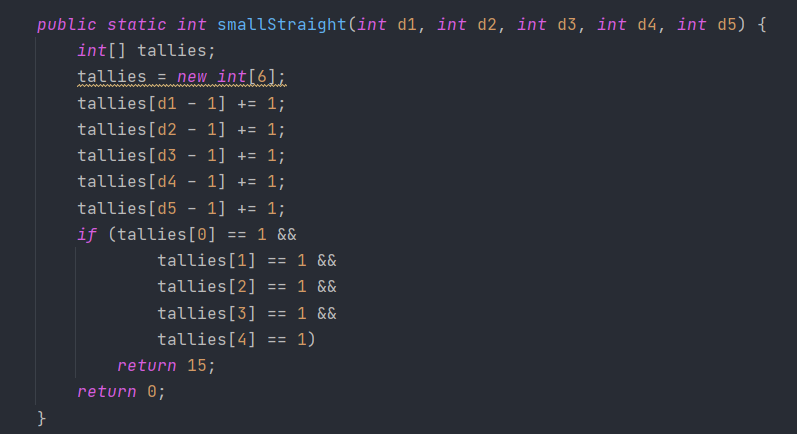
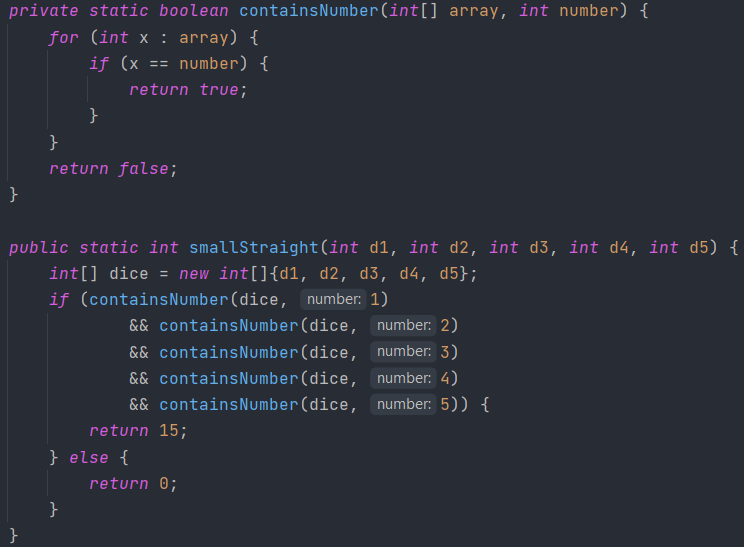
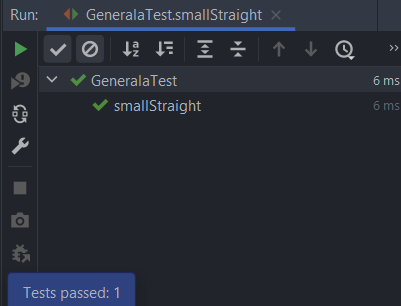
1. Refactorizando **smallStraight**

Antes de refactorizar:

Se codifica un booleano que chequea si un arreglo contiene algún número en específico y luego se llama dentro de **smallStraight**. Esto se utilizó para reducir la cantidad de líneas del método y hacer mucho más claro lo que se está haciendo dentro de este. El resultado es el siguiente:

Se comprueba que la prueba siga funcionando:



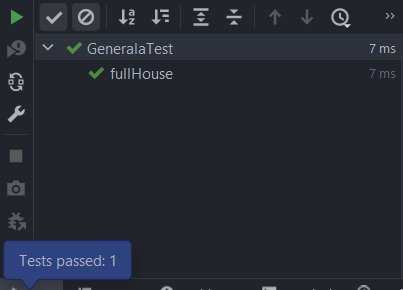
1. Refactorizando **fullHouse**

Para la refactorización de **fullHouse** se codificaron cuatro métodos más pequeños:

* **int getFrequency**: Este método retorna la cantidad de veces en la que se repite un número dentro de un arreglo de enteros.
* **boolean isTwoOfKind**: Haciendo uso de **getFrequency**, si un número se repite dos veces retorna verdadero.
* **boolean isThreeOfKind**: Haciendo uso de **getFrequency**, si un número se repite tres veces retorna verdadero.
* **int sumAllNumbers**: Recibe un arreglo de enteros y suma todos sus componentes.

Lo que resulta en que el método **fullHouse** sea mucho más corto y facilita su comprensión.

Se comprueba que la prueba siga funcionando:

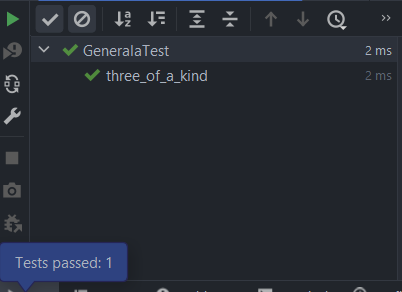


1. Refactorizando **three\_of\_a\_kind**

Como ya contamos con una condicional que verifique si un número se repite tres veces podemos implementarla en este método. Una vez verificado el número que se repite tres veces se implementaron los siguientes métodos:

* **Int getRepeatedTimes**: retorna la cantidad de veces que se repite un número.
* **Int getRepeatedNumber**: retorna un número que se repita más de una vez.
* **Int getRepeatedSum**: Retorna getRepeatedTimes\* getRepeatedNumber

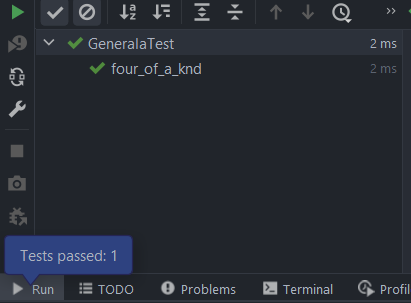
Se comprueba que la prueba pase:



1. Refactorizando **four\_of\_a\_kind**

De la misma manera que con **three\_of\_a\_kind**, se codifica un nuevo booleano **isFourOfKind** que compruebe que un número se repita cuatro veces y haciendo uso de los mismos métodos definidos en el punto anterior se obtiene un método más corto y legible.

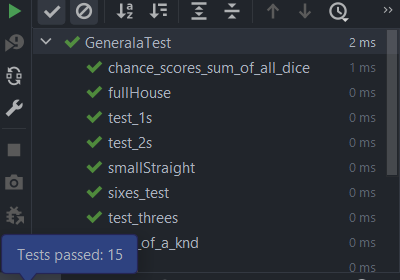
Se comprueba que la prueba siga pasando:



1. Refactorizando **two\_pair**

Se codificaron dos nuevos booleanos **isPair** e **isTwoPairs** para verificar si un arreglo contiene dos pares de números repetidos.

Se comprueba que todas las pruebas corran:

****