**Práctica Sesión 2 Junit.**

Usando el proyecto de Laboratorio de Unit Tests, se requiere agregar algunos unit tests para las clases existentes.

**Práctica 1.**

**Aritmetica**

* Agregue unit tests para cada uno de los métodos de la clase Aritmética
* Agregue además unos métodos que utilicen @BeforeEach, @AfterEach y @BeforeAll, puede simplemente imprimir un mensaje en consola para ver cuando se ejecutan.

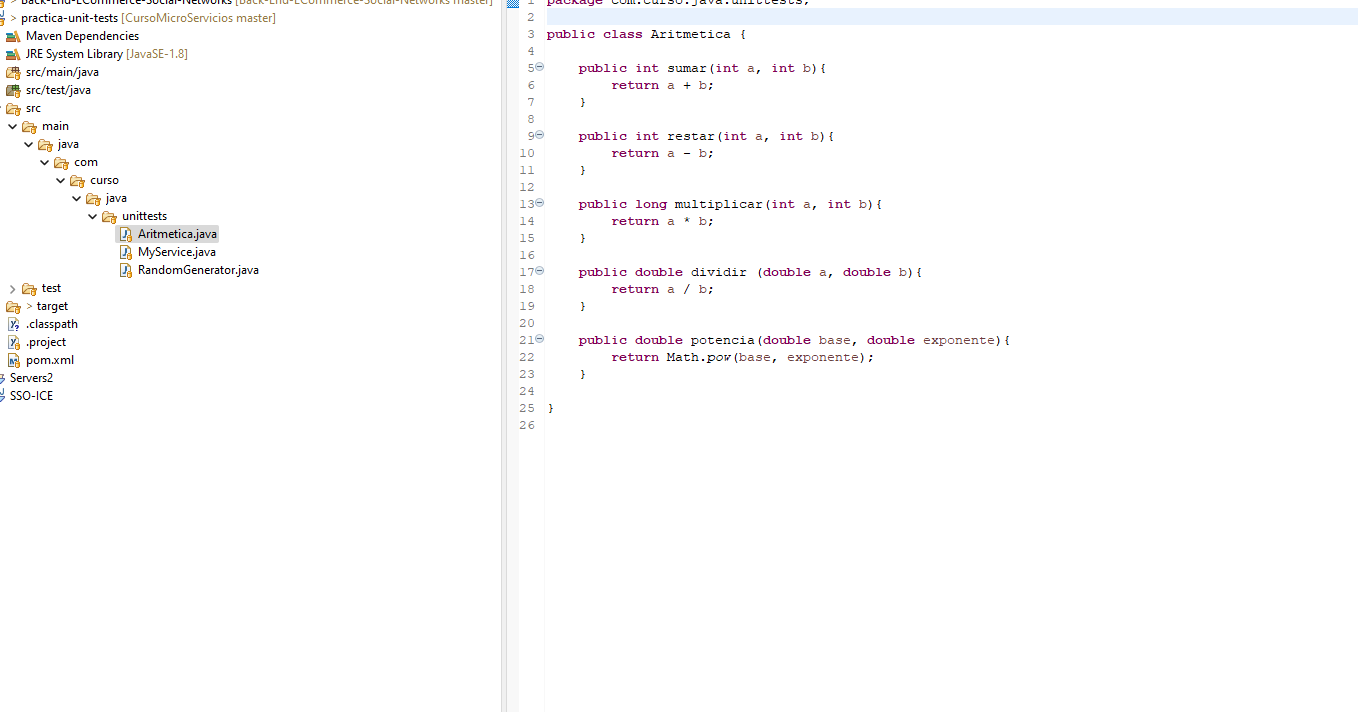
**Práctica 2.**

**MyService** En estos escenarios deseamos usar Mockito

* test -> evenOrOdd\_even: Debe validar que cuando randomGenerator retorna un número par, el método evenOrOdd retorne el string "El numero es par"
* test -> eventOrOdd\_odd: Debe validar que cuando randomGenerator retorn un número impar, el método eventOrOdd retorne el string "El numero es impar"
* test -> convertirString\_ok: Debe validar que si el método convertirString recibe el parámetro "Hola Mundo", el resultado esperado debe ser "HOLA MUNDO:ABCDEF" y que si el parámetro "Java Rules" es recibido, entonces el parámetro debe ser "JAVA RULES:123ZXY". Además verifique que el método generator.getRandomString fue llamado 2 veces.

