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())
```

2. **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
```

3. **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.