Antonio José Picó Pina

**Exámen Entornos de Desarrollo 2ª EV.**

1. **Encontrar cinco errores de normas de estilo en el fichero loto.cs, indicando número de línea, error encontrado y solución.**

* Línea 7:

public class loto

El nombre de la clase debe empezar por mayúscula (notación Pascal)

Solución: Cambiar el nombre por “Loto”.

* Línea 15:

public bool ok = false;

El nombre de la variable es poco descriptivo.

Solución: Cambiar el nombre por “combinacionValida”.

* Línea 26:

Random r = new Random();

El nombre de la variable no es descriptivo.

Solución: Cambiar el nombre por “aleatorio”.

* Línea 33:

for (j=0; j<i; j++)

No hay separación con los elementos de comparación.

Solución: for (j = 0; j < i; j++)

* Línea 48:

public loto(int[] misnums)

Nombre del parámetro sin notación camelCase.

Solución: public loto(int[] misNums)

* Línea 68:

ok=true;

Mala indentación.

Solución: Debe tabularse hacia la derecha a la altura de la llave de la línea anterior.

1. **Si existen, detectar y aplicar al menos tres patrones de refactorización (tanto en el fichero Loto.cs como en el fichero Form1.cs), indicando el patrón que se aplica y, si es posible aplicarlo con Visual Studio, la opción que se usa.**

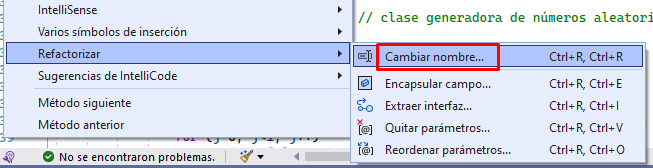
* Renombrar

Cambiamos el nombre de la variable “ok” definida en la línea 15:

public bool ok = false;

Por “combinacionValida”.

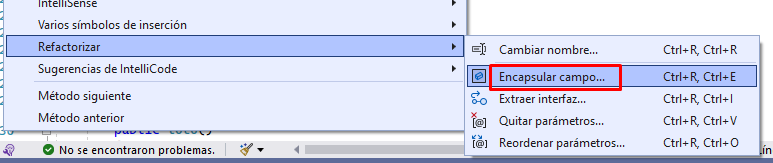
Mediante la opción “Cambiar nombre” del menú Editar > Refactorizar.

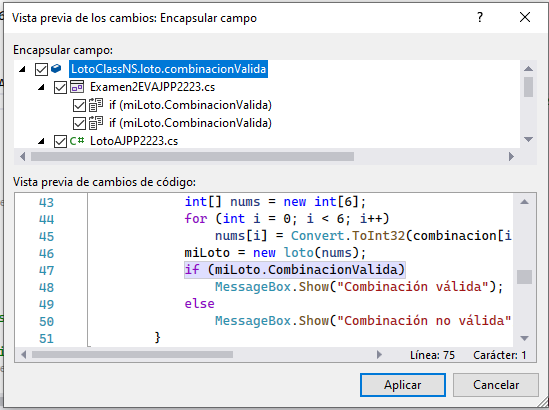


* Encapsulación del mismo atributo (ahora “combinacionValida”).

Hay que pasarlo a private y crear la propiedad.

Se puede hacer desde Editar > Refactorizar > Encapsular campo…









* Extracción de método.

Como en el constructor con parámetros hay muchas condiciones y for concatenados optamos por extraer en un método la búsqueda de la repetición del número.

