

## QLC-2.3) Limitations of Stubs tests

```
@Test
public void
find_highest_score_with_array_of_many_return_array_of_many_using_stub()
{
    //[{"Physics", { 56, 67, 45, 89} }, {"Art", { 87, 66, 78} },
    {"Comp Sci", { 45, 88, 97, 56} }]
    // Arrange
    int[] physics_scores = { 56, 67, 45, 89 };
    String physics = "Physics";
    int[] art_scores = { 87, 66, 78 };
    String art = "Art";
    int[] compSci_scores = { 45, 88, 97, 56 };
    String compSci = "Comp Sci";
    ArrayList<TopicScores> topicScores = new ArrayList<>();
    topicScores.add(new TopicScores(physics, physics_scores));
    topicScores.add(new TopicScores(art, art_scores));
    topicScores.add(new TopicScores(compSci, compSci_scores));
    // Use a stub version of HighestNumberFinder
    com.s2s.demos.topicmanager.HighestNumberFinder hnf =
        new com.s2s.demos.topicmanager.
HighestNumberFinder();
    TopicManager cut = new TopicManager(hnf);
    ArrayList<TopicTopScore> expectedResult = new ArrayList<>();
    expectedResult.add( new TopicTopScore(physics, 89));
    expectedResult.add( new TopicTopScore(art, 87));
    expectedResult.add( new TopicTopScore(compSci, 97));

    // Act
    ArrayList<TopicTopScore> result = cut.findTopicHighScores
(topicScores);

    // Assert
    assertEquals(expectedResult.get(0).getTopicName(), result.get(0).
getTopicName());
    assertEquals(expectedResult.get(0).getTopScore(), result.get(0).
getTopScore() );
    assertEquals(expectedResult.get(1).getTopicName(), result.get(1).
getTopicName());
    assertEquals(expectedResult.get(1).getTopScore(), result.get(1).
getTopScore() );
    assertEquals(expectedResult.get(2).getTopicName(), result.get(2).
getTopicName());
    assertEquals(expectedResult.get(2).getTopScore(), result.get(2).
getTopScore() );
}
```