

PROGRAMACIÓN II Trabajo Práctico 1:

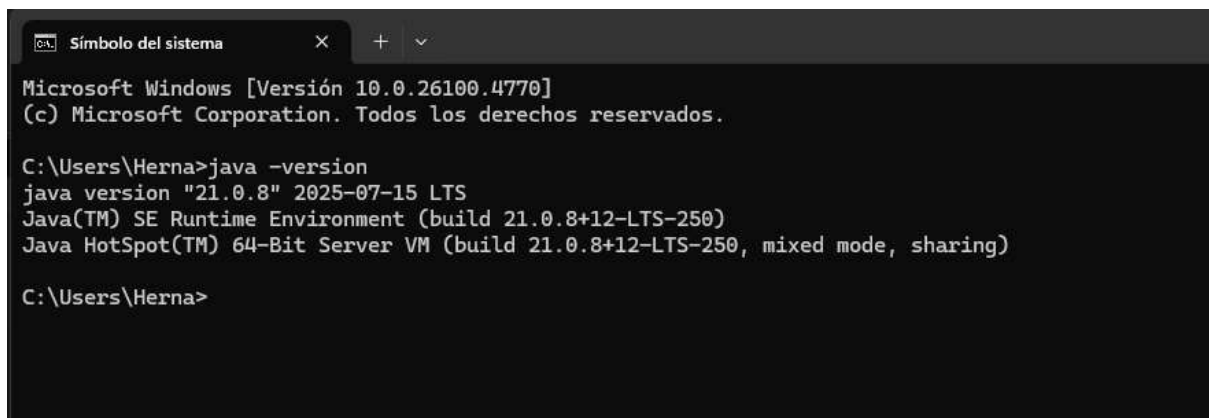
Introducción a Java

NOMBRE: COCERES, GABRIEL HERNAN

COMISION: 16

Consigna 1:

a) Confirmo que tengo Java JDK instalado.

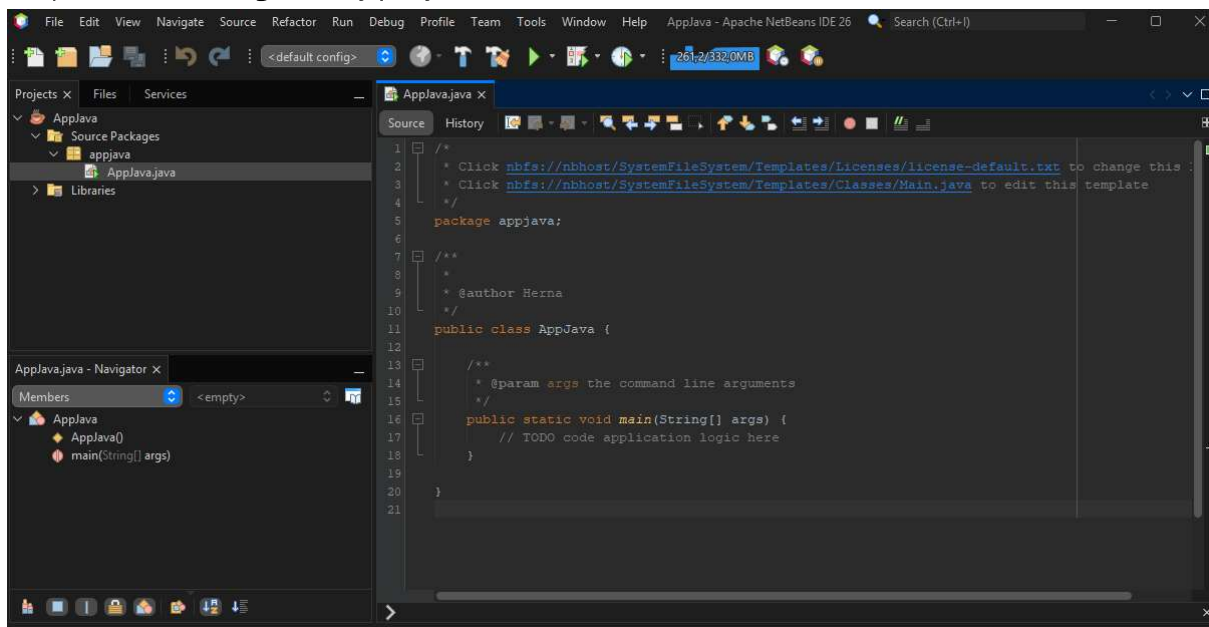


```
Símbolo del sistema
Microsoft Windows [Versión 10.0.26100.4770]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\Herna>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)

C:\Users\Herna>
```

b,c) Entorno configurado y proyecto creado:



