SW Testing / Quality Assurance

Assignment 2 Report

Nolan Royse

GitHub Username - NolanRoyse

Setup and execution-

Programming Language: Python - https://www.python.org/

Version of Python Used: 3.8.1

Operating system Used: Ubuntu Linux - https://ubuntu.com/

Version of Ubuntu Used: 19.10

Programming Environment Used: Sublime Text Editor - https://www.sublimetext.com/

Program Execution Environment Used- Terminal in Ubuntu

Repository Program Used: GitHub- https://github.com/

***Repository for this Project: https://github.com/NolanRoyse/SW-Testing-QA-Homework-2 ***

Execution instructions:

1. Open a terminal application
2. Navigate to location of file in OS using the “cd” command
3. type “python *file_name*.py” and then hit enter (see a screenshot later in report for exact formatting)
4. Run through program

Program Description-

Within this program there are three main functions for the program. These functions are the main, BMI Calculator, and the Retirement Calculator. There will be a series of tests for each of these functions to ensure that they are working correctly. The main function is essentially the driver for the rest of the function. It directs the program to other functions when the input is correct. The BMI calculators main function is calculating the BMI when given input. The retirement calculators main function is calculating the retirement age that a person could based on input.

Test Cases for Main Function -

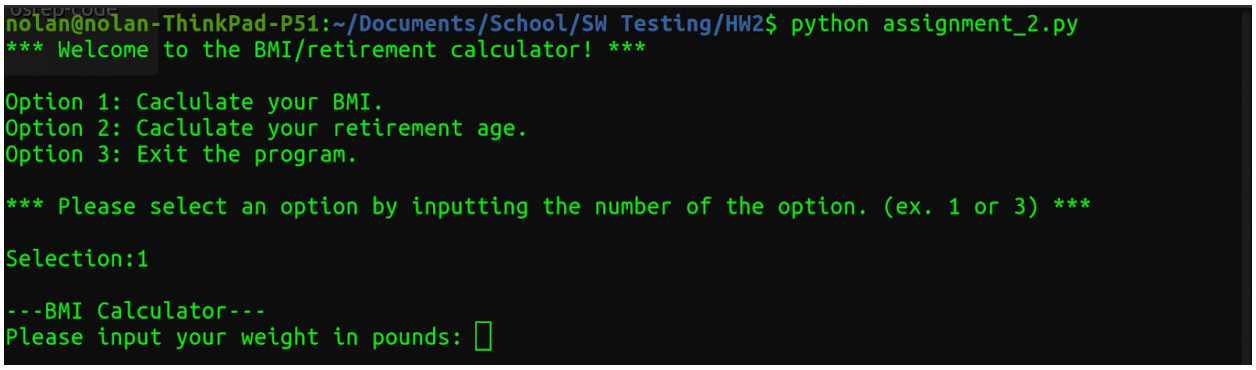
All that the main function is responsible for is taking an input and going to a certain function based on that input. The inputs are either 1 to go to BMI calculator, 2 to go to retirement calculator, or 3 to exit the program. These inputs will be tested, as well as testing incorrect values to ensure that the error validation is working correctly.

Test 1.1-
Test Case- Input of 1 successfully takes the user to the BMI Calculator function

Input- Integer: 1

Pass or Fail- Pass

Screenshot of Process-

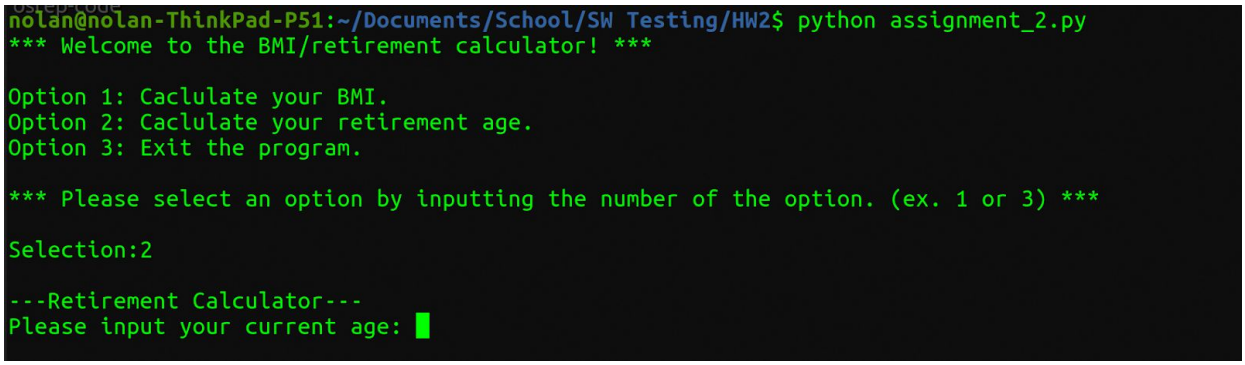


Test 1.2-
Test Case - Input of 2 successfully takes the user to the Retirement Calculator

