

Tema 3: Pruebas del software

Práctica 2: Debugger y JUnit

Primero: Debugger Eclipse

Usando **Eclipse**, crea un proyecto **Java** llamado **Enteros** e incluye dentro de ese proyecto las clases **Enteros** y **Principal** que se te proporcionan.

Establece un **punto de ruptura** o **breakpoint** en la línea del **while** del método **esPrimo()**:

```
while (primo && i <= n / 2) {
```

Usando el **debugger** de **Eclipse**, ejecuta el método **main()** de la clase **Principal** y recoge un pantallazo sin recortes de cada vez que la ejecución se pare en esa línea con valor de **275** en la variable **i**. Cada uno de esos pantallazos deberá llamarse **eclipse1.png**, **eclipse2.png**, etc.

En los pantallazos deberá verse claramente la perspectiva de depuración, al menos, las ventanas de código, variables, consola y traza.

Nota: Recuerda que durante la ejecución del **debugger** en **Eclipse**, la tecla **F8** te permite reanudar la ejecución hasta que vuelva a encontrar un **punto de ruptura**.

Segundo: Debugger Netbeans

Usando **Netbeans**, crea un proyecto **Java** llamado **Enteros** e incluye dentro de ese proyecto las clases **Enteros** y **Principal** que se te proporcionan.

Establece un **punto de ruptura** o **breakpoint** en la línea del **if** del método **divisible()**:

```
if (multiplo % divisor == 0) {
```

Usando el **debugger** de **Netbeans**, ejecuta el método **main()** de la clase **Principal** y recoge en un video sin sonido (llámalo **netbeans.mp4**), captado con **OBS** o una aplicación similar, la ejecución con los siguientes detalles:

- El **debugger** debe pararse en el punto de ruptura solamente cuando la variable **divisor** tenga el valor **275**. En esas paradas deberá mostrarse la ventana de variables.
- Al finalizar la ejecución se debe mostrar la consola con el texto de salida del programa.

Nota: Si las sugerencias estorban para ver los **puntos de ruptura**, puedes deshabilitarlas en **Tools/Options/Editor/Hints** y desmarcar **Suggestions**.

Tercero: Completa las pruebas con JUnit

En la anterior práctica de este tema hemos determinado el conjunto de casos de prueba que hay que aplicar a los métodos **factorial**, **divisible()** y **esPrimo()** pertenecientes a la clase **Enteros**.

Nota: Con esta práctica se te proporcionan los casos de prueba que se debieron obtener en la práctica anterior. Usa los casos de prueba proporcionados en lugar de los que tú obtuviste.

Ahora debes crear (dentro del proyecto **Enteros** de **Eclipse**, creado en el ejercicio Primero) un **Test JUnit** (**EnterosTest**) que ejecute todos esos casos de prueba. Para facilitar este primer encuentro con **JUnit**, no uses parámetros.

El objetivo final que perseguimos es encontrar posibles errores en los métodos que estamos testeando. **Si encuentras errores debes aportar la solución (respuestas.pdf) explicando:**

- qué problema hay
- cuál es la causa
- el nuevo código sin ese problema.

Cuarto: Pruebas JUnit parametrizadas

En el proyecto **Enteros** de **Eclipse**, crea una nueva clase de pruebas **TestEnterosParametrizados** que ejecute los mismos casos de prueba del ejercicio anterior (es decir, los de la solución proporcionada de la práctica 1), pero ahora usando parámetros.

Las entradas a los métodos parametrizados debes indicarlo con **@CsvSource** o **@ValueSource**. Utiliza tantos métodos **@ParameterizedTest** como sea necesario para incluir todas las pruebas

Último:

Una vez terminados los ejercicios, deberás entregar todos los ficheros generados (imágenes, vídeo, todas las clases **.java** creadas, la clase **Enteros.java** modificada y un **PDF** (**respuestas.pdf**) con las respuestas del ejercicio tercero en el caso de que encuentres errores), en una carpeta con el nombre **apellido1_apellido2**. Asegúrate que está limpia en el sentido de que sólo incluye los ejercicios de este boletín y los ficheros tienen los nombres que se han pedido. Comprime la carpeta en un **ZIP** llamado **apellido1_apellido2.zip** y súbelo con el enlace correspondiente de la plataforma **Moodle**.

EL INCUMPLIMIENTO DE ESTAS INSTRUCCIONES PUEDE TENER COMO CONSECUENCIA LA NO CORRECCIÓN DE LA PRÁCTICA.