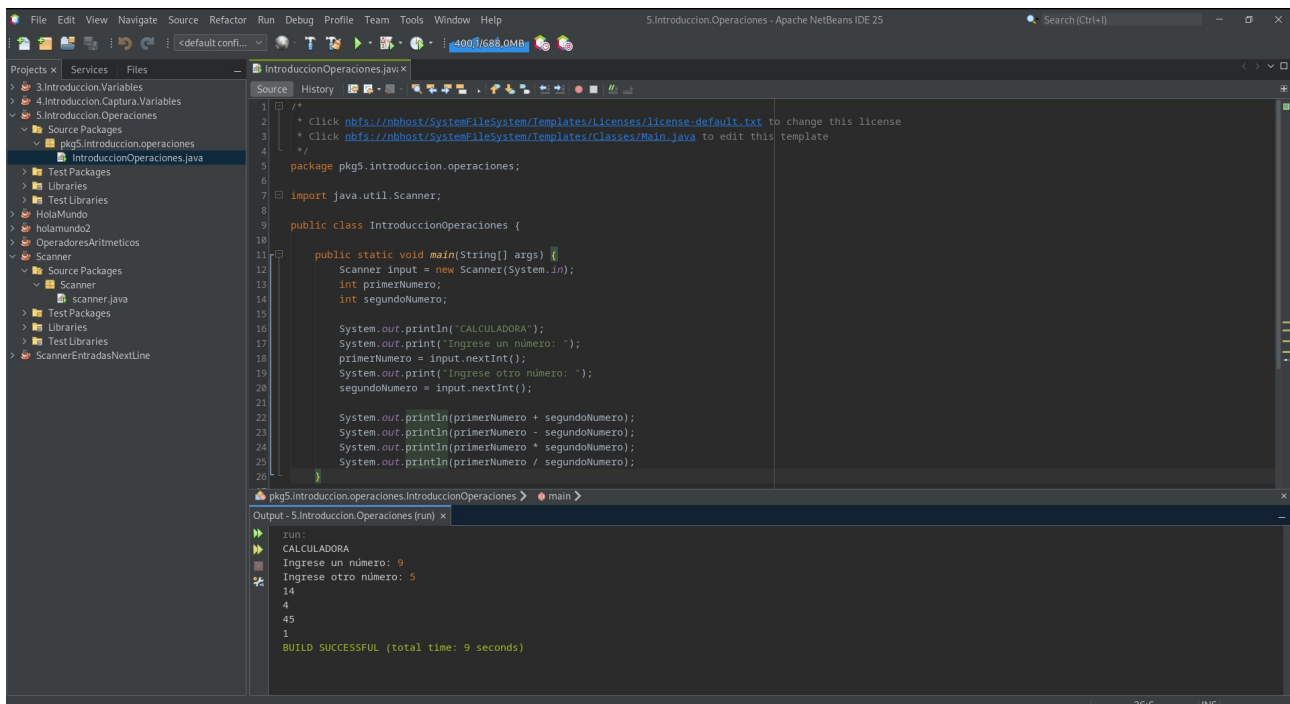
The screenshot shows the Apache NetBeans IDE interface. The main editor displays the source code for `IntroduccionCapturaVariables.java` in the `pkg4.introduccion.captura.variables` package. The code imports `java.util.Scanner` and defines a public class `IntroduccionCapturaVariables` with a `main` method. Inside the `main` method, a `Scanner` object is created to read input from `System.in`. The program prompts the user to enter their name and age, reads the input using `input.nextLine()`, and then prints the entered values back to the console. The output window shows the successful execution of the program, displaying the user input: `Ingrese, por favor, su nombre: Federico` and `Ingrese su edad: 36`, followed by the confirmation message: `Su nombre es Federico y su edad es de 36 años`. The build was successful with a total time of 11 seconds.

```
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 pkg4.introduccion.captura.variables;
6
7  import java.util.Scanner;
8
9  public class IntroduccionCapturaVariables {
10
11      public static void main(String[] args) {
12         Scanner input = new Scanner(System.in);
13
14         String nombre;
15         String edad;
16
17         System.out.print("Ingrese, por favor, su nombre: ");
18         nombre = input.nextLine();
19
20         System.out.print("Ingrese su edad: ");
21         edad = input.nextLine();
22
23         System.out.println("Su nombre es " + nombre + " y su edad es de " + edad + " años");
24     }
25 }
26
```

