FICHA DE TRABAJO PARA LOS EJERCICIOS 2 Y 3 DE LA TAREA

Apellidos y nombre:

José Bueno Cruz

Instrucciones:

En la primera captura debes incluir también, como fondo, tu *login* **en la plataforma**, donde se pueda observar tu nombre y tu foto, para así comprobar que el trabajo lo has realizado tú.

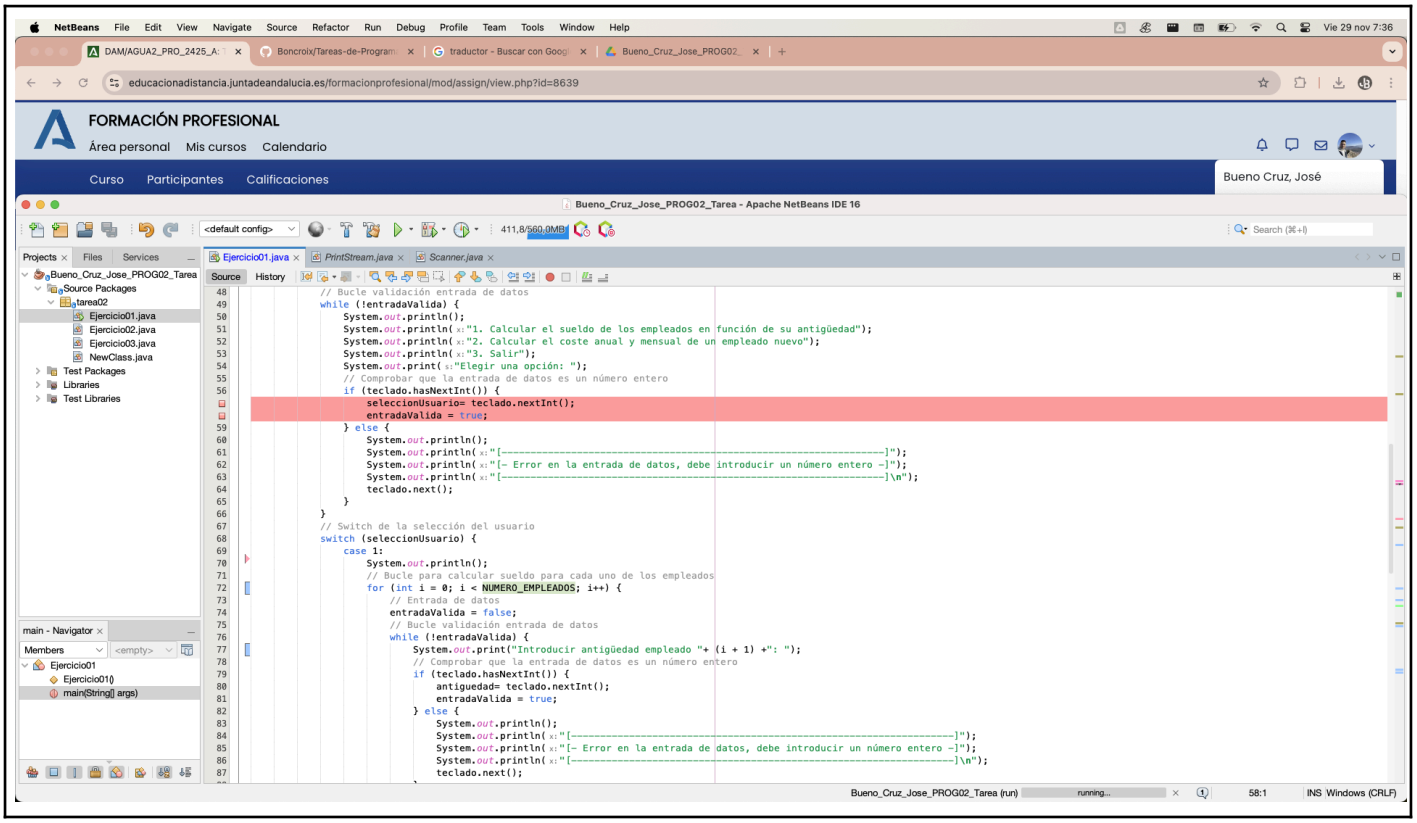
Una vez hayas terminado de cumplimentar el documento, genera un PDF a partir de él para evitar problemas de formato a la hora de corregirlo. ¡Y no olvides incluir ese documento en el paquete junto con el resto de tu proyecto! (En el zip generado exportando tu proyecto de NetBeans y que has nombrado incluyendo tu nombre y apellidos, añade el pdf que has obtenido rellenando el modelo de ficha, como en el documento de ejemplo).

