Testing software is one of the most important aspects to managing and maintaining code. As code files grow, it can become impossible to keep track of all the changes and to make sure everything is running smoothly. This is where software testing comes into play. There is Manual testing which is labor intensive and extremely hard to scale and there is automated testing. The latter allows for large codebases to be maintained effectively throughout time.

One of the resources I found that offers a few different programs to test Python code was:

<https://www.fullstackpython.com/testing.html>

This resource included articles written as well as some links to various Git repositories that one can install and test some python code.

Here are links to the 3 main ones:

Green: <https://github.com/cleancut/green>

Requestium: <https://github.com/tryolabs/requestium>

Coverage.py: <https://coverage.readthedocs.io/en/coverage-5.4/>

It also mentions the difference between Unit Testing and Integration Testing. UT is when a single function of a program is tested, and IT is when testing of more than one function in application takes place at the same time. In this resource I also learned that Python is a heavily tested software due to the fact that it is dynamically typed and not a statically typed language. This means the developer must take extra precautions to maintain a proper codebase for users

Overall, I really found a lot of value within this resource as it provided a lot of links to a variety of things that can be of use to any developer. Testing code is incredibly important and I also learned just how much that is the case with Python in particular as well.