2)

The screenshot shows an IDE window titled 'AppJava.java X'. The 'Source' tab is active, displaying the following Java code:

```
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 appjava;
6
7   /**
8    *
9    * @author Herna
10   */
11   public class AppJava {
12
13       /**
14        * @param args the command line arguments
15        */
16       public static void main(String[] args) {
17
18           String saludo = "Hola Java";
19           System.out.println(saludo);
20
21       }
22   }
23
24
```

Below the code editor, the 'Output - AppJava (run)' tab is active, showing the execution results:

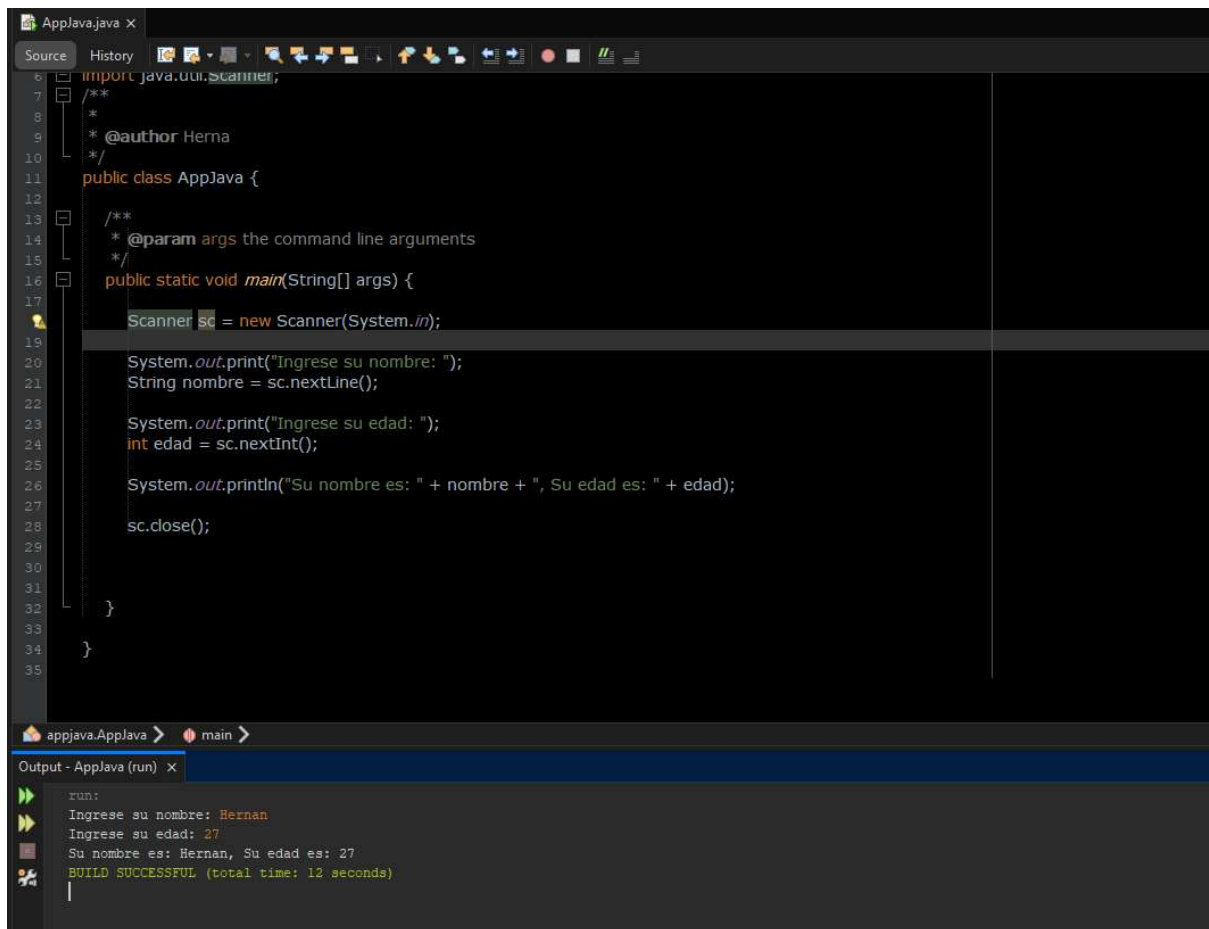
```
run:
Hola Java
BUILD SUCCESSFUL (total time: 0 seconds)
```