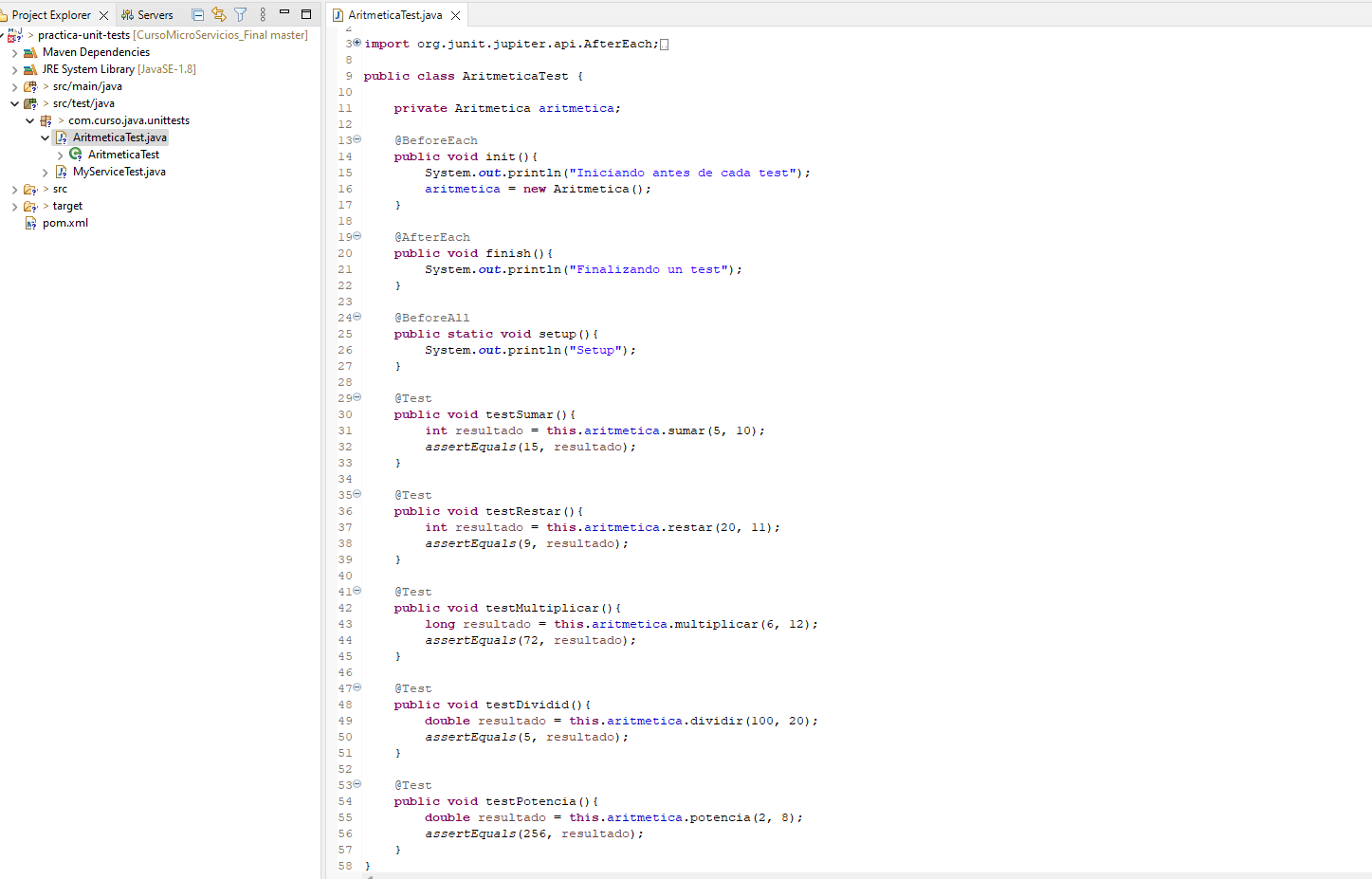
**Bonus** Pruebe correr los unit tests con el comando de maven.

