**JUnit and TestNG**

**Prework**

* Utilizar JUnit.
* Utilizar TestNG.
* Comparar las herramientas JUnit y TestNG.

JUnit es un framework que se utiliza para hacer pruebas unitarias de software en Java.

TestNG es un framework para pruebas y testing que trabaja con Java y está basado en JUnit (para Java) y NUnit (para .NET).

|  | **JUnit** | **TestNG** |
| --- | --- | --- |
| **Desarrollado por** | JUnit fue desarrollado por Kent Beck, David Saff, Erich Gamma. Erich Gamma y Kris Vasudevan. | TestNG es un marco de prueba desarrollado por Cédric Beust. |
| **Fuente abierta** | JUnit es un marco de código abierto que se utiliza para activar y escribir pruebas. | TestNG es un marco basado en Java que es una opción mejorada para ejecutar pruebas. |
| **Ejecuciones de prueba en paralelo** | JUnit no admite la ejecución de pruebas paralelas. | TestNG puede ejecutar pruebas paralelas. |
| **Admite anotación** | No es compatible con la anotación avanzada. | Es compatible con la anotación avanzada. |
| **Pruebas de dependencia** | Faltan las pruebas de dependencia en JUnit. | Las pruebas de dependencia están presentes en TestNG. |
| **Pruebas de agrupación** | No es posible agrupar pruebas en JUnit. | Las pruebas se pueden agrupar y ejecutar en paralelo. |
| **Facilidad de uso** | La ejecución de pruebas necesita una cierta dependencia de JUnit. | Escribir pruebas y configurarlas es más fácil en TestNG que en JUnit. |

**Work**

* Utilizar JUnit
* Utilizar TestNG
* Comparar las herramientas JUnit y TestNG

**Diferencias entre JUnit 4 y JUnit 5**

* **Importaciones**

JUnit 5 usa el nuevo org.JUnit.jupiter paquete para sus anotaciones y clases. Por ejemplo, org.JUnit.Test se convierte en org.JUnit.jupiter.api.Test.

* **Anotaciones**
* **Arquitectura**

JUnit 4 tiene todo incluido en un solo archivo jar.

JUnit 5 se compone de 3 subproyectos, es decir, JUnit Platform, JUnit Jupiter y JUnit Vintage.

* **JUnit Platform**

Define la API TestEngine para desarrollar nuevos marcos de prueba que se ejecutan en la plataforma.

* **JUnit Júpiter**

Tiene todas las nuevas anotaciones JUnit y la implementación de TestEngine para ejecutar pruebas escritas con estas anotaciones.

* **JUnit Vintage**

Para admitir la ejecución de pruebas escritas JUnit 3 y JUnit 4 en la plataforma JUnit 5.

* **Aserciones (Assertions)**

En Junit 4, org.junit.Assert tiene todos los métodos de aserción para validar los resultados esperados y resultantes. Aceptan un parámetro adicional para el mensaje de error como PRIMER argumento en la firma del método.

En JUnit 5, org.junit.jupiter.Assertions contiene la mayoría de los métodos de aserción, incluidos los métodos adicionales de assertThrows () y assertAll (). assertAll () está en estado experimental a partir de hoy y se usa para aserciones agrupadas. Los métodos de aserciones de JUnit 5 también tienen métodos sobrecargados para admitir que se imprima un mensaje de error en caso de que la prueba falle

* **Suposiciones (Assumptions)**

En Junit 4, org.junit.Assume contiene métodos para establecer suposiciones sobre las condiciones en las que una prueba es significativa. Tiene los siguientes cinco métodos:

* + assumeFalse()
  + assumeNoException()
  + assumeNotNull()
  + assumeThat()
  + assumeTrue()

En Junit 5, org.junit.jupiter.api.Assumptions contiene métodos para establecer suposiciones sobre las condiciones en las que una prueba es significativa. Tiene los siguientes tres métodos:

* + assumeFalse()
  + assumingThat()
  + assumeTrue()

**Postwork**

* Comparar el funcionamiento de JUnit y TestNG en escenarios similares
* Analizar cómo funciona la ejecución en paralelo en JUnit 5
* Analizar cómo funciona la ejecución en paralelo en TestNG