3)

The screenshot shows the same IDE window with 'AppJava.java X'. The 'Source' tab is active, displaying the following Java code:

```
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 appjava;
6
7   /**
8    *
9    * @author Herna
10   */
11   public class AppJava {
12
13       /**
14        * @param args the command line arguments
15        */
16       public static void main(String[] args) {
17
18           String nombre = "Hernan";
19           int edad = 27;
20           double altura = 1.79;
21           boolean estudiante = true;
22
23           System.out.println("Nombre: " + nombre + ", Edad: " + edad + ", Altura: " + altura + ", ¿Es estudiante?: " + estudiante);
24
25       }
26   }
27
28
```

4)



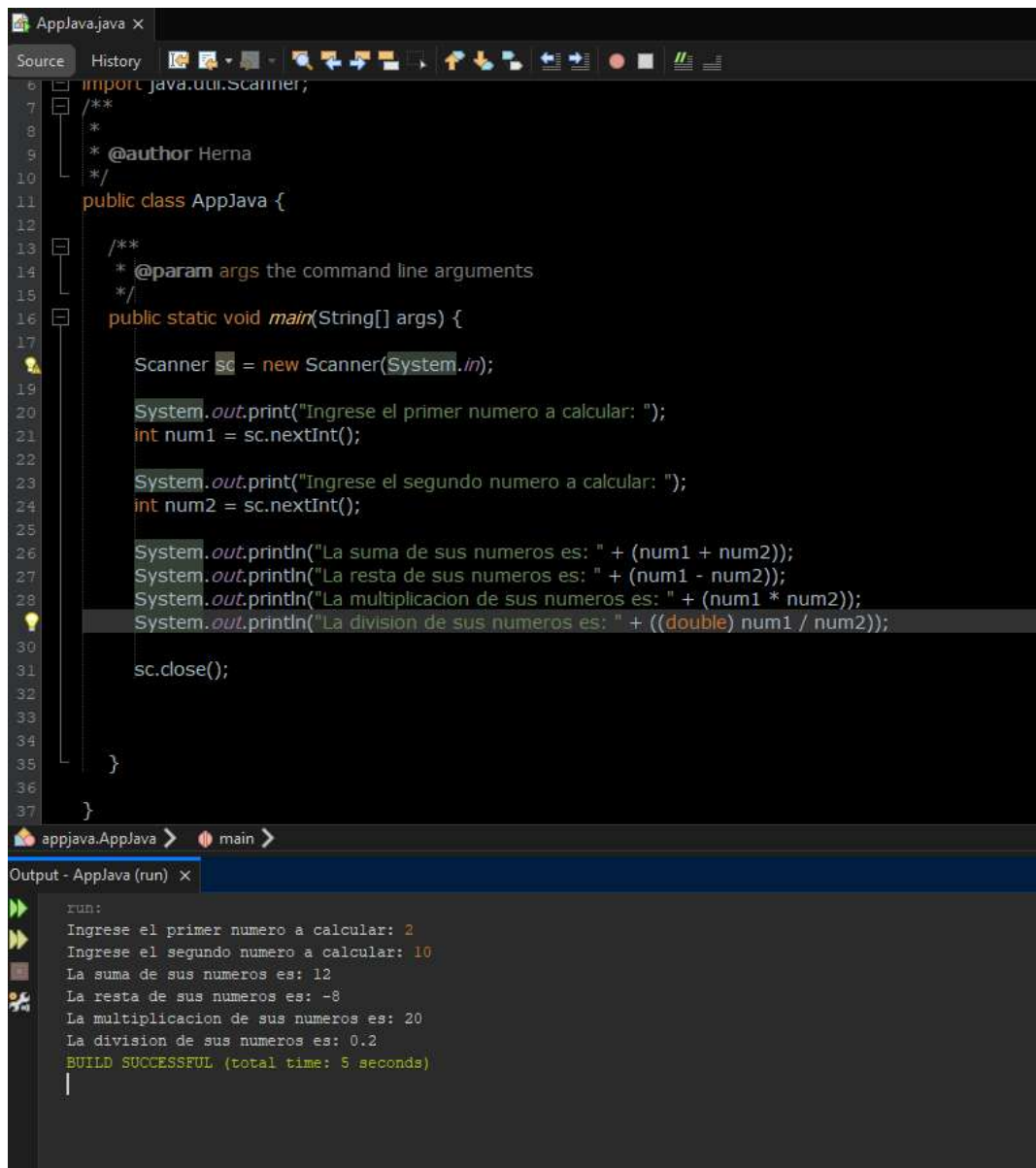
The image shows a screenshot of an IDE with a dark theme. The main editor window displays the source code for a Java file named `AppJava.java`. The code imports `java.util.Scanner` and includes a class-level comment: `/** @author Herna */`. The `AppJava` class contains a `main` method that takes a `String[] args` parameter. Inside the `main` method, a `Scanner` object named `sc` is created using `System.in`. The program then prompts the user to enter their name and age, reads the input, and prints a summary message. Finally, the `Scanner` is closed.

```
6 import java.util.Scanner;
7
8 /**
9  * @author Herna
10 */
11 public class AppJava {
12
13     /**
14      * @param args the command line arguments
15      */
16     public static void main(String[] args) {
17
18         Scanner sc = new Scanner(System.in);
19
20         System.out.print("Ingrese su nombre: ");
21         String nombre = sc.nextLine();
22
23         System.out.print("Ingrese su edad: ");
24         int edad = sc.nextInt();
25
26         System.out.println("Su nombre es: " + nombre + ", Su edad es: " + edad);
27
28         sc.close();
29
30     }
31 }
32
33
34
35
```