Recomendaciones:

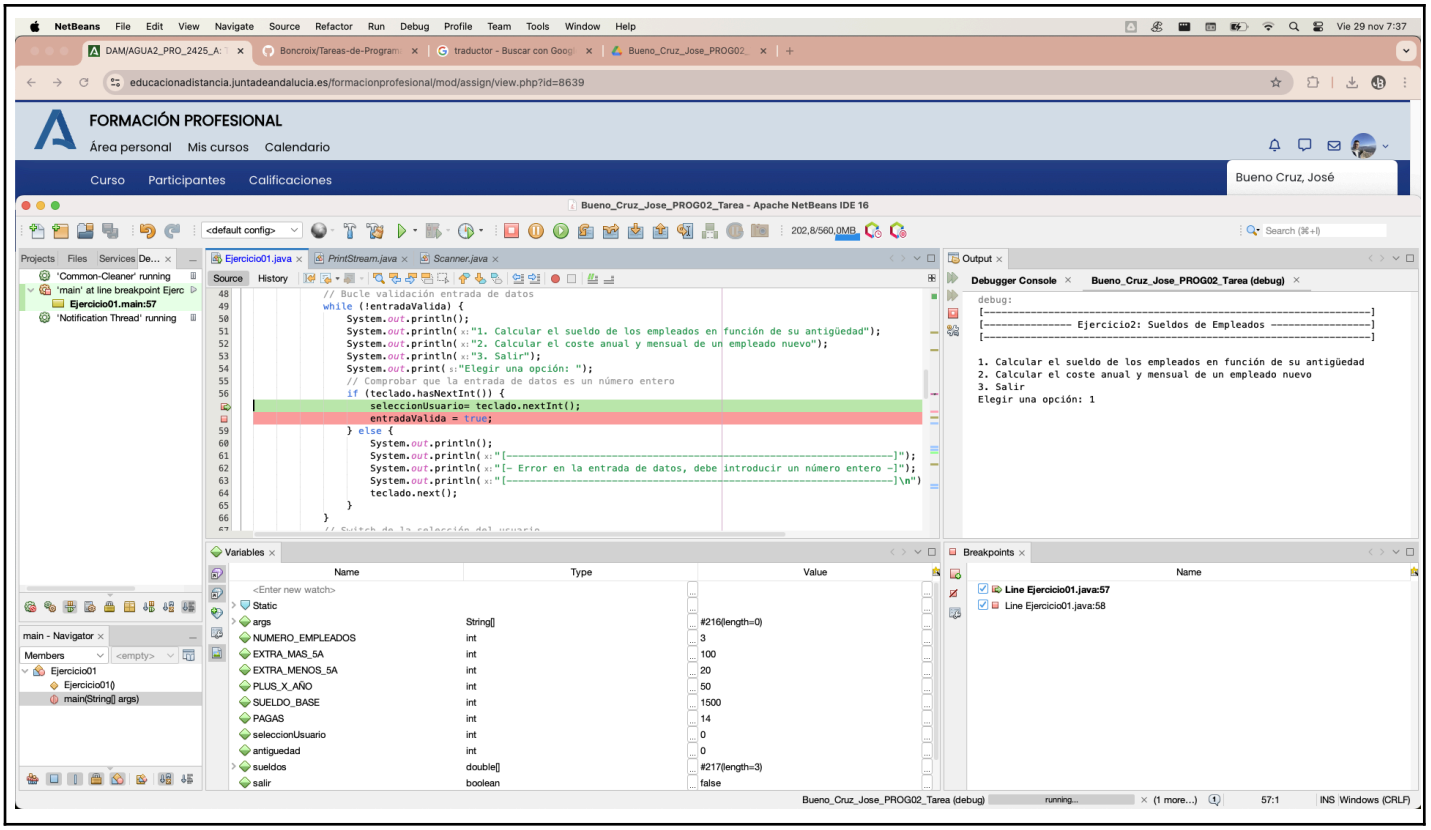
- Para realizar las capturas de pantalla dispones de gran cantidad de herramientas útiles, empezando por la propia herramienta "Recortes" que viene integrada en Windows, aunque una buena opción para Windows, más sofisticada que la herramienta "Recortes", es la aplicación GreenShot. Puedes descargarla desde su sitio oficial: https://getgreenshot.org/downloads/.
- Para quien utilice Linux, puede buscar algunas alternativas en https://alternativeto.net/software/greenshot/?platform=linux.
- Y lo mismo para los usuarios de Mac: https://alternativeto.net/software/greenshot/?platform=mac. Para los usuarios de MacOS pueden utilizar para realizar las capturas las siguientes combinaciones de teclas (puedes encontrar más información aquí):
  - Para relizar una captura de toda la pantalla en Mac OSX pulsa simultáneamente las teclas: Shift + Comand + 3
  - Para relizar una captura de parte de la pantalla en Mac OSX pulsa simultáneamente las teclas: Shift + Comand + 4
  - Para realizar la captura de una ventana o menú en Mac OSX pulsa simultáneamente las teclas: Shift + Comand + 4 + espacio