**Solución Páctica 1.**

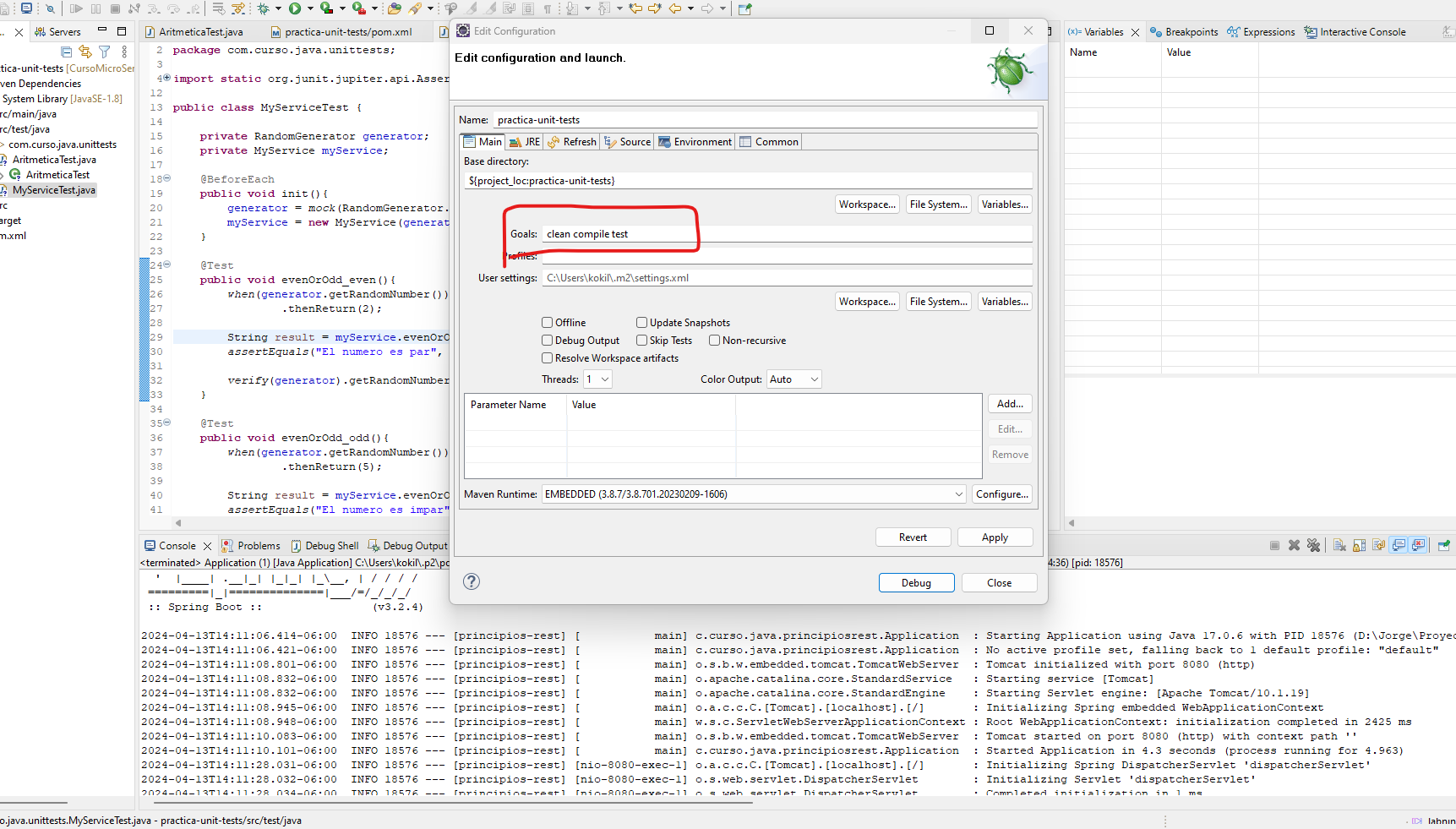
****

Primero se nos pide agregar los unit tests para la clase Aritmética, vamos al archivo en src/main y lo abrimos para ver el contenido.

Creamos los test com.curso.java.unittests.



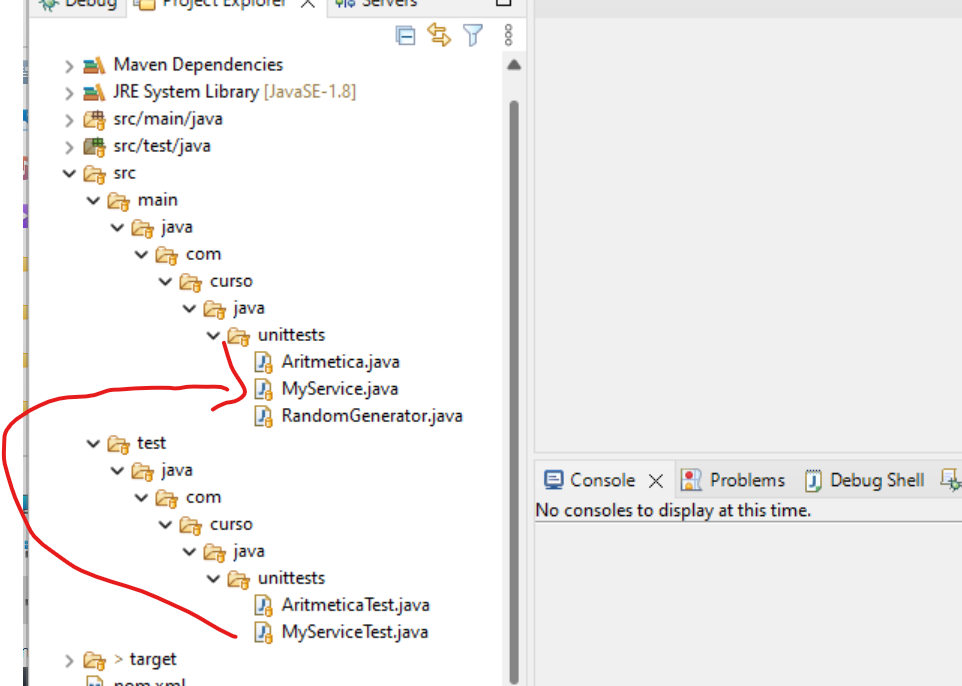