Output - 4.Introduccion.Captura.Variables (run) x

```
run:
Ingrese, por favor, su nombre: Federico
Ingrese su edad: 36
Su nombre es Federico y su edad es de 36 años
BUILD SUCCESSFUL (total time: 11 seconds)
```

5. Escribe un programa que solicite dos números enteros y realice las siguientes operaciones: Suma, Resta, Multiplicación y División.

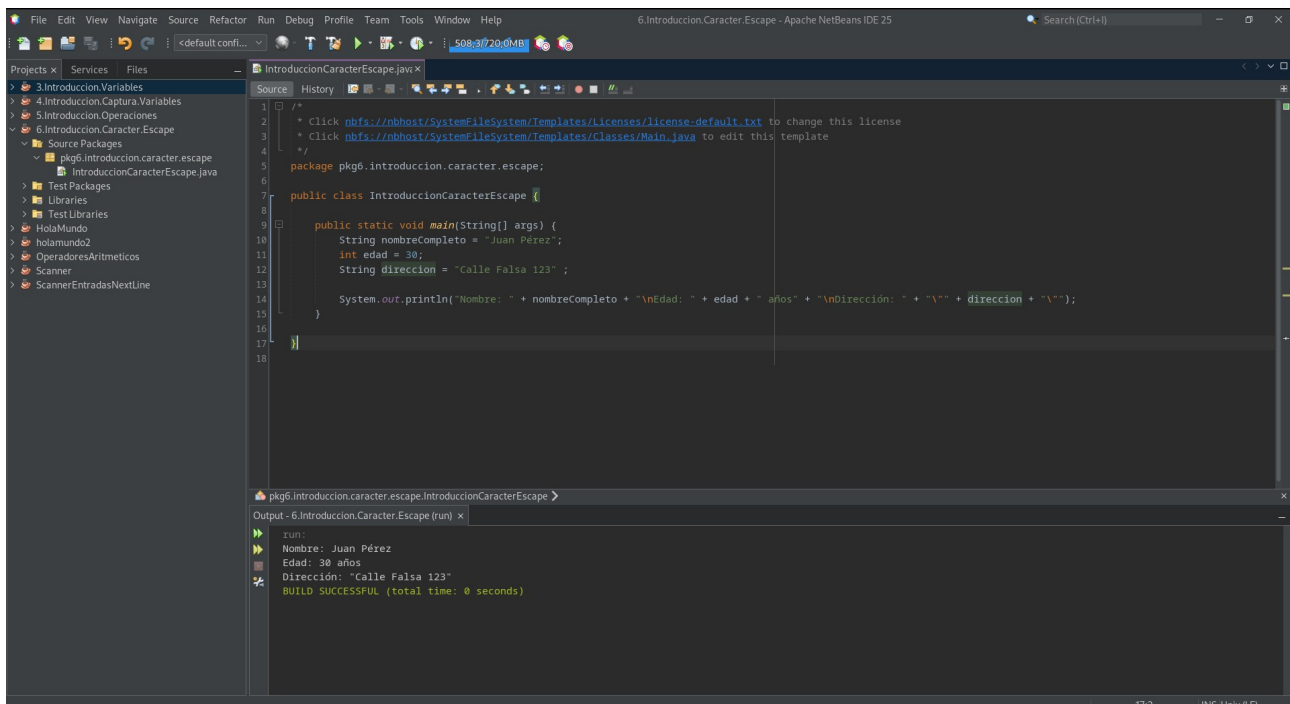


The screenshot shows the Apache NetBeans IDE with a project named '5.Introduccion.Operaciones'. The main editor displays the file 'IntroduccionOperaciones.java'. The code defines a package 'pkg5.introduccion.operaciones', imports 'java.util.Scanner', and creates a public class 'IntroduccionOperaciones'. The 'main' method uses a Scanner to take two integer inputs from the user, then calculates and prints the sum, difference, product, and quotient of the two numbers. The output window shows the program running successfully, with the user inputting 9 and 5, resulting in the following output: 'run: CALCULADORA', 'Ingrese un número: 9', 'Ingrese otro número: 5', '14', '4', '45', '1', and 'BUILD SUCCESSFUL (total time: 9 seconds)'.

```
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 pkg5.introduccion.operaciones;
6
7  import java.util.Scanner;
8
9  public class IntroduccionOperaciones {
10
11     public static void main(String[] args) {
12         Scanner input = new Scanner(System.in);
13         int primerNumero;
14         int segundoNumero;
15
16         System.out.println("CALCULADORA");
17         System.out.print("Ingrese un número: ");
18         primerNumero = input.nextInt();
19         System.out.print("Ingrese otro número: ");
20         segundoNumero = input.nextInt();
21