EJERCICIO 2

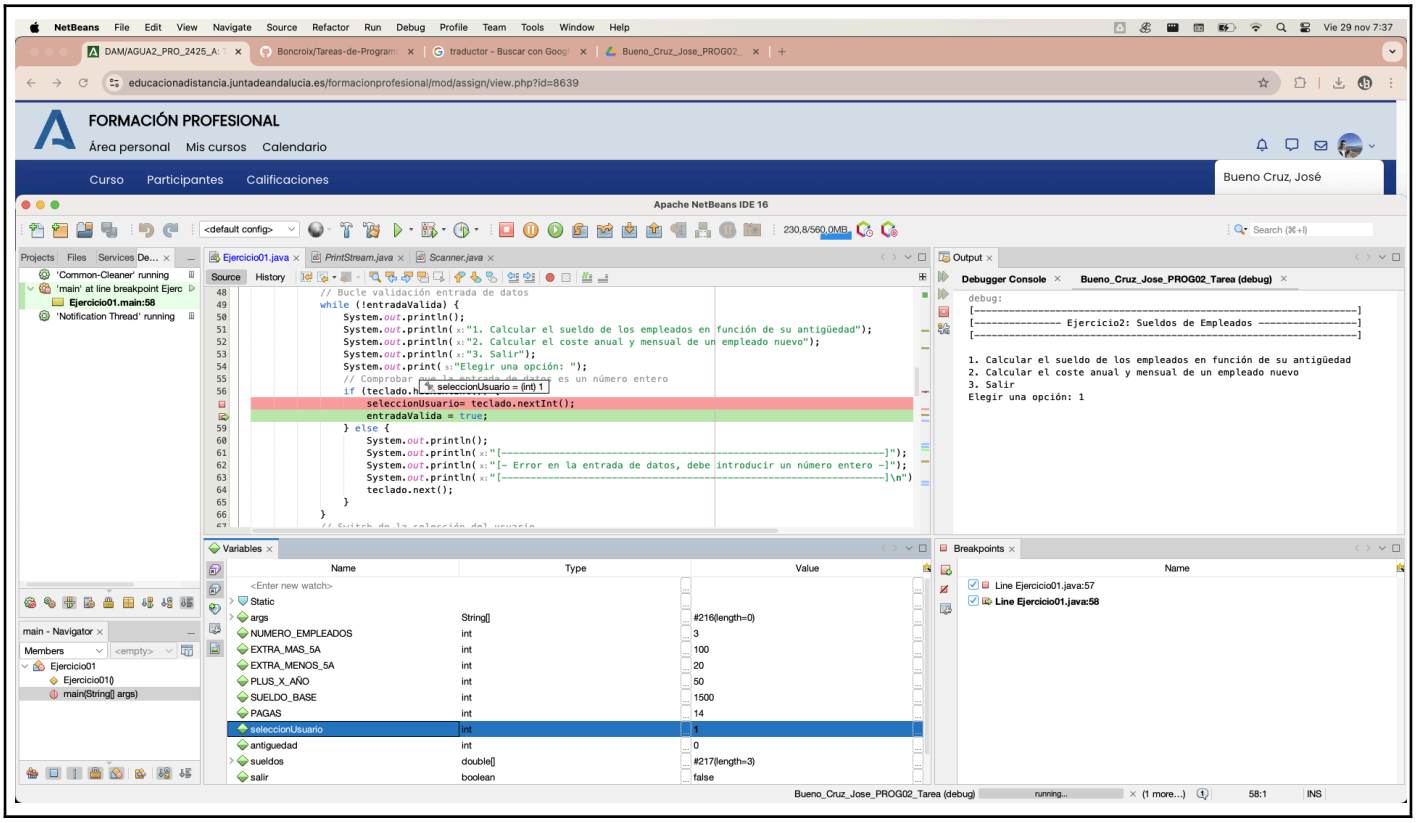
1.- Establecer un punto de ruptura en la línea donde se lea la opción de menú elegida por el usuario. Muestra una captura de pantalla del breakpoint junto con tu perfil en la plataforma.