Input- Integer: 2

Pass or Fail- Pass

Screenshot of Process-

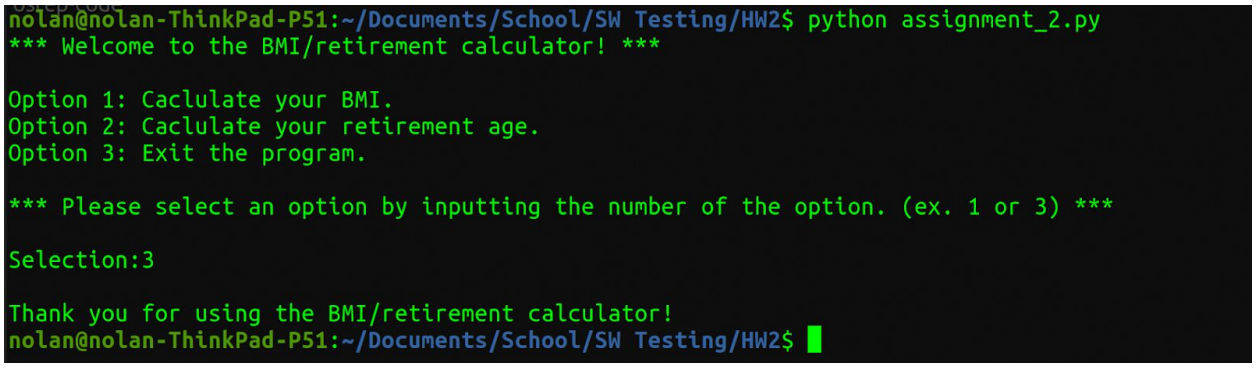


Test 1.3 -
Test Case- Input of 3 successfully closes the program

Input- Integer: 3

Pass or Fail- Pass

Screenshot of Process-

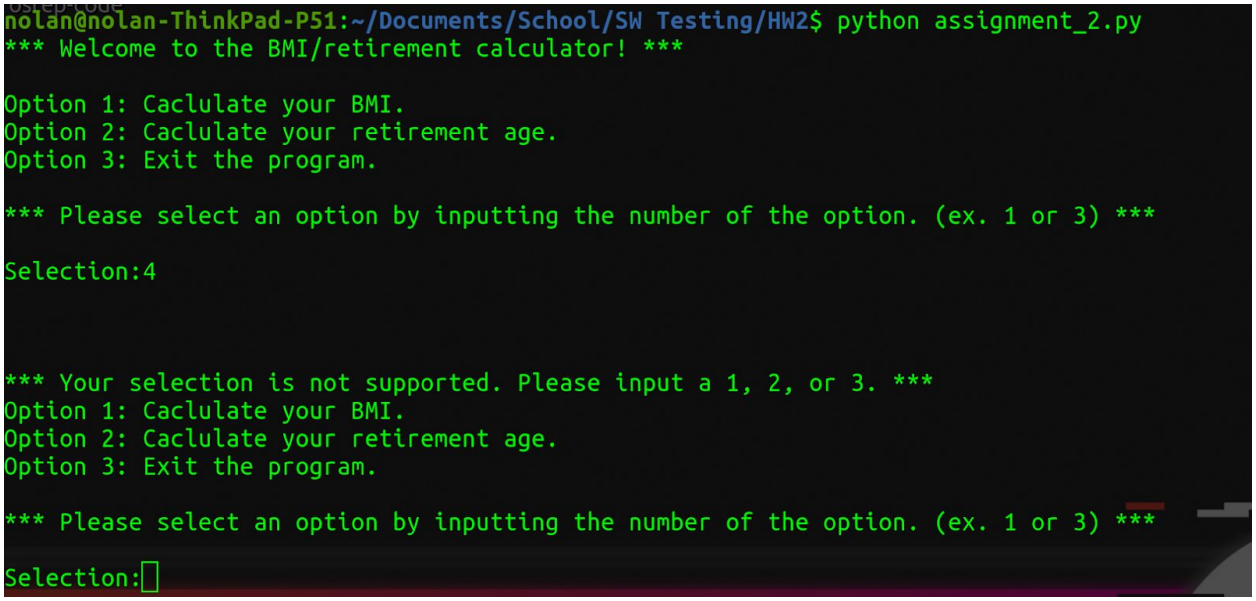


Test 1.4-
Test Case- Input of a integer that is not 1, 2, or 3 will prompt user to enter a new, correct solution

