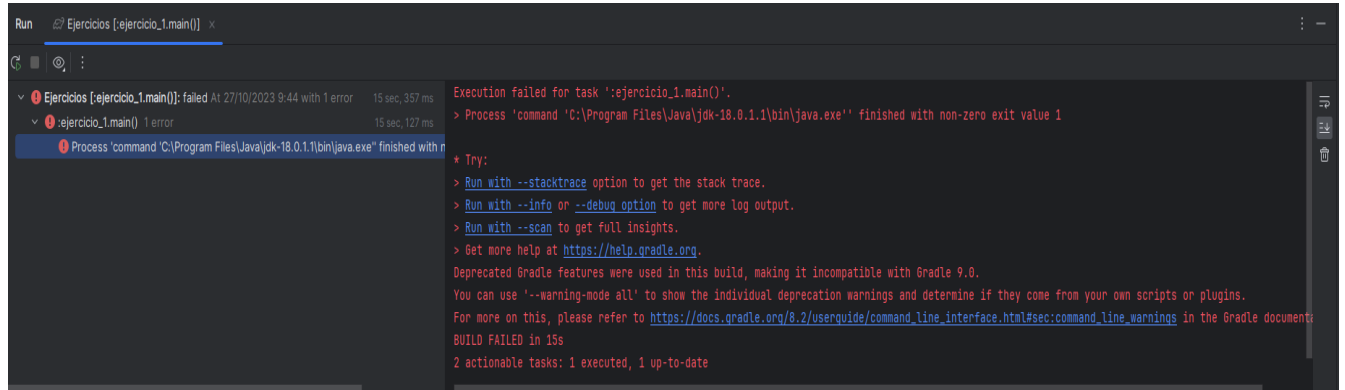
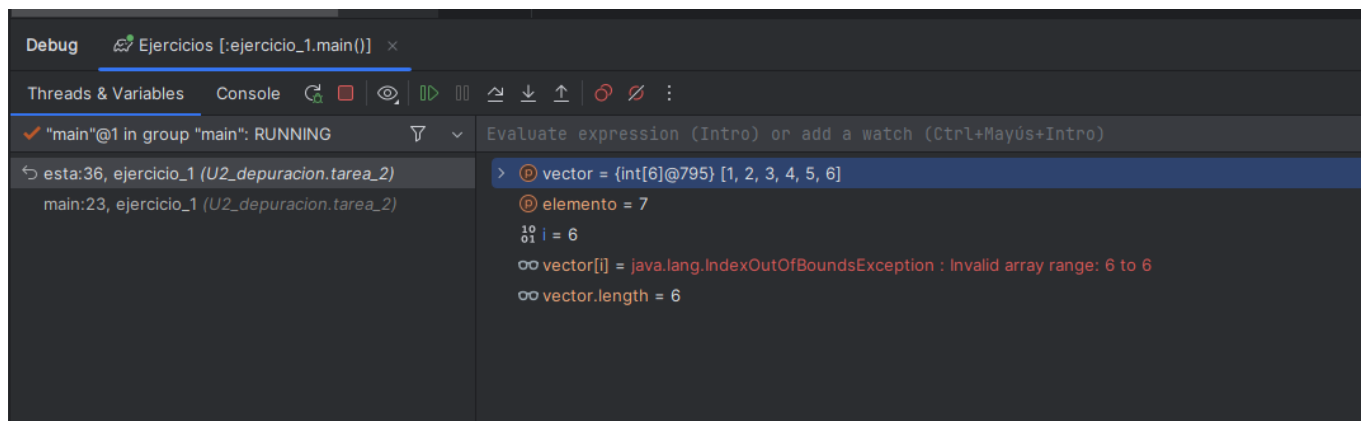


1. Ejecutar el programa , provocar el fallo del mismo y capturar un pantallazo del error que aparece. ¿Cuál es la hipótesis que según tu opinión ha provocado el fallo?