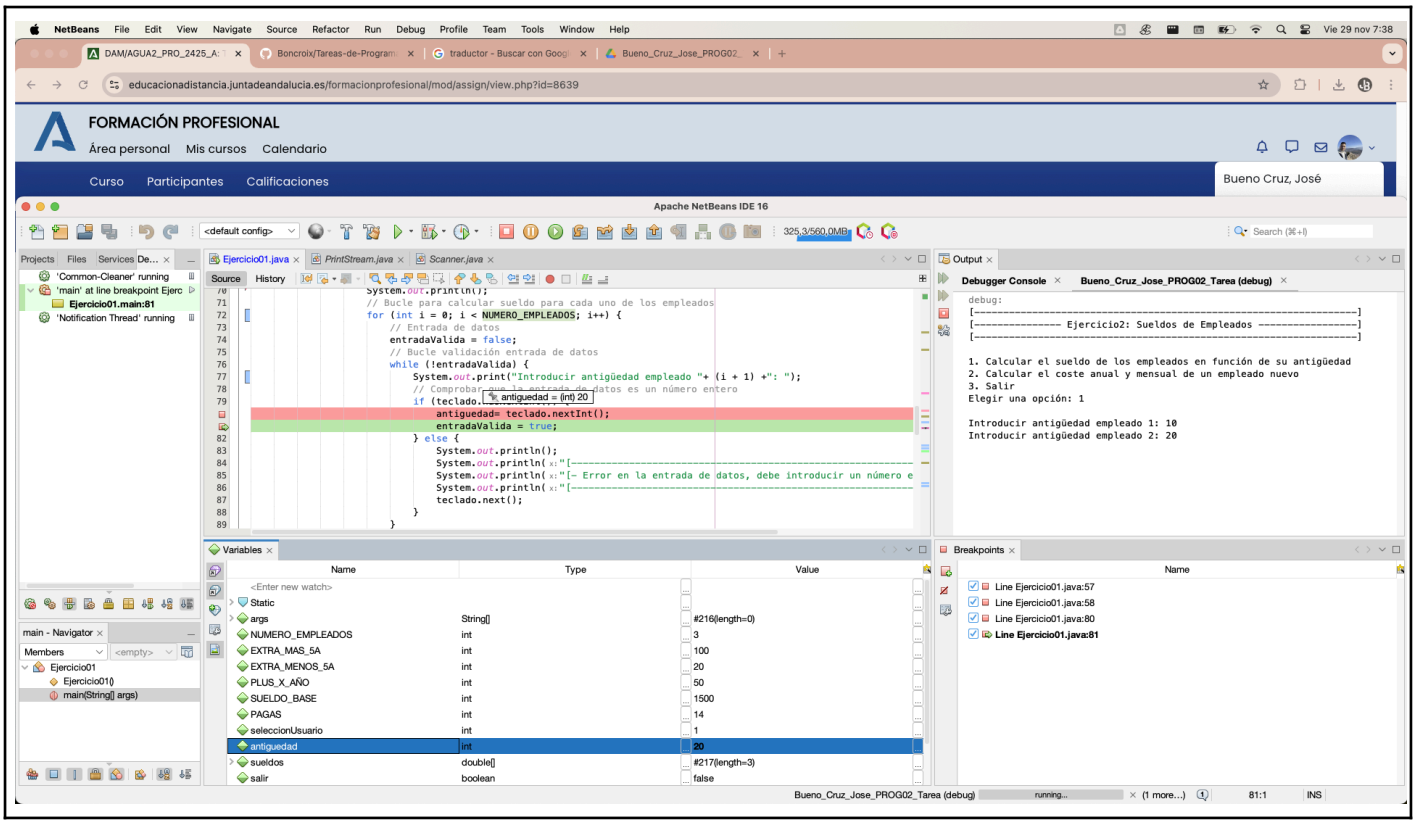
2.- Ejecuta en el programa modo depuración. Muestra la ventana de variables y sitúala a la izquierda debajo de los proyectos con el programa detenido en el breakpoint creado anteriormente.



