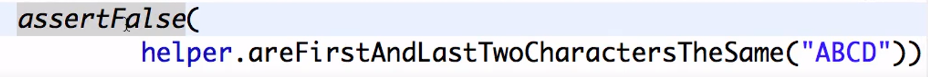


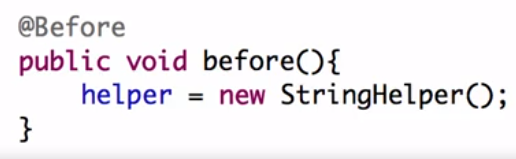
Test only accept public void methodName, no other signature





@Before

Execute the method annotated with this before each of the unit tests

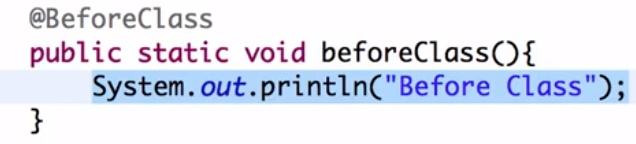


@After

Execute a method annotated with @After after each of the test executed.

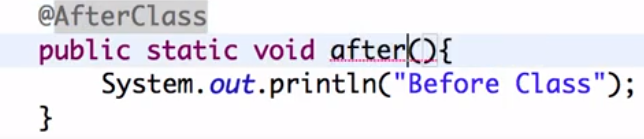
@BeforeClass

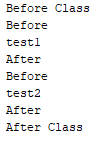
Executed before all test run, only once (must be static)



@AfterClass

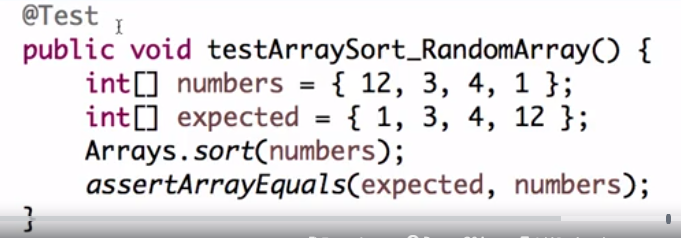
Executes once after all tests have run must be static



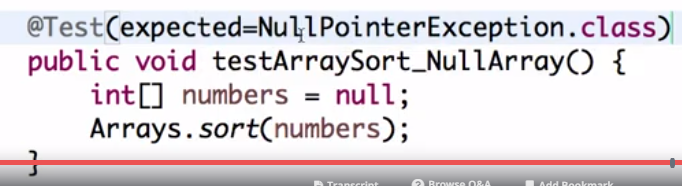


assertArrayEquals

allows to compare the content of two arrays

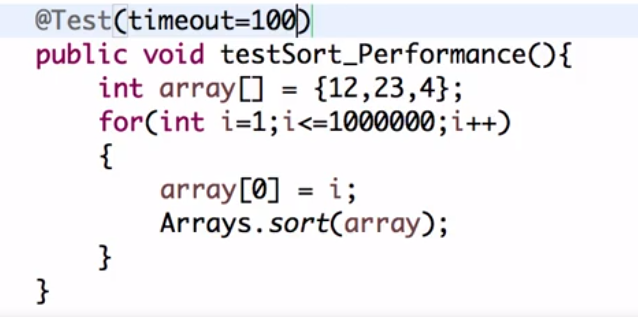


Testing Exceptions



Test Performance

Test will pass if its completed within the timeout specified



Parameterized Test

Allows to test different values for the same test



testCondition (you can use any name) will provide the input and output to the class which will be used on the test, it will run a test for each element in the collection.

