**Project Report**

**NOTE: To see my edits refer to my branch -> castro/tests**

**Unit Testing with Sufficient Coverage**

**Player File**

* <https://github.com/csun-comp587-s20/Basketball-Statistics-Tracking/blob/castro/tests/src/test/java/basketball/PlayerTest.java> (File modified)
* Modified two methods in the main Player file to allow me to test those methods directly. (I placed comments on top of the methods in question. Essentially, I changed the methods from private to public)

**A screenshot of a cell phone

Description automatically generated**

* Original Player file for comparison: <https://github.com/csun-comp587-s20/Basketball-Statistics-Tracking/blob/master/src/Player.java>
* Tested functions with Jean Claude Hasrouty to get 89% coverage

A screenshot of a cell phone

Description automatically generated

* Sufficient coverage 100% across most functions. Some switch cases like in setStat were difficult to reach because of some string error. Was not able to solve that part.
* Ran a single test exception on line 159 of PlayersTest.java

A close up of a screen

Description automatically generated

**Undo File**

* <https://github.com/csun-comp587-s20/Basketball-Statistics-Tracking/blob/castro/tests/src/test/java/basketball/UndoTest.java> (File modified)
* Modified two methods in the main Undo file to allow me to test those methods directly. (I placed comments on top of the methods in question. Essentially, I changed the methods from private to public)

**A screenshot of text

Description automatically generated**

* Original Undo file for comparison: <https://github.com/csun-comp587-s20/Basketball-Statistics-Tracking/blob/master/src/Undo.java>
* Wrote multiple tests for each function to achieve 93% coverage

**A screenshot of a cell phone

Description automatically generated**

* Sufficient coverage 100% across most functions. An if else case in compareTo(Undo) were difficult to reach due to how the string is taken into the comparePeriods method.

**Automated Testing**

**GameSettings File**

* Created an automated test (lines 17 and 47) in the following file:

<https://github.com/csun-comp587-s20/Basketball-Statistics-Tracking/blob/castro/tests/src/test/java/basketball/GameSettingsTest.java>

**A screenshot of text

Description automatically generated**

* Automated test worked with no problems and contributed to most of the overall/branch coverage.

**Player File**

* Did not run any automated tests as it was difficult to automate the specific methods given that the string cases are not consistent.

**Undo File**

* Did not run any automated tests as it was difficult to automate certain cases of the undo action since the process was not consistent.

**Lessons Learned**

I learned through this process on how important it is to find well written code that has clear comments and tells you how methods work in order to test them. If I were to do this whole thing again, I would look for a project base that has well written code that allows for testing to be done. I wouldn’t want to test a project that requires you to go into the base code and change the access modifier from private to public for example. Clear indications on how the method functions also allows for more sufficient testing to be written.