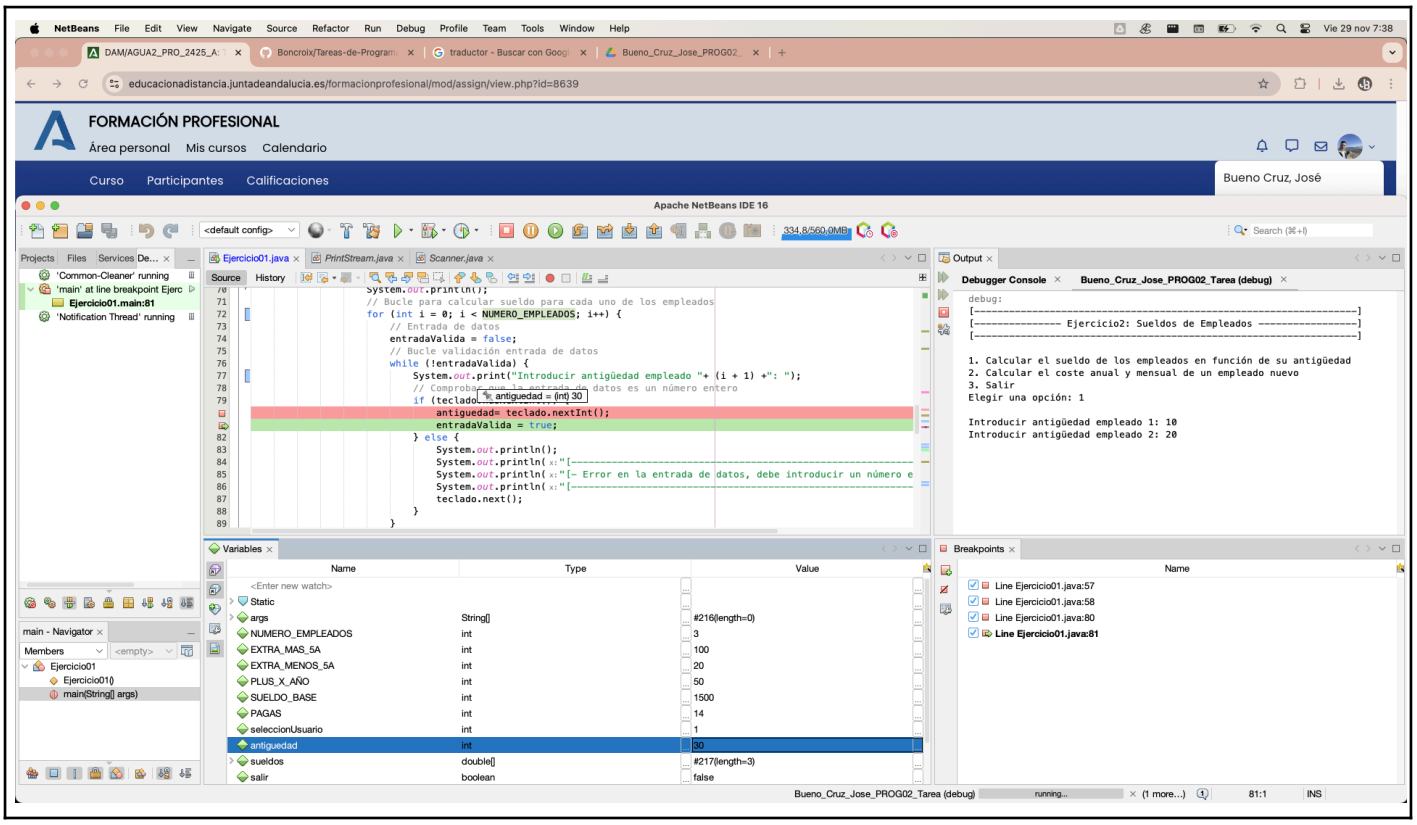
3.- Muestra el valor de la variable que has utilizado para recoger la opción de menú elegida.