Input- Integer: 4

Pass or Fail- Pass

Screenshot of Process -



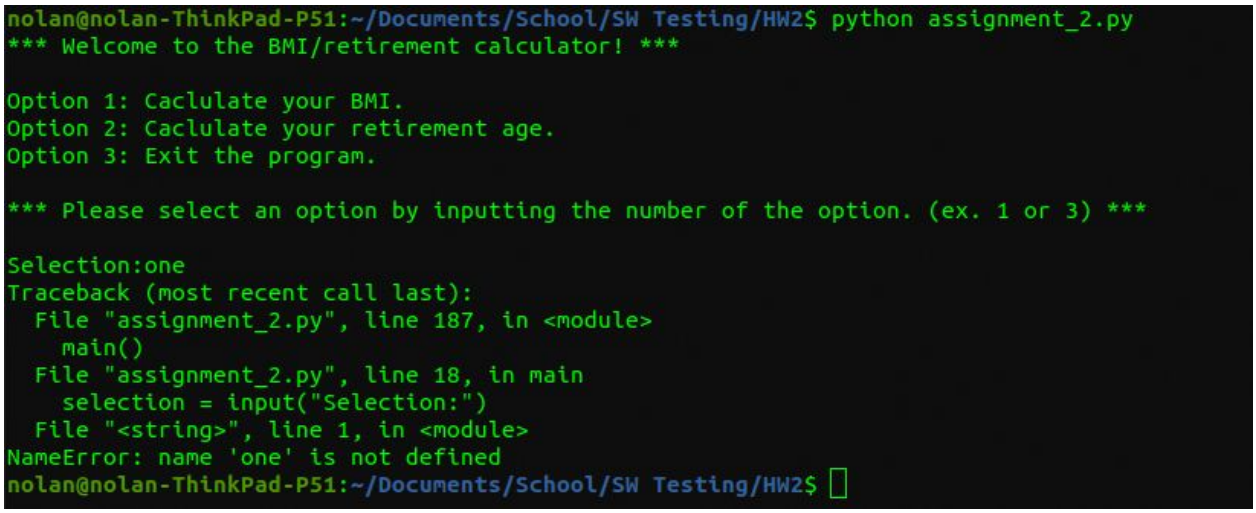
Test 1.5-

Test Case- Input of a string to test the error validation

Input- String: one

Pass or Fail - Fail

Screenshot of Process-



Test 1 Summary-

Altough there was not much testing to be done on the main function, these tests were necessary for allowing the user to be able to access the other functions. There were enough tests to ensure that most bases were covered and that all inputs that a user might put. Overall most of the tests were successful, aside from test 1.5.

Test for BMI Calaculator-

The tests for the BMI calculator are going to test the function of the calculator to produce the correct results and if the error validation is correct. There are also going to be boundary tests for the set boundary values in the function. There are certain boundary values set because there are values that are realistic within the confines of the calculator.

Boundary Values for BMI Calculator-

Weight: Minimum Value - 5lbs. This was set as the minimum value because most humans even at birth are going to wiegh in excess of 5lbs.

Weight: Maximum Value - 500bls. This was set as the maximum weight because most of the general public is under this weight and the calculations most of the time do not have to excees this.

Height: Minimum Value - 5in. This was set as the minimum value because most humans are taller than 5 inches, but there still needs to be room for exceptions

Height Maximum Value - 100in. This was set as the maximum value because most of the general public is below 100in. And there is not much need to exceed this by much further.