Después corremos clean compile test para correr el ciclo de vida hasta los test de lo que generamos.



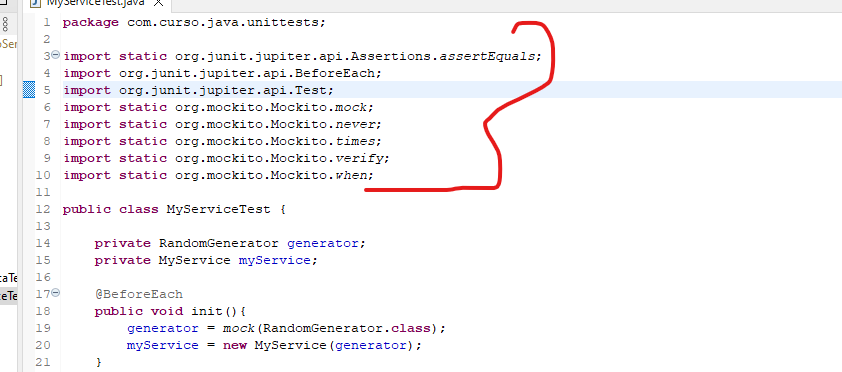
Hacemos todos los tests con la librería de Junit utilizando las anotaciones @AfterEach, BeforeEach, @BeforeAll y los correspondientes assertEquals para cada operación.

