

Entornos de Desarrollo

Examen evaluación 3

Pregunta 2 - A

Se han realizado los siguientes cambios en cuanto a código limpio y refactorización:

- **Renombrado de la clase** (PasCal) para mejorar su semántica utilizando el atajo de refactorización de Visual Studio: **Ctrl + R + R**.
- En renombrado del constructor se produce de forma automática gracias al comando anterior.
- **Renombrado del fichero** para coincida con el nombre de la clase. De haber habido otra clase dentro del mismo fichero se habría extraído a uno nuevo y se hubiera seguido esta misma convención.
- **Renombrado de los miembros de clase**. En lugar de utilizar el nombre semántico del miembro en un comentario, lo utilizamos directamente como nombre del atributo. Ya no se hacen necesarios los comentarios “in line”.
- **Cambio de privacidad** de los miembros de la clase a “private” y extraída su consulta a getters.
- **Refactorizamos el constructor vacío** para que cada asignación tenga su línea exclusiva.
- **Refactorizamos el constructor sobrecargado** para renombrar su parámetro (caMel) y añadir las llaves faltantes en el if/else.
- **Renombrado del método** para calcular las estadísticas (PasCal) a un nombre más semántico.
- **Renombrado del parámetro** del método para calcular estadísticas (caMel).
- **Inclusión de llaves** faltantes en el método del cálculo de estadísticas.
- **Corrección de espaciados** entre variables, asignaciones y operadores en el método de cálculo de estadísticas.
- **Comentarios “in line”** dentro de métodos y constructores movidos una línea exclusiva.

Pregunta 2 - B

Se ha hecho una refactorización para que el constructor sobrecargado use el método CalcularEstadísticas() ya que el código estaba duplicado.

Pregunta 2 – C

Al ejecutar por primera vez los test vemos que nos devuelve un OutOfRangeException. Esto es debido a que durante la comprobación de si todas las notas están entre 0 y 10 recorreremos un “for” de 10 iteraciones. Este número de iteraciones están “hardcodeadas” con lo que lo sustituiremos por el valor de la longitud de la lista de notas.

Una vez corregido este error y ejecutado otra vez la prueba vemos que no coinciden el número de sobresalientes esperados. Esto se debe a que en la comprobación no incluimos el 9 como nota a tener en cuenta, solo los mayores a nueve. Cambiamos el operador “>” por “>=”.

Pregunta 2 – D

Se han reemplazado los “return -1” del método “CalcularEstadisticas” por lanzamientos de excepciones que serán usados en los métodos de prueba implementados. Estos casos son:

- Lista vacía.
- Notas inválidas.

Estos dos casos se han identificado como clases de equivalencia y se han incluido en la batería de pruebas.

Pregunta 2 – E

La clase ha sido documentada en el código.

Pregunta 3

Clases de equivalencia para el método CalcularEstadisticas(lista):

- Lista sin elementos.
- Lista con elementos entre 0 y 10.
- Lista con elementos menores a 0.
- Lista con elementos mayores a 10.

Casos frontera:

- Lista con un solo elemento.
- Lista con alguna nota igual 0.
- Lista con alguna nota igual a 10.

Se han implementado casos de prueba para todos los casos identificados.