Test 2.1-

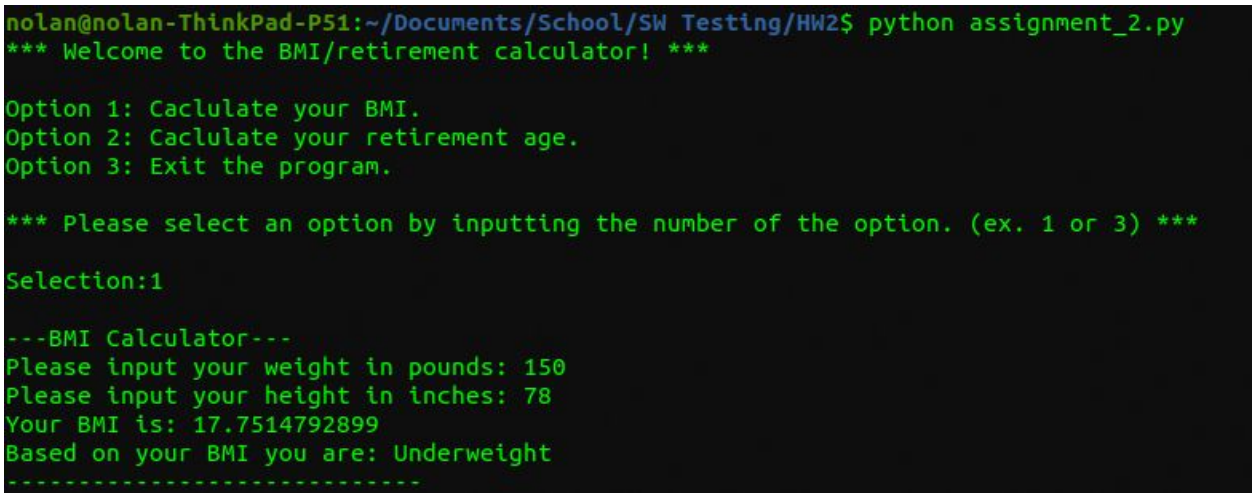
Test Case- successful inputs that lead to correct output of BMI type

Input- First Selection- Integer: 1 (to get to BMI Calculator)
Weight- Integer: 150
Height- Integer: 78

Desired Output- Underweight

Pass or Fail- Pass

Screenshot of Process-



Test 2.2-

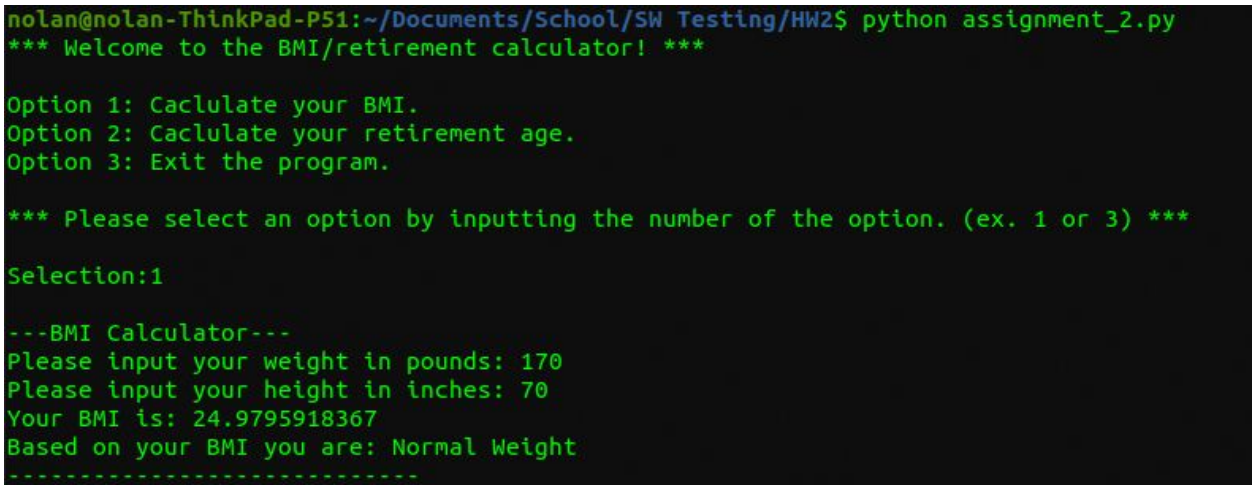
Test Case- successful inputs that lead to correct output of BMI type

Input- First Selection- Integer: 1 (to get to BMI Calculator)
Weight- Integer: 170
Height- Integer: 70

Desired Output- Normal Weight

Pass or Fail- Pass

Screenshot of Process-