**Solución Práctica 2.**

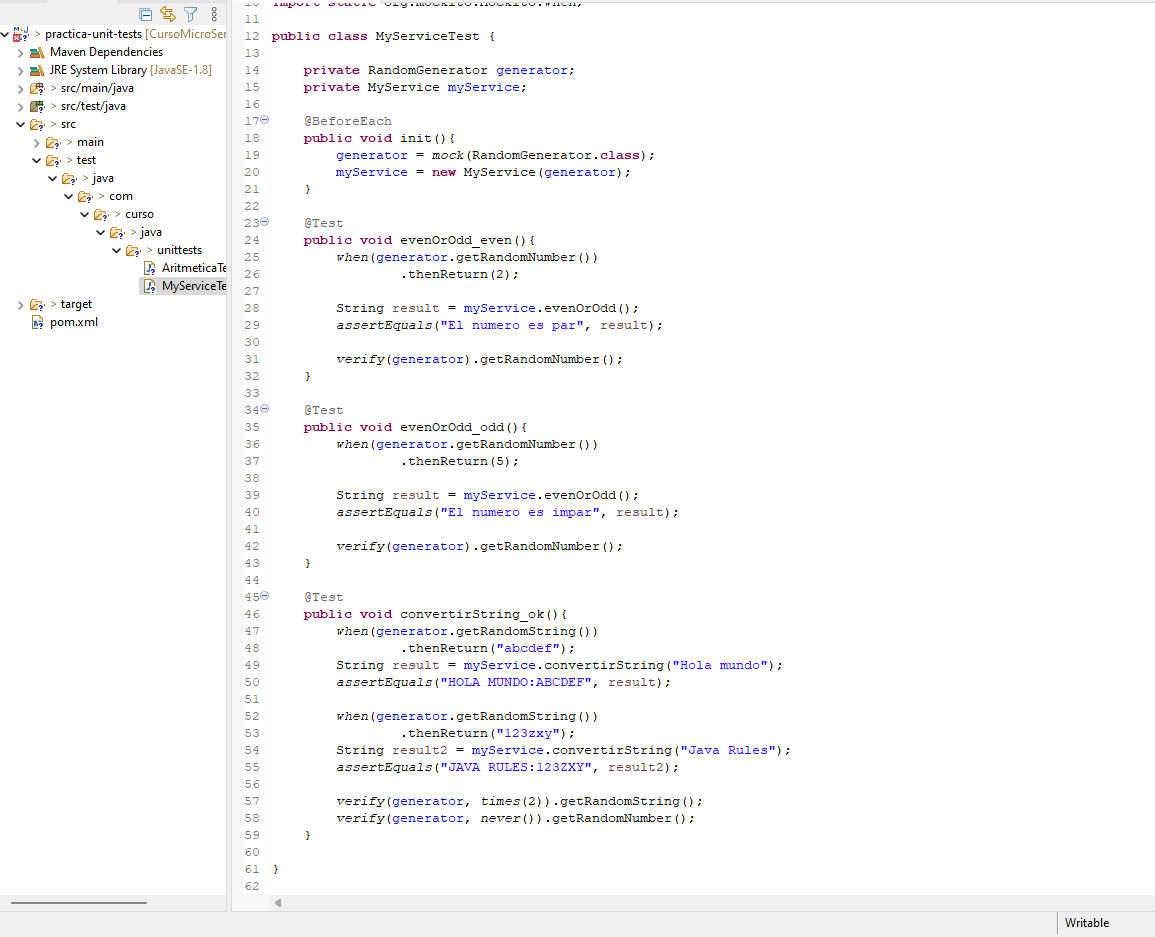
Creamos MyServiceTest en al mismo directorio del main pero de las clases test ya que está creada fines de test.

