Ricardo Luna Guerrero. A01561680

Tec de Monterrey Campus Chihuahua

Software Quality and Testing

Lab 3. Mocking Around

1. **Explain what happened with variables annotated as @Mock**

   The @Mock annotation is used to create and inject mocked instances. We do not create real objects, rather ask Mockito to create a mock for the class. The @Mock annotation is an alternative to Mockito.mock(classToMock). They both achieve the same result. Using @Mock is usually considered "cleaner". [1] A mock is an object that simulates the behavior of a real object in controlled ways.

2. **What is the purpose of the tuple when/thenReturn?**

   The purpose of the tuple when/thenReturn is to configure simple return behavior for a mock. This method chain is used to specify a condition and a return value for this condition.

3. **What is the difference of the objects annotated as @Mock and @Spy?**

   The main function of using mocks is that they give full control over the behavior of the mocked objects. Spies are known as partially mock objects. It means spy creates a partial object or a half dummy of the real object by stubbing or spying the real ones. [2] Mock objects replace mocked classes entirely, returning recorded or default values, in spying you can take an existing object and replace only some methods.

4. **When is useful to use the @InjectMocks annotation?**

   @InjectMocks creates objects and inject mocked dependencies. You shall use it to create instances which needs to be tested in test class, when actual method body needs to be executed for a given class and when need all internal dependencies initialized with mock objects to work method correctly. [3]

5. **Give an example of a scenario when is a good idea to use Mocks**

   Mocking is very used in Unit Testing. You may have an object under test that have a lot of dependencies on other objects. To isolate the behavior of the object you should replace the other objects by mocks that simulate the behavior of the real objects and help you to focus on the current object's testing.

References

[1] https://howtodoinjava.com/mockito/mockito-annotations/

[2] https://www.javatpoint.com/mock-vs-stub-vs-spy

[3] https://howtodoinjava.com/mockito/mockito-mock-injectmocks/