22         System.out.println(primerNumero + segundoNumero);
23         System.out.println(primerNumero - segundoNumero);
24         System.out.println(primerNumero * segundoNumero);
25         System.out.println(primerNumero / segundoNumero);
26     }
27 }
```

run: CALCULADORA  
Ingrese un número: 9  
Ingrese otro número: 5  
14  
4  
45  
1  
BUILD SUCCESSFUL (total time: 9 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().



The screenshot shows the Apache NetBeans IDE with a project named '6.introduccion.Caracter.Escape'. The main editor displays the file 'IntroduccionCaracterEscape.java'. The code defines a package 'pkg6.introduccion.caracter.escape', creates a public class 'IntroduccionCaracterEscape', and includes a 'main' method. The 'main' method declares three String variables: 'nombreCompleto' with the value 'Juan Pérez', 'edad' with the value 30, and 'direccion' with the value 'Calle Falsa 123'. It then uses 'System.out.println' to print these values, employing escape characters for the newline and double quote. The output window shows the program running successfully, displaying the output: 'run: Nombre: Juan Pérez', 'Edad: 30 años', 'Dirección: "Calle Falsa 123"', and 'BUILD SUCCESSFUL (total time: 0 seconds)'.

```
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 pkg6.introduccion.caracter.escape;
6
7  public class IntroduccionCaracterEscape {
8
9     public static void main(String[] args) {
10         String nombreCompleto = "Juan Pérez";
11         int edad = 30;
12         String direccion = "Calle Falsa 123";
13
14         System.out.println("Nombre: " + nombreCompleto + "\nEdad: " + edad + " años" + "\nDirección: " + "\"" + direccion + "\"");
15     }
16
17 }
18 }
```