****

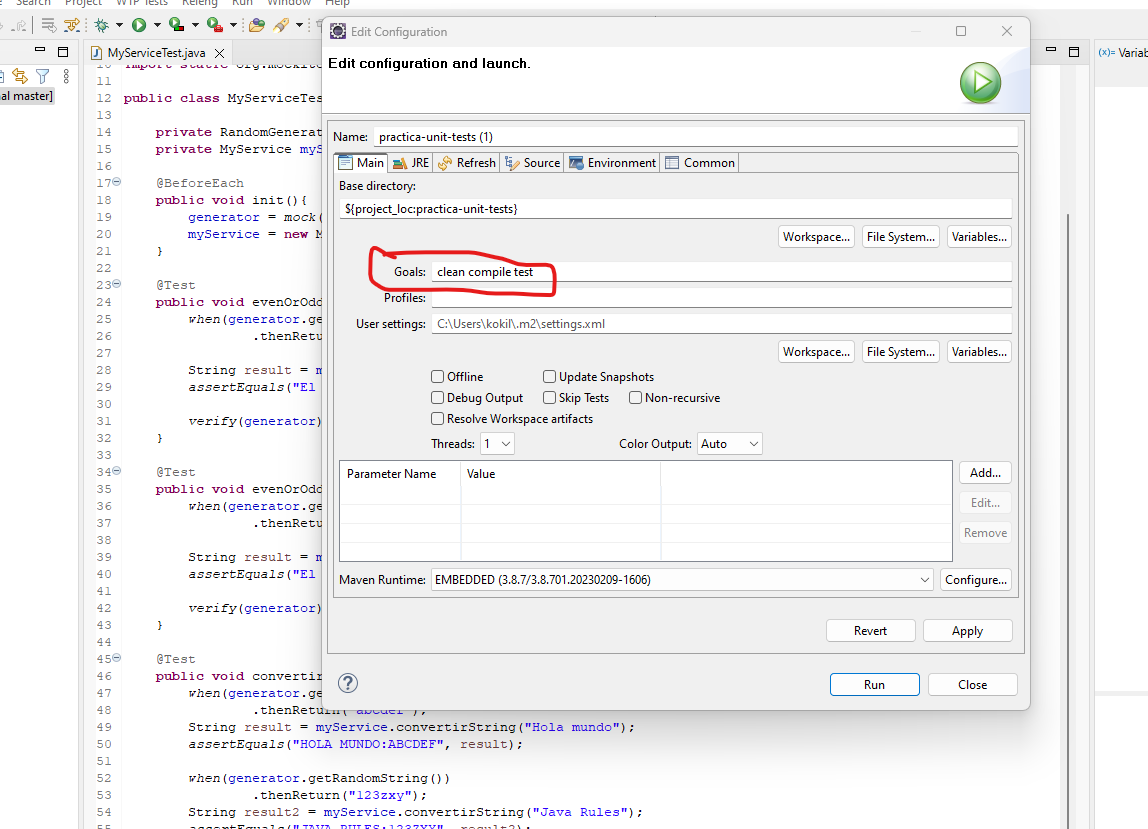
En la nueva clase MyServiceTest deben importarse las librerías siguientes:

****

Implementar sus test con las librerías de mockito y los métodos when…. Para simular comportamientos de los mockitos.

****

Usamos el goal clean compile test:



Revisamos que nuestros test fueron exitosos para MyServiceTest.

