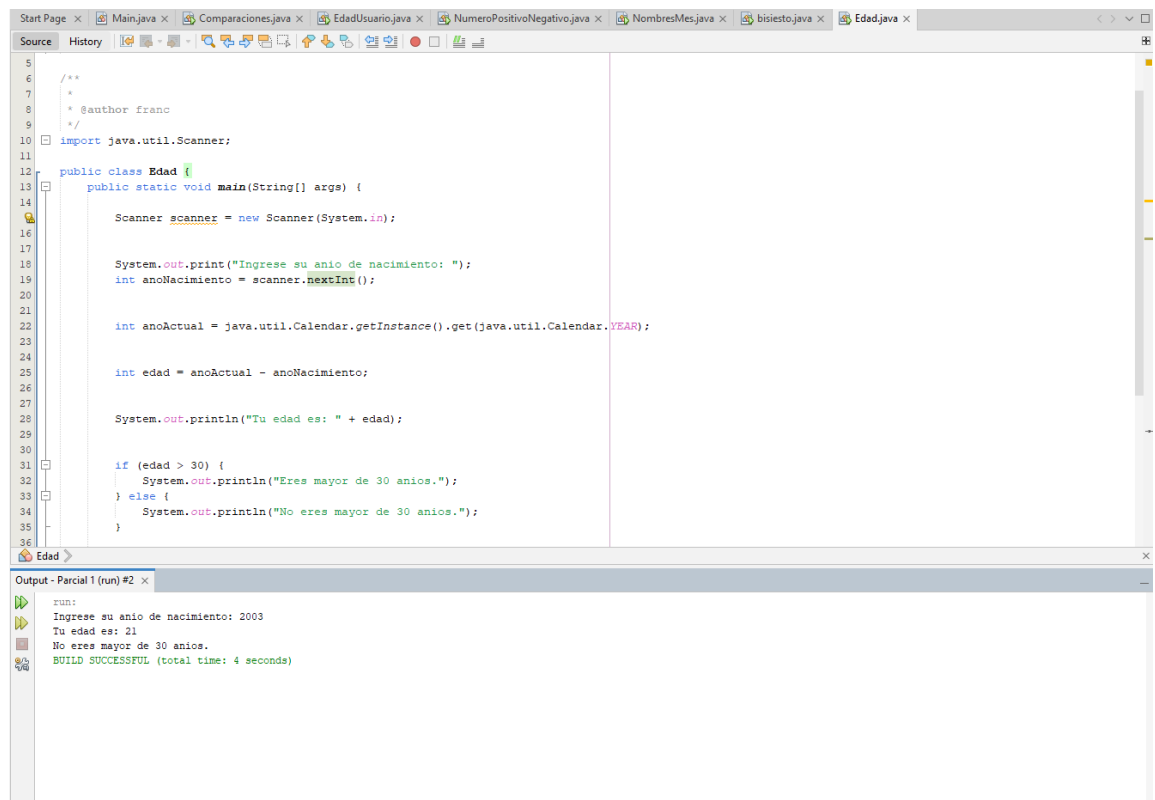


PARCIAL 1 PROGRAMACION II

Hugo Ismael Taque Franco

6590-23-20358

1.



```
5
6  /**
7   *
8   * @author franco
9   */
10 import java.util.Scanner;
11
12 public class Edad {
13     public static void main(String[] args) {
14
15         Scanner scanner = new Scanner(System.in);
16
17         System.out.print("Ingrese su año de nacimiento: ");
18         int anoNacimiento = scanner.nextInt();
19
20
21         int anoActual = java.util.Calendar.getInstance().get(java.util.Calendar.YEAR);
22
23
24         int edad = anoActual - anoNacimiento;
25
26
27         System.out.println("Tu edad es: " + edad);
28
29
30         if (edad > 30) {
31             System.out.println("Eres mayor de 30 años.");
32         } else {
33             System.out.println("No eres mayor de 30 años.");
34         }
35     }
36 }
```

Output - Parcial 1 (run) #2 x

```
run:
Ingrese su año de nacimiento: 2003
Tu edad es: 21
No eres mayor de 30 años.
BUILD SUCCESSFUL (total time: 4 seconds)
```

Explicación del código:

Importación de Scanner: Usamos la clase Scanner para leer la entrada del usuario.

Obtención del año actual: Utilizamos la clase Calendar para obtener el año actual.

Cálculo de la edad: Restamos el año de nacimiento del año actual.

Condición para verificar la edad: Usamos una estructura if para comprobar si la edad es mayor de 30 años y mostrar el mensaje adecuado.

2.

The screenshot shows an IDE with several open files. The active file is `EdadUsuario.java`, which contains the following code:

```
37  
38  
39     System.out.print("Ingrese la cantidad a depositar: ");  
40     double cantidadDeposito = scanner.nextDouble();  
41     cuenta.depositar(cantidadDeposito);  
42     break;  
43  
44     case 2:  
45         do ... while (opcion != 4) > switch (opcion) > case 2: >
```

The Output window shows the execution of the program:

```
run:  
Ingrese el nombre del titular de la cuenta: Hugo Ismael Taque Franco  
Ingrese el saldo inicial de la cuenta: 5000  
  
Menu:  
1. Depositar  
2. Retirar  
3. Consultar saldo  
4. Salir  
Seleccione una opcion: 1  
Ingrese la cantidad a depositar: 3000  
Deposito realizado con exito. Saldo actual: 8000.0  
  
Menu:  
1. Depositar  
2. Retirar  
3. Consultar saldo  
4. Salir  
Seleccione una opcion: 2  
Ingrese la cantidad a retirar: 2500  
Retiro realizado con exito. Saldo actual: 5500.0  
  
Menu:  
1. Depositar  
2. Retirar  
3. Consultar saldo  
4. Salir  
Seleccione una opcion: 3  
Saldo actual: 5500.0  
  
Menu:  
1. Depositar  
2. Retirar  
3. Consultar saldo  
4. Salir  
Seleccione una opcion: 4  
Gracias por usar el sistema. Adios!  
BUILD SUCCESSFUL (total time: 37 seconds)
```

Explicación del código:

Atributos:

titular: Almacena el nombre del titular de la cuenta.

saldo: Guarda el saldo actual de la cuenta.

Constructor:

Inicializa el objeto CuentaBanco con el nombre del titular y un saldo inicial.

Método depositar(double cantidad):

Permite agregar dinero al saldo de la cuenta. Verifica que la cantidad sea positiva antes de añadirla al saldo.

Método retirar(double cantidad):

Permite retirar dinero del saldo de la cuenta. Verifica que la cantidad sea positiva y que haya suficiente saldo disponible antes de realizar el retiro.

CuentaBancaria encapsula la lógica relacionada con la cuenta bancaria.

Main proporciona una interfaz de usuario simple mediante un menú y permite interactuar con una instancia de CuentaBancaria.