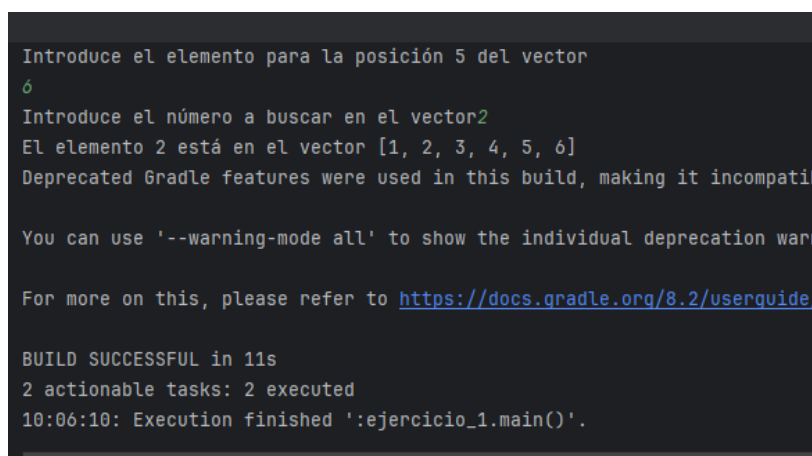
```
Run Ejercicios [:ejercicio_1.main()] x
Execution failed for task ':ejercicio_1.main()'.
> Process 'command 'C:\Program Files\Java\jdk-18.0.1\bin\java.exe'' finished with non-zero exit value 1
* Try:
> Run with --stacktrace option to get the stack trace.
> Run with --info or --debug option to get more log output.
> Run with --scan to get full insights.
> Get more help at https://help.gradle.org.
Deprecated Gradle features were used in this build, making it incompatible with Gradle 9.0.
You can use '--warning-mode all' to show the individual deprecation warnings and determine if they come from your own scripts or plugins.
For more on this, please refer to https://docs.gradle.org/8.2/userguide/command_line_interface.html#sec:command_line_warnings in the Gradle documenta
BUILD FAILED in 15s
2 actionable tasks: 1 executed, 1 up-to-date
```



```
Debug Ejercicios [:ejercicio_1.main()] x
Threads & Variables Console
"main"@1 in group "main": RUNNING
esta:36, ejercicio_1 (U2_depuracion.tarea_2)
main:23, ejercicio_1 (U2_depuracion.tarea_2)
> @ vector = [int[6]@795] [1, 2, 3, 4, 5, 6]
@ elemento = 7
@ i = 6
@ vector[i] = java.lang.IndexOutOfBoundsException: Invalid array range: 6 to 6
@ vector.length = 6
```

La ejecución da error porque un rango 6 a 6 no puede existir, ya que es el mismo número.

2. Una vez corregido el error añade el código corregido al fichero y captura pantallazos de la consola donde se pueda ver que el programa funciona correctamente tanto para un caso en el que esté el elemento como para un caso en el que no esté el elemento.



```
Introduce el elemento para la posición 5 del vector
6
Introduce el número a buscar en el vector2
El elemento 2 está en el vector [1, 2, 3, 4, 5, 6]
Deprecated Gradle features were used in this build, making it incompatible with Gradle 9.0.
You can use '--warning-mode all' to show the individual deprecation warnings and determine if they come from your own scripts or plugins.
For more on this, please refer to https://docs.gradle.org/8.2/userguide/command_line_interface.html#sec:command_line_warnings in the Gradle documenta
BUILD SUCCESSFUL in 11s
2 actionable tasks: 2 executed
10:06:10: Execution finished ':ejercicio_1.main()'.
```

```

Introduce el elemento para la posición 5 del vector
6
Introduce el número a buscar en el vector7
El elemento 7 no está en el vector[1, 2, 3, 4, 5, 6]
Deprecated Gradle features were used in this build, making it incompatible with Gradle 9.0.

You can use '--warning-mode all' to show the individual deprecation warnings and determine if

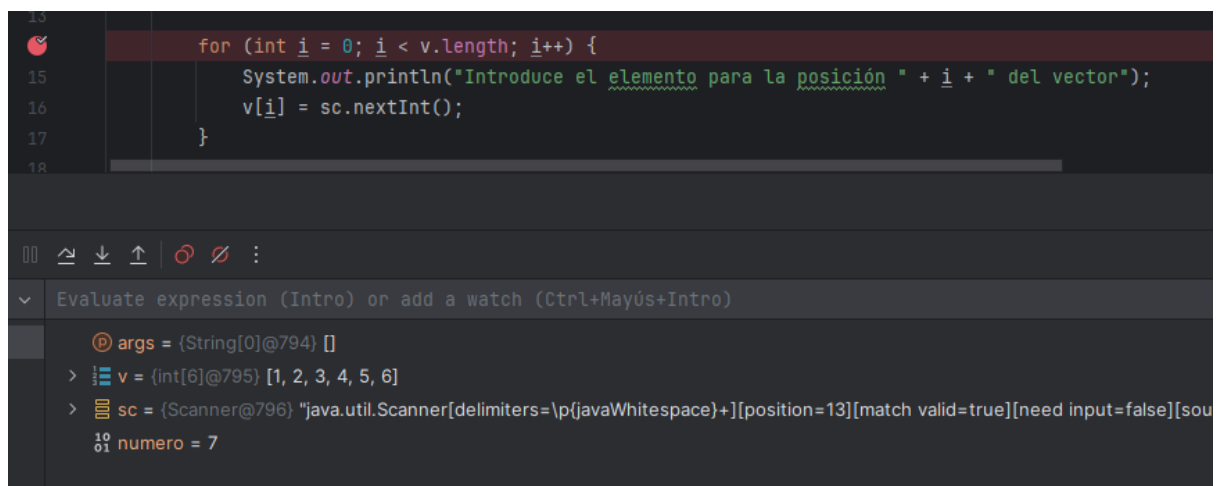
For more on this, please refer to https://docs.gradle.org/8.2/userguide/command\_line\_interface

BUILD SUCCESSFUL in 12s
2 actionable tasks: 1 executed, 1 up-to-date
10:08:49: Execution finished ':ejercicio_1.main()'.