Below the editor, the IDE shows the execution of the program. The output window displays the following text:

```
run:
Ingrese su nombre: Hernan
Ingrese su edad: 27
Su nombre es: Hernan, Su edad es: 27
BUILD SUCCESSFUL (total time: 12 seconds)
```

5)



The image shows a screenshot of an IDE with a dark theme. The top pane displays the source code for a Java application named `AppJava.java`. The code imports `java.util.Scanner` and defines a `public class AppJava` with a `main` method. The `main` method uses a `Scanner` to read two integers from the user, performs addition, subtraction, multiplication, and division, and prints the results. The bottom pane shows the output of the program, which matches the user input and calculations shown in the code.

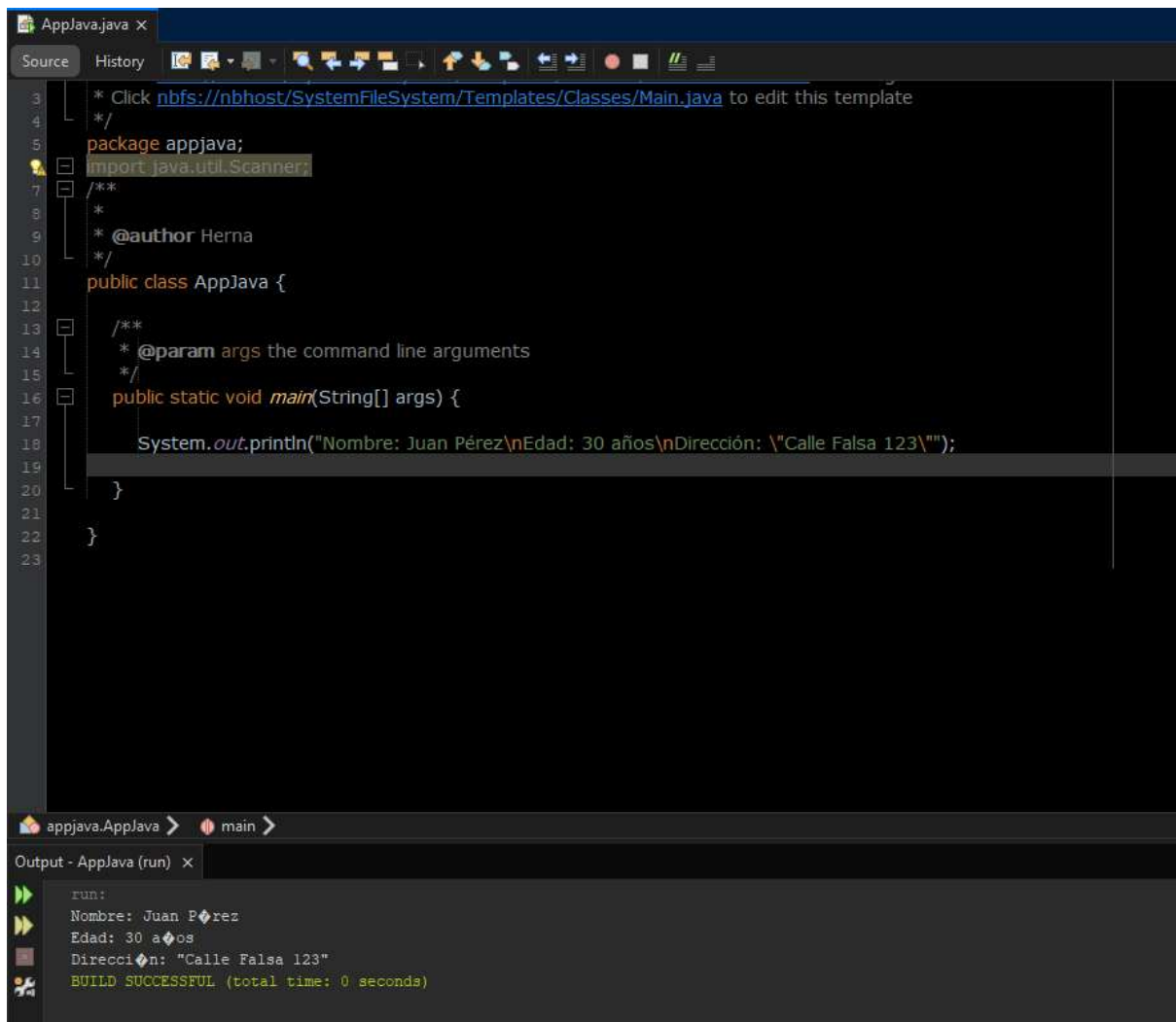
```
6 import java.util.Scanner;
7 /**
8  *
9  * @author Herna
10 */
11 public class AppJava {
12
13     /**
14      * @param args the command line arguments
15      */
16     public static void main(String[] args) {
17
18         Scanner sc = new Scanner(System.in);
19
20         System.out.print("Ingrese el primer numero a calcular: ");
21         int num1 = sc.nextInt();
22
23         System.out.print("Ingrese el segundo numero a calcular: ");
24         int num2 = sc.nextInt();
25
26         System.out.println("La suma de sus numeros es: " + (num1 + num2));
27         System.out.println("La resta de sus numeros es: " + (num1 - num2));
28         System.out.println("La multiplicacion de sus numeros es: " + (num1 * num2));
29         System.out.println("La division de sus numeros es: " + ((double) num1 / num2));
30
31         sc.close();
32
33
34     }
35 }
36
37 }
```

appjava.AppJava > main >

Output - AppJava (run) x

```
run:
Ingrese el primer numero a calcular: 2
Ingrese el segundo numero a calcular: 10
La suma de sus numeros es: 12
La resta de sus numeros es: -8
La multiplicacion de sus numeros es: 20
La division de sus numeros es: 0.2
BUILD SUCCESSFUL (total time: 5 seconds)
```

