

Selección: He escogido analizar el framework *JUnit*, con la intención de comprender la aplicación del patrón de *decorator* (también conocido como *wrapping*). Enlace del repositorio: <https://github.com/junit-team/junit5>.

Descripción

JUnit es un framework que permite realizar pruebas de forma constante y sencilla al código del desarrollador (esto en Java, cabe recalcar). JUnit se basa en el método de Unit Testing para simplificar el análisis y chequeo de código escrito; esto es, en pocas palabras, realizar pruebas a fragmentos de código de tamaño considerablemente “reducido”, para poder escalar de forma progresiva las pruebas a todo el proyecto que se está llevando a cabo.

Análisis

En específico, se puede dar una vista a la descripción proporcionada para la interfaz *BeforeEach* (<https://github.com/junit-team/junit5/blob/main/junit-jupiter-api/src/main/java/org/junit/jupiter/api/BeforeEach.java>). Esta última es una de las anotaciones (*decorators*) que JUnit permite hacer para extender el comportamiento atribuido a métodos de clases de prueba (también conocidos como *TestCases*). La idea general es que cada una de las anotaciones que se pueden añadir en una *TestCase* va a editar o priorizar el comportamiento de cierto método. En su caso, *BeforeEach* hace que siempre que se ejecute una clase de pruebas, aquel método se llamara previamente.