```

3. Establecer un breakpoint para comprobar que v se rellena correctamente. Justifica por qué has elegido establecer el breakpoint en esa línea de código e incluye una captura de pantalla donde se pueda comprobar que se ha parado la ejecución y que muestre en el área de depuración "Variables" todos los elementos del vector.

He colocado el breakpoint en el 'for' que te pide introducir los 6 valores que va a tener el rango. Cuando se introducen los 6 valores el 'for' termina y la ejecución se pausa.



4. Establecer un breakpoint en el programa principal que nos permita comprobar que la función está devolviendo un valor true (deberás

introducir los valores necesarios para así sea). Justificar por qué has elegido establecer el breakpoint en esa línea de código. Incluye un pantallazo donde se pueda comprobar que se efectivamente se ha devuelto el resultado esperado (true).

He introducido un breakpoint ahí para comprobar que el resultado sea 'true' y así el 'if' se cumpla.

```
22
23
24
25 boolean resultado = esta(v, numero); v: [1, 2, 3, 4, 5, 6] numero: 4 resultado: true
26
27
28
29
30
31
32
```

```
25 if (resultado == true) { resultado: true
26     System.out.printf("El elemento " + numero + " está en el vector " + Arrays.toString(v));
27 } else {
28     System.out.printf("El elemento " + numero + " no está en el vector" + Arrays.toString(v));
29 }
30
31
32 }
```

5. Modificar el valor de de resultado obtenido en el apartado anterior a false desde la zona de Variables. Describir el proceso realizado y capturar un pantallazo donde se pueda comprobar que el mensaje ya no es el adecuado "No está" y donde se pueda ver en la zona de variables que efectivamente se ha modificado el resultado de la función.

Aunque el valor esté dentro del rango, he modificado el valor booleano manualmente para que sea 'false' (en el apartado de variables del debug, he modificado el valor con 'Set value'). Por lo tanto, el 'if' no se ejecutará y saltará al 'else'.

```
25
26
27
28
29
30
31
32
33
34
35
36
37
```

```
25 if (resultado == false) { resultado: false
26     System.out.printf("El elemento " + numero + " está en el vector " + Arrays.toString(v));
27 } else {
28     System.out.printf("El elemento " + numero + " no está en el vector" + Arrays.toString(v));
29 }
30
31
32 }
```

```
1 usage
33 @
34 public static boolean esta(int[] vector, int elemento) {
35
36     for (int i = 0; i < vector.length; i++) {
37         if (vector[i] == elemento) {
38             return true;
39         }
40     }
41     return false;
42 }
```

```
Evaluate expression (Intro) or add a watch (Ctrl+Mayús+Intro)
```

```
@ args = (String[0]@827) []
> v = (int[6]@828) [1, 2, 3, 4, 5, 6]
> sc = (Scanner@829) "java.util.Scanner[delimiters=\p(javaWhitespace)+][position=13][match valid=true][need input=false][source closed=
@ numero = 3
@ resultado = false
```

**6. Establecer un breakpoint dentro de la función y realizar un pantallazo donde se pueda comprobar que se ha parado la ejecución y el contenido del vector recibido.**

Cuando el programa ha parado de pedir valores por medio del 'Scanner', la ejecución ha parado y ya se puede comprobar el apartado 'Variables'

```
10:42:55: Executing '!ejercicio_1.main()'...

> Task :compileJava UP-TO-DATE
> Task :processResources NO-SOURCE
> Task :classes UP-TO-DATE
Connected to the target VM, address: 'localhost:54525', transport: 'socket'

> Task :ejercicio_1.main()
Introduce el elemento para la posición 0 del vector
1
Introduce el elemento para la posición 1 del vector
2
Introduce el elemento para la posición 2 del vector
3
Introduce el elemento para la posición 3 del vector
4
Introduce el elemento para la posición 4 del vector
5
Introduce el elemento para la posición 5 del vector
6
```

```
13
14
15 for (int i = 0; i < v.length; i++) {   v: [1, 2, 3, 4, 5, 6]   i: 5
16     System.out.println("Introduce el elemento para la posición " + i + " del vector");
17     v[i] = sc.nextInt();
18 }
19
20 System.out.printf("Introduce el número a buscar en el vector");
21 numero = sc.nextInt();
22
```

7. Utilizando la zona de variables y una vez se ha parado en el breakpoint anterior modificar el contenido del vector para modificar el resultado previsto en la función. Si debía de ser true pasará a ser false y al revés. incluid los pantallazos necesarios para justificar dicho cambio.

He cambiado el valor original '3' a '9' manualmente para que dé como resultado final 'false' aunque haya introducido un valor correcto originalmente.



```

Evaluate expression (Intro) or add a watch (Ctrl+Mayús+Intro)

args = {String[0]@829} []
v = {int[6]@827} [1, 2, 9, 4, 5, 6]
  0 = 1
  1 = 2
  2 = 9
  3 = 4
  4 = 5
  5 = 6
sc = {Scanner@830} "java.util.Scanner[delimiters=\p{javaWhitespace}+][po
numero = 3
resultado = false
```