6)



```
AppJava.java x
Source History
3  /* Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Main.java to edit this template
4  */
5  package appjava;
6  import java.util.Scanner;
7  /**
8   *
9   * @author Herna
10  */
11  public class AppJava {
12
13      /**
14       * @param args the command line arguments
15       */
16      public static void main(String[] args) {
17
18          System.out.println("Nombre: Juan Pérez\nEdad: 30 años\nDirección: \"Calle Falsa 123\"");
19
20      }
21  }
22
23
appjava.AppJava > main >
Output - AppJava (run) x
run:
Nombre: Juan Pérez
Edad: 30 años
Dirección: "Calle Falsa 123"
BUILD SUCCESSFUL (total time: 0 seconds)
```

7)

Apartir del siguiente código determino que:

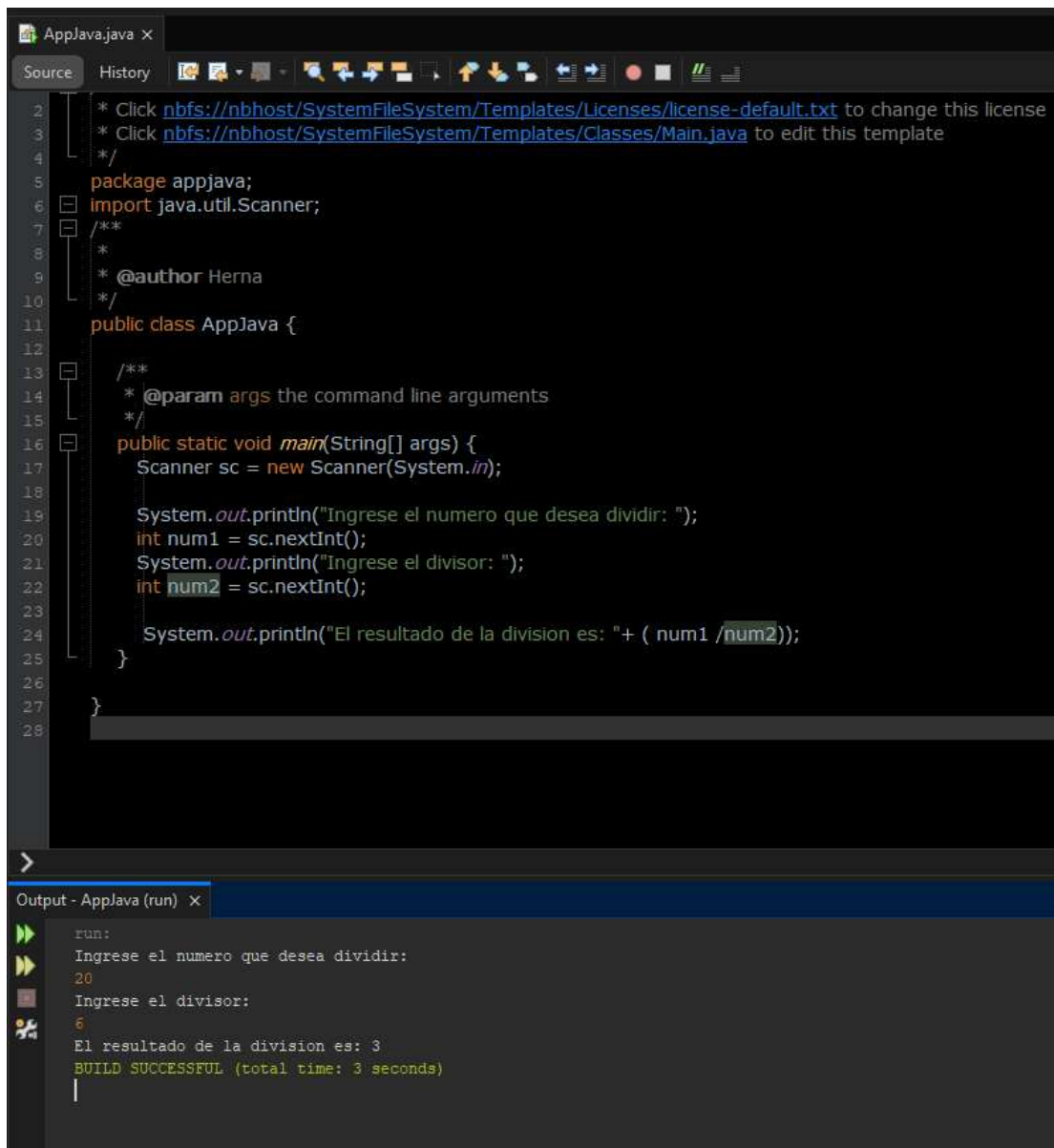
int x = 10; // Línea 1 = EXPRESION

x = x + 5; // Línea 2 = INSTRUCCION

System.out.println(x); // Línea 3 = EXPRESION

La diferencia entre una expresión y una instrucción en programación puede decirse que principalmente en una expresión el código busca mostrar o declarar un valor o varios en una misma expresión, y en una instrucción el código busca realizar una acción con los datos expresados.

8) a)



