**GitHub Link:** https://github.com/Brandon-Seidman/TriangleHW2/tree/HW05

**Assignment Description:** The objective of this assignment is to apply the techniques from the lecture to static testing of your Triangles program. Specifically:

* You will run a static code analyzer on your code, e.g. Pylint, identify and fix any problems reported by the static code analyzer;
* You will run a code coverage tool on your code, e.g. Coverage.py, and extend your test cases to demonstrate at least 80% code coverage;

**Author:** Brandon Seidman

**Summary:** I used pylint and coverage.py to analyze and update my triangle code to make sure that it conforms to standards and is testing correctly and effectively. At first when I ran both of these tests, I was seeing many issues in pylint but started out with an overall 96% coverage. I worked at updating everything within pylint and got it to the point where I received a 10 on the \_triangle.py and a 9.88 on the testtriangle.py. This also boosted uo my coverage score resulting in me receiving an overall 99% in coverage.

**Reflection:** I thought the assignment was very interesting and gave me a good understanding of QA tools used in the real world. I think coverage is great and will definitely be using that on many of my future projects, but I did have some trouble with pylint asking to fix things such as lines of code being too long or using camelCase rather than \_snake\_case. However, there were a lot of ways that it did prove itself very useful such as catching repeated or unnecessary code.

**Detailed Results:**

-Final Output:

Text

Description automatically generated

-Unedited Output:

Text

Description automatically generated

A screen shot of a computer

Description automatically generated

**Honor Pledge:** I pledge my honor that I have abided by the Stevens Honor System. -BS