**Mockito Tool**

**Prework**

* Aplicar técnica de fixtures para proveer información adicional en nuestras pruebas.
* Aplicar el patrón de diseño factory para crear entidades y modelos dinámicamente en nuestras pruebas.
* Utilizar mockito tool para generar mocks en nuestras pruebas.

**Mockito** es una librería Java que permite simular el comportamiento de una clase de forma dinámica. De esta forma nos aislamos de las dependencias con otras clases y sólo testeamos la funcionalidad concreta que queremos**.**

**Mock:** un mock es un objeto simulado.

**Método setUp()**

Se usa para inicializar objetos. Puedes hacerlo creando un método public void setUp () en el que podrás declarar las variables de instancia para los objetos comunes. Coloca este método setUp () debajo de la anotación @BeforeEach. Con la anotación @BeforeEach, el framework ejecutará el método setUp() antes de la ejecución de cada caso de prueba.

**Método tearDown()**

Podría agregarse para la limpieza de los objetos después de que se haya completado la ejecución del caso de prueba. Se puede agregar un método public void teardown () debajo de la anotación @AfterEach. JUnit se asegura de que después de que se ejecute cada caso de prueba, el método en @AfterEach se ejecute con seguridad.

**¿Qué es un fixture?**

Es cualquier herramienta de prueba que te permite configurar los datos de prueba en un archivo separado de las pruebas en sí.

Crear un fixture es crear un conjunto de objetos inicializados a ciertos estados.

Si un grupo de pruebas requiere diferentes fixtures de prueba, puede escribir código dentro del método de prueba para crear su propio dispositivo de prueba.

Si un grupo de pruebas comparte los mismos fixtures, debe escribir un código de configuración por separado para crear el dispositivo de prueba común.

El propósito de usar Test Fixture es eliminar la duplicación del código común para todos los casos de prueba.

**Work**

* Aplicar la técnica de fixtures para proveer información adicional a nuestras pruebas
* Aplicar el patrón de diseño factory para crear entidades y modelos dinámicamente en nuestras pruebas
* Utilizar mockito tool para generar mocks en nuestras pruebas

**Postwork**

* Añadir la funcionalidad de resta a la clase SomeBusinessLogic.
* Añadir pruebas con inyección de mocks.
* Reconocer las herramientas de mockito.