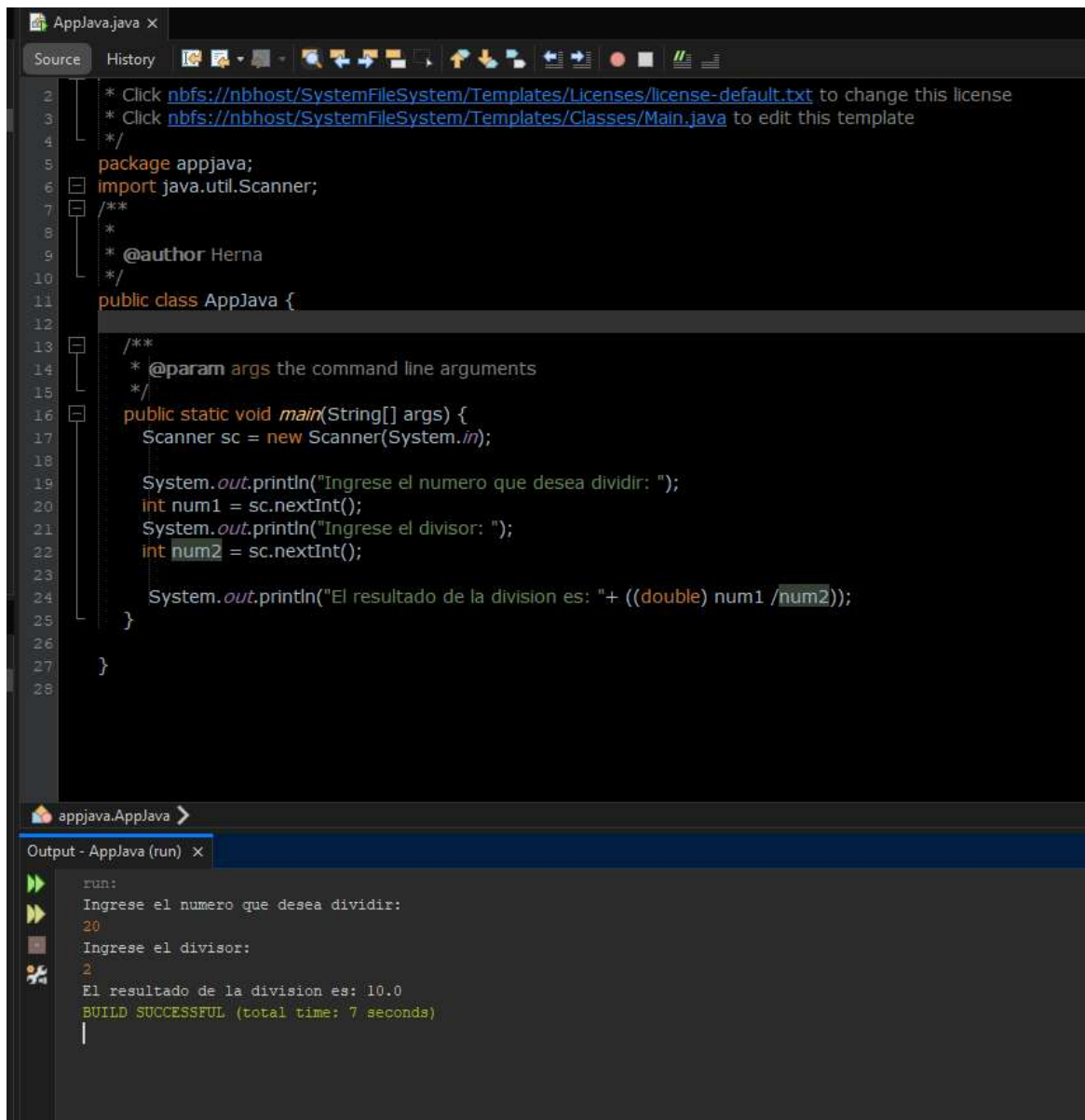
The image shows a screenshot of an IDE with a dark theme. The top pane displays the source code for a Java file named `AppJava.java`. The code includes package declarations, imports, and a `main` method that uses a `Scanner` to take user input for two numbers and prints the result of their division. The bottom pane shows the output of running the program, where the user has entered 20 and 6, resulting in a division of 3. The output also indicates a successful build.

```
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 appjava;
6  import java.util.Scanner;
7  /**
8   *
9   * @author Herna
10  */
11  public class AppJava {
12
13      /**
14       * @param args the command line arguments
15       */
16      public static void main(String[] args) {
17          Scanner sc = new Scanner(System.in);
18
19          System.out.println("Ingrese el numero que desea dividir: ");
20          int num1 = sc.nextInt();
21          System.out.println("Ingrese el divisor: ");
22          int num2 = sc.nextInt();
23
24          System.out.println("El resultado de la division es: " + ( num1 / num2));
25      }
26  }
27
28
```