Junit Suit

Allows to organize a group of Junit tests to run together

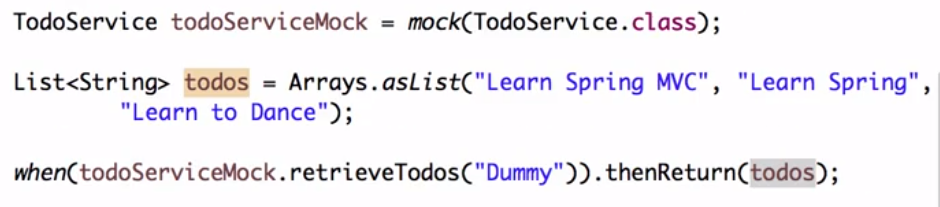


Mockito

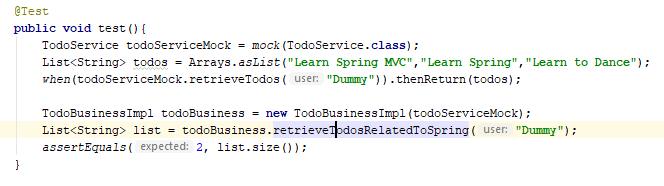
Stub

Sample implementation

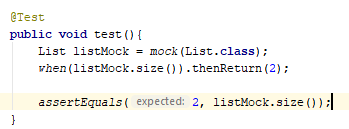
Mock, creating objects that simulate the behavior of real objects



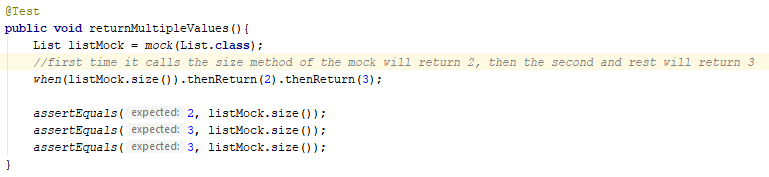
When retrieveTodos method from todoServiceMock is called will return todos.



Mock method from List interface

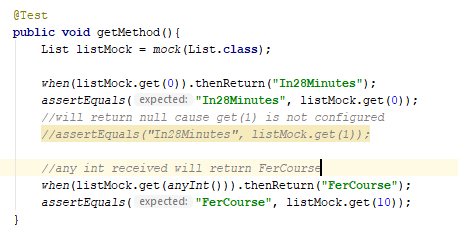


Return different values for same method

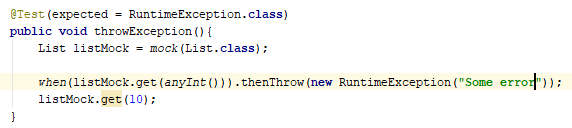


anyInt() (anyChar,AnyLong, any, any(someSpecificClass), AnyList,AnySet,AnyMap, anyString, anyObject, anyBoolean available)

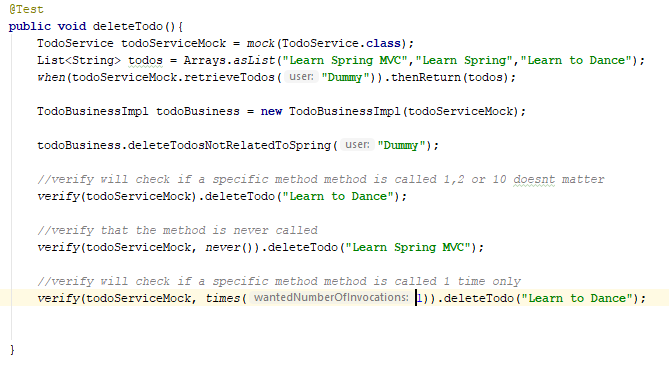
on any integer sent will return same output



Return an exception with then throw

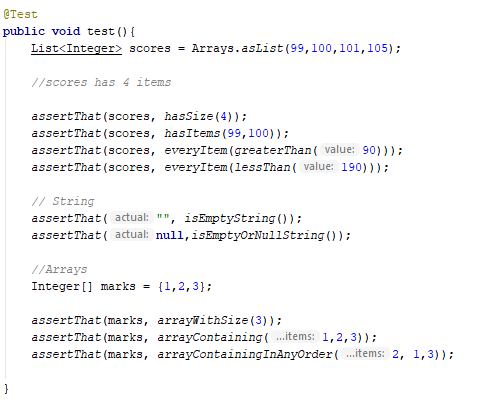


Verify will check if a method was called or not (with never()) or how many times you expected it to be called (times()). Also available atLeast(5) to specify you want that method to run at least those times



