**QLC-2.3) – Limitations of Stubs.**

**Solution**

This is one solution for working with stubs to meet the requirements.

* [[“Physics”, [56, 67, 45, 89]], [“Art”, [87, 66, 78]], the result should be [[“Physics”, 89], [“Art”, 87]]

1. **Write your Test Highest score of many items (RED)**

Write a new test that finds the highest score with one topic using a stub.

* Name the test:

test\_find\_highest\_score\_with\_list\_of\_many\_returns\_list\_of\_many\_using\_stub

def test\_find\_highest\_score\_with\_list\_of\_many\_returns\_list\_of\_many\_using\_stub(self):

# Arrange

physics\_scores = [56, 67, 45, 89]

art\_scores = [87, 66, 78]

compsci\_scores = [45, 88, 97, 56]

topic\_scores = [

TopicScores("Physics", physics\_scores),

TopicScores("Art", art\_scores),

TopicScores("Comp Sci", compsci\_scores)

]

expected\_result = [

TopicTopScore("Physics", 89),

TopicTopScore("Art", 89),

TopicTopScore("Comp Sci", 89)

]

cut = TopicManager(HighestNumberFinderStub())

# Act

result = cut.find\_topic\_high\_scores(topic\_scores)

# Assert

for i in range(len(expected\_result)):

self.assertEqual(result[i].get\_topic\_name(), expected\_result[i].get\_topic\_name())

self.assertEqual(result[i].get\_top\_score(), expected\_result[i].get\_top\_score())

1. **Write minimal Production Code (GREEN)**

Modify the topic\_manager.py file to add a for loop to iterate through the many topics and their scores (topic\_scores\_list):

* Ensure test passes
* Commit code to Git

from app.topic\_top\_score import TopicTopScore

from final.app.highest\_number\_finder import HighestNumberFinder

class TopicManager:

def \_\_init\_\_(self, highest\_number\_finder=None):

if highest\_number\_finder is None:

highest\_number\_finder = HighestNumberFinder()

self.\_highest\_number\_finder = highest\_number\_finder

def find\_topic\_high\_scores(self, topic\_scores\_list):

top\_scores = []

for ts in topic\_scores\_list:

top\_score = self.\_highest\_number\_finder.find\_highest\_number(ts.get\_scores())

top\_scores.append(TopicTopScore(ts.get\_topic\_name(), top\_score))

return top\_scores

1. **Refactor Code - Optional.**

No refactoring required at this stage.

* Re-Run ALL tests to confirm no Regression.
* Commit code to Git, if not done in previous step.