Output - AppJava (run) ×

```
run:
Ingrese el numero que desea dividir:
20
Ingrese el divisor:
6
El resultado de la division es: 3
BUILD SUCCESSFUL (total time: 3 seconds)
|
```

b)



```
AppJava.java x
Source History
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 appjava;
6  import java.util.Scanner;
7  /**
8   *
9   * @author Herna
10  */
11  public class AppJava {
12
13      /**
14       * @param args the command line arguments
15       */
16      public static void main(String[] args) {
17          Scanner sc = new Scanner(System.in);
18
19          System.out.println("Ingrese el numero que desea dividir: ");
20          int num1 = sc.nextInt();
21          System.out.println("Ingrese el divisor: ");
22          int num2 = sc.nextInt();
23
24          System.out.println("El resultado de la division es: "+ ((double) num1 / num2));
25      }
26  }
27
28
appjava.AppJava >
Output - AppJava (run) x
run:
Ingrese el numero que desea dividir:
20
Ingrese el divisor:
2
El resultado de la division es: 10.0
BUILD SUCCESSFUL (total time: 7 seconds)
```

Comparando los resultados al manejar la división con “double” la división se hace correctamente en caso de que el resultado sea un entero y que contenga decimales.

Por otra parte, sin usar “double” la división en caso de que el resultado sea con decimales, solo devuelve la parte entera del resultado.

9)

El siguiente código presenta un error:

Scanner scanner = new Scanner(System.in);

System.out.print("Ingresa tu nombre: ");

String nombre = scanner.nextInt(); //ERROR

```
System.out.println("Hola, " + nombre);
```

El error esta en el método que utiliza el objeto scanner, porque la variable esta declarada como un string pero utiliza el método para leer un integer(entero), el método correcto para poder leer una línea de texto es “scanner.nextLine()” o en el caso de que solo se quiera leer una palabra hasta el primer espacio se puede usar solamente “scanner.next()”.

De la siguiente manera el código funcionaria como se espera:

```
Scanner scanner = new Scanner(System.in);

System.out.print("Ingresa tu nombre: ");

String nombre = scanner.nextLine();//ERROR

System.out.println("Hola, " + nombre);
```

10)

Tabla de prueba de escritorio:

Línea	a	b	resultado
1	—	—	—
2	5	—	—
3	5	2	—
4	5	2	2

EL valor de resultado es 2, porque cuando intentamos dividir dos números enteros en java, solo nos devuelve el resultado con su parte entera y descarta el resto, para poder hacer una división más exacta podemos usar “double” en uno de los dos valores para que el resultado pueda ser decimal y poder hacer la división correctamente. Con que uno de los dos valores a dividir sea decimal, alcanza para que la división de un resultado con decimal.