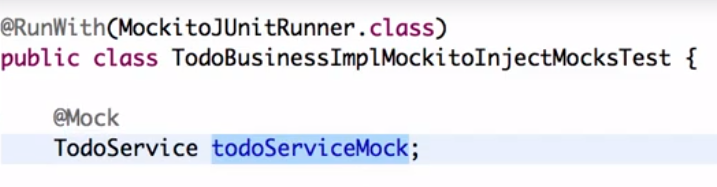
Hamcrest Matchers

Variety of methods to validate lists, arrays, string, collections on general and for each item



@Mockito

Whenever there is a test class with @RunWith(MockitoJUnitRunner.class), it will look for @Mock annotation and will inject that dependency into the class



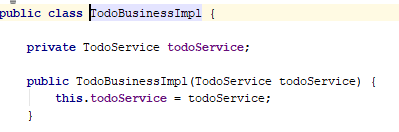
Will inject a TodoService into todoServiceMock

@InjectMocks

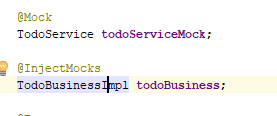
Whenever there is an object annotated with inject mocks, it will look in the class type for dependencies required, if one annotated with @Mock is available, it will pass that to the annotated with inject class



TodoBusiness from TodoBusinessImpl is annotated with @InjectMocks

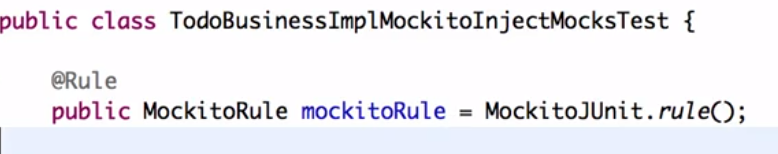


It checks TodoBusinessImpl has a instance var of type TodoService



Checks in the test a variable of that type annotated with @Mock, so pass it to TodoBusinessImpl for the test

Only one @RunWith per class, if you need to use different on the same class you should use rules

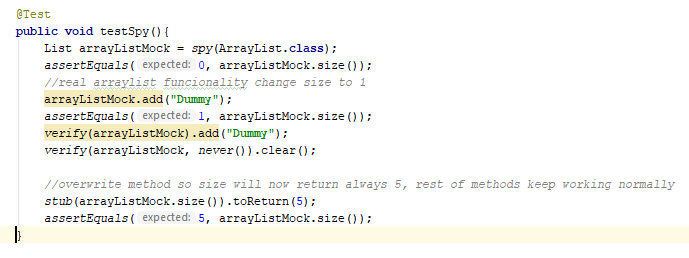


Added the rule for Mockito and remove @RunWith, if you need to add another kind of rule, then you can do it

Spy

to spy on a real object.

This will allow us to call all the normal methods of the object while still tracking every interaction, just as you would with a mock.



Stub method

Allows to return something on determined method call



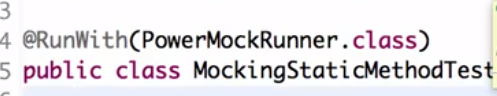
Mockito doesn’t mock private methods

Cannot mock final classes

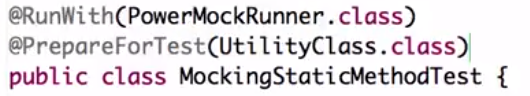
Cannot mock final methods

Powermock to mock static and private methods

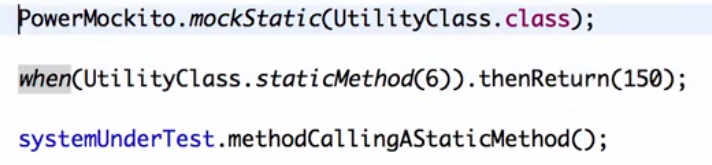
When you want to combine Mockito and powemock, you need to use a specific runner



mock the class with the static method for mocking



Utility class is the class with the static method



Mock the static method



Invoke private methods

Using Whitebox from PowerMockito we can access the private method



Mock Constructor

PrepareForTest has to be configured, but not for the class which the constructor belong, the class which will call the new object of that class is the one to configure. Ex need to mock the constructor of ArrayList which will be created in TestClass, then TestClass should be configure in PrepareForTest