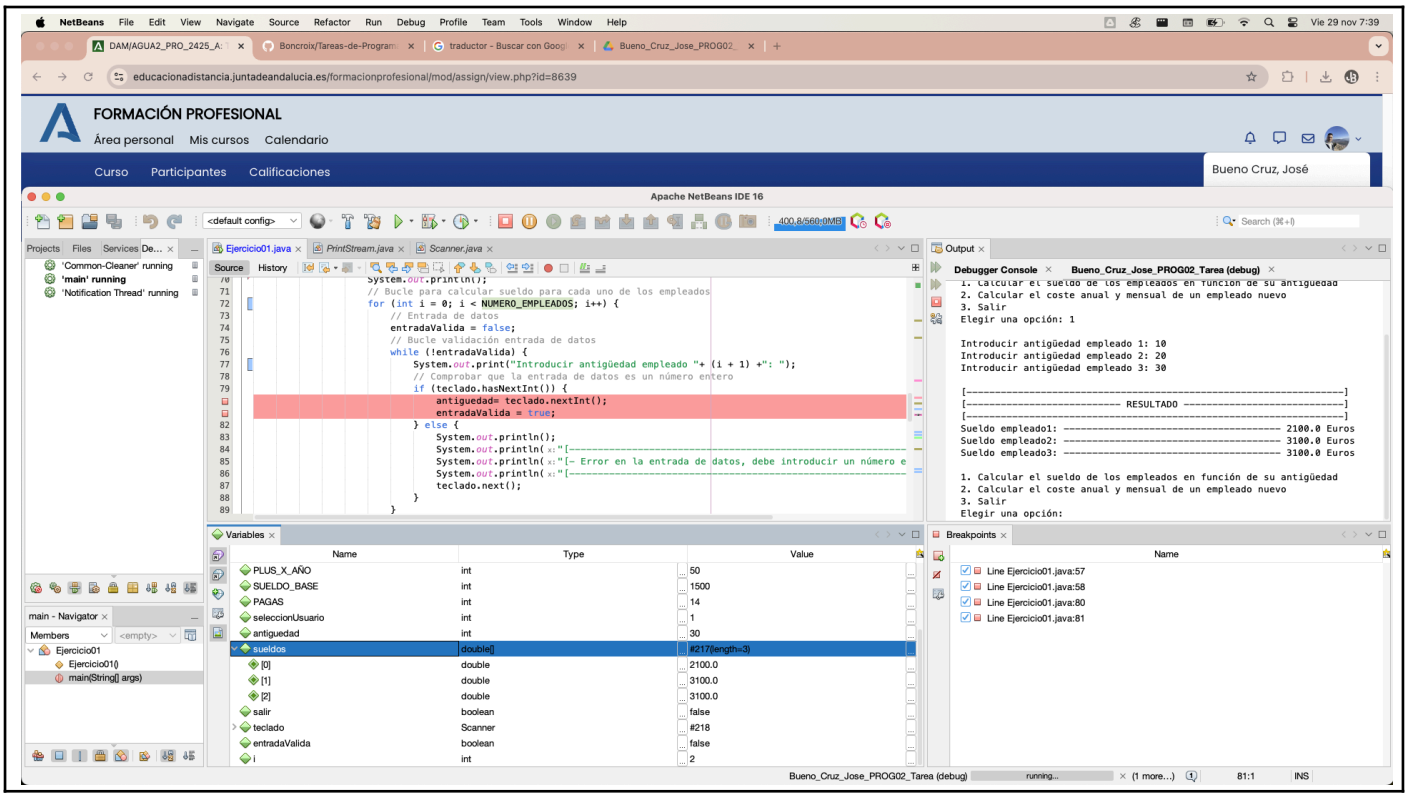
4.- Avanza paso a paso por la ejecución del programa hasta que éste solicite la antigüedad del "Empleado 2", introduce como valor para este dato 20. Muestra el valor de la variable colocando el cursor sobre la variable mediante la captura correspondiente.



