Lab 1: Python Start Documentation + Test Cases

Jade Pearl

SDEV 300 Building Secure Python Applications

Prof. Craig Poma

Pylint Score: Overall 10/10.0

The Pylint Score for my voter.py application started out as about a 7.5 because each of my functions needed a docuheader in them to comment on what the functions were. I also used the exit() function whereas Pylint suggested import sys and use the sys.exit() statement instead as it is a more desirable use of that function. My naming conventions of the functions did not at first have an “\_” character so I also changed the names to fit better naming conventions. After the proper reformatting of the program, I ended with a 10/10.0 pylint score with my lab still functioning properly.

Test Case Table

|  |  |  |
| --- | --- | --- |
| Test Case (Expectation) | Input/Output as shown on Terminal Console | Pass/Fail |
| 1: Ask the user the required information |  | Pass: user is asked all questions given that they are eligible |
| 2a: User must be 18 or older to register |  | Pass |
| 2b. User must be a U.S. Citizen |  | Pass |
| 2c. User must enter a valid State’s initials |  | Pass |
| 3a. Logic check (age < 120) |  | Pass |
| 3b. Logic check (State must be 2 characters) |  | Pass |
| 4. Allow the user to stay or leave in the application between each question. Validate by asking the user if they are sure they want to leave in case they enter anything but yes by accident. |  | Pass |
| 5. Input Validity: give the user the chance to reenter information if they give invalid data.  Names, citizenship, and state have to be a string while age and zip have to be an integer |  | Pass |