Lab Assignment 11: Code Implementation and Testing using JUnit and Mockito frameworks

Part 1:

1. **Implementing Unit Tests with the JUnit framework: (10:20 - 11:10)**

   Implement code for a component named, ArrayFlattener, as a Java command-line (console) application. In your component (class), implement a method named, flattenArray(), which takes as input, a two dimensional (2-D) nested array and returns as output, the flattened array (i.e. a one dimensional array containing the nested arrays, flattened). For example, given an input a_in = [[1,3], [0], [4,5,9]], your method should return the input elements in a new flattened array as, a_out = [1,3,0,4,5,9].

   Using JUnit, implement unit tests for the/your ArrayFlattener component and its flattenArray() method; covering the following 2 test-cases:

   a. When the input is a legit 2-D nested array, [[1,3], [0], [4,5,9]].

   b. When the input is a null.

   Create a JUnit TestSuite containing your 2 test-cases defined above.

2. **Implementing Unit Testing and mocking a remote service, using JUnit and Mockito frameworks:**

   Implement code for a component named, ArrayReversor, as a Java command-line (console) application. In your component (class), implement a method named, reverseArray(), which takes as input, a two dimensional (2-D) nested array and returns as output, the reverse of the flattened array (i.e. a one dimensional array containing elements of the nested array, flattened and reversed). For example, given an input a_in = [[1,3], [0], [4,5,9]], your method should return the input elements in a new flattened and reversed array as, a_out = [9,5,4,0,3,1].

   **Note**: In order to reverse a given 2-D nested input array, your component needs to first flatten it. To perform the flattening, your solution should use a service named, ArrayFlattenerService; with a method named, flattenArray, which takes a 2-d nested array, flattens it and returns a flattened 1-d array. However, this service is remote and/or unavailable; therefore you will have to implement code to create a mock of it and then use the mock in implementing the unit testing of your ArrayReversor component.

   Using JUnit, implement unit tests for your ArrayReversor component and its reverseArray() method; covering the following two test-cases:

   a. When the input is a legit 2-D nested array, such as [[1,3], [0], [4,5,9]].

   b. When the input is a null.

   In both of your unit test-cases, include code that verifies that the ArrayFlattenerService was

indeed invoked and not just by-passed (e.g. by harding-coding the correct expected result/output from its invocation).

Finally, create a JUnit TestSuite named, ArrayReversorTestCases, containing your 2 test cases.


//-- The End --//