5.- Utiliza el inspector de variables para cambiar el valor del dato de la antigüedad de 20 a 30. Muestra la captura correspondiente.



6.- Finaliza la ejecución del programa mostrando el resultado final con la captura de pantalla correspondiente.



EJERCICIO 3

1. Dados los siguientes bloques de código, indica cuál de ellos corresponde a una excepción y cuál corresponde a una aserción, explicando el por qué de tu respuesta.

Bloque de código 1:

```java
int numerador = 10;
int denominador = 0;

try {
    int resultado = numerador / denominador;
    System.out.println("El resultado es: " + resultado);
} catch (ArithmeticException e) {
    System.out.println("Error: No se puede dividir por cero.");
}
```

Bloque de código 2:

```java
int edad = -5;

assert edad > 0 : "La edad debe ser mayor que 0";

System.out.println("La edad es: " + edad);
```

Respuesta:

Bloque de código 1: Este bloque corresponde a una excepción porque se utiliza **try{}** y **catch{}** para manejar posibles errores.
Bloque de código 2: Corresponde a una aserción porque se utiliza la palabra clave **assert** para verificar condiciones durante la ejecución del programa.

2. ¿Para qué se usan las excepción y aserciones en java?. Justifica tu respuesta de acuerdo a lo explicado en las videoconferencias.

Aserción: Las aserciones en Java son una forma de verificar que ciertas condiciones se mantengan verdaderas durante la ejecución de un programa. Se utilizan para detectar errores lógicos o inconsistencias durante el desarrollo, en tiempo de depuración. **Excepción:** Las excepciones son eventos que indican que algo ha salido mal durante la ejecución del programa (por ejemplo, errores en el usuario, problemas de E/S, errores de red, división por cero, etc.). Se utilizan para manejar condiciones excepcionales que pueden ocurrir durante la ejecución y permiten que el programa recupere el control o se maneje de manera adecuada. Las excepciones están orientadas a un manejo robusto de situaciones inesperadas o erróneas.