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
```

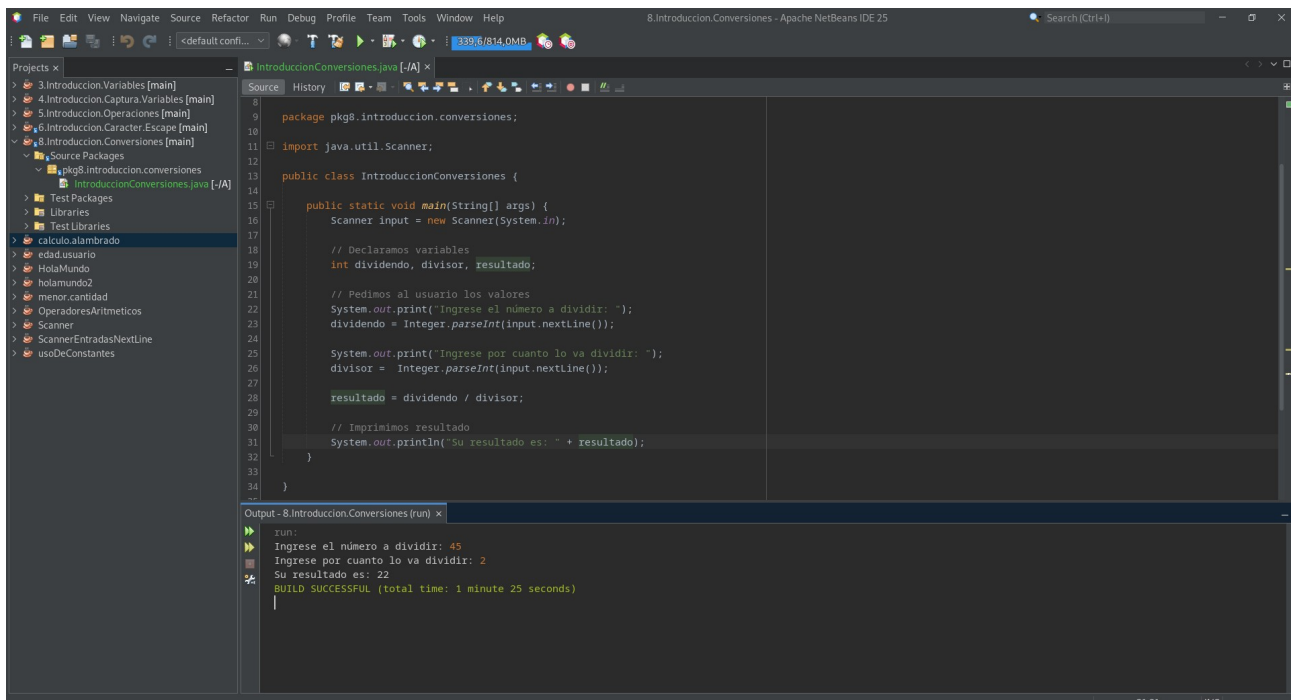
Todas las líneas son instrucciones. Son instrucciones porque existen cambios en el código, sea de, variables, valores, etc. Son sentencias que se pueden interpretar. Una expresión en este caso sería **x + 5** solamente que sería una operación pero que para el lenguaje JAVA no puede interpretar generando un error en el código.

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.

Uso de **int**:



The screenshot displays the Apache NetBeans IDE interface. The main editor window shows a Java file named `IntroduccionConversiones.java` with the following code:

```
8 package pkg8.introduccion.conversiones;
9
10
11 import java.util.Scanner;
12
13 public class IntroduccionConversiones {
14
15     public static void main(String[] args) {
16         Scanner input = new Scanner(System.in);
17
18         // Declaramos variables
19         int dividendo, divisor, resultado;
20
21         // Pedimos al usuario los valores
22         System.out.print("Ingrese el número a dividir: ");
23         dividendo = Integer.parseInt(input.nextLine());
24
25         System.out.print("Ingrese por cuanto lo va dividir: ");
26         divisor = Integer.parseInt(input.nextLine());
27
28         resultado = dividendo / divisor;
29
30         // Imprimimos resultado
31         System.out.println("Su resultado es: " + resultado);
32     }
33 }
34
```

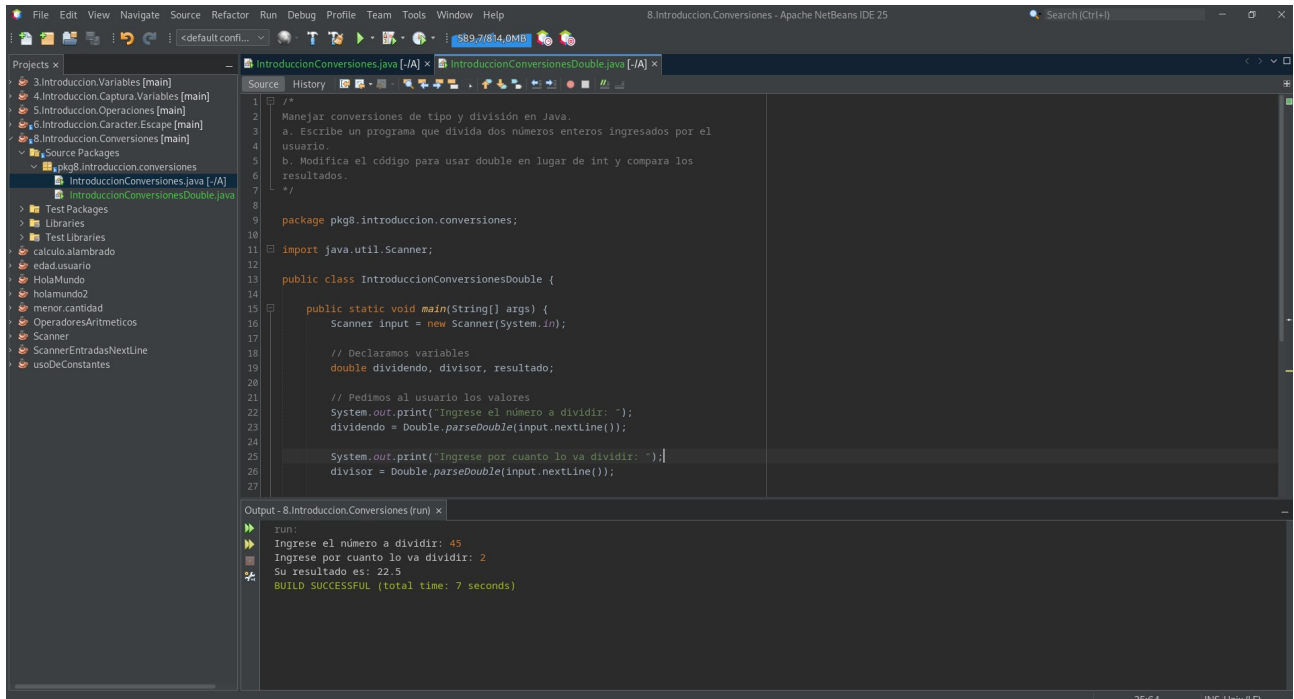
The left sidebar shows the project structure with the following items:

- 3.Introduccion.Variables [main]
- 4.Introduccion.Captura.Variables [main]
- 5.Introduccion.Operaciones [main]
- 6.Introduccion.Caracter.Escape [main]
- 8.Introduccion.Conversiones [main]
- Source Packages
  - pkg8.introduccion.conversiones
    - IntroduccionConversiones.java [-/A]
- Test Packages
- Libraries
- Test Libraries
  - calculo.alambrado
  - edad.usuario
  - HolaMundo
  - holamundo2
  - menor.cantidad
  - OperadoresAritmeticos
  - Scanner
  - ScannerEntradasNextLine
  - usoDeConstantes

The bottom output window, titled "Output - 8.Introduccion.Conversiones (run)", shows the following execution results:

```
run:
Ingresa el número a dividir: 40
Ingresa por cuanto lo va dividir: 2
Su resultado es: 20
BUILD SUCCESSFUL (total time: 1 minute 25 seconds)
```

## Uso de **double**:



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: ");

        // La siguiente linea se cambia:
        String nombre = scanner.nextInt(); // ERROR
        // Por esta:
        String nombre = scanner.nextLine();
        System.out.println("Hola, " + nombre);
    }
}
```

El error que tenía era sobre la función **scanner**. Estaba tomando un valor de **numero entero** (`scanner.nextInt()`) cuando lo que estaba ingresando era un valor de **String**. El valor de String estaba dado por la variable **nombre** que se asigna su tipo en la misma línea (`String nombre = scanner.nextLine();`).

**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);  
    }  
}
```

Prueba de escritorio				
Linea	Variables	a	b	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

La prueba de escritorio da como valor de la variable **resultado = 2**. Esto se debe a que resultado es de **tipo entero (int)** y por lo tanto los números decimales solo pueden ser alojados en variables de tipo **double**. Al intentar guardar el valor 2,5 en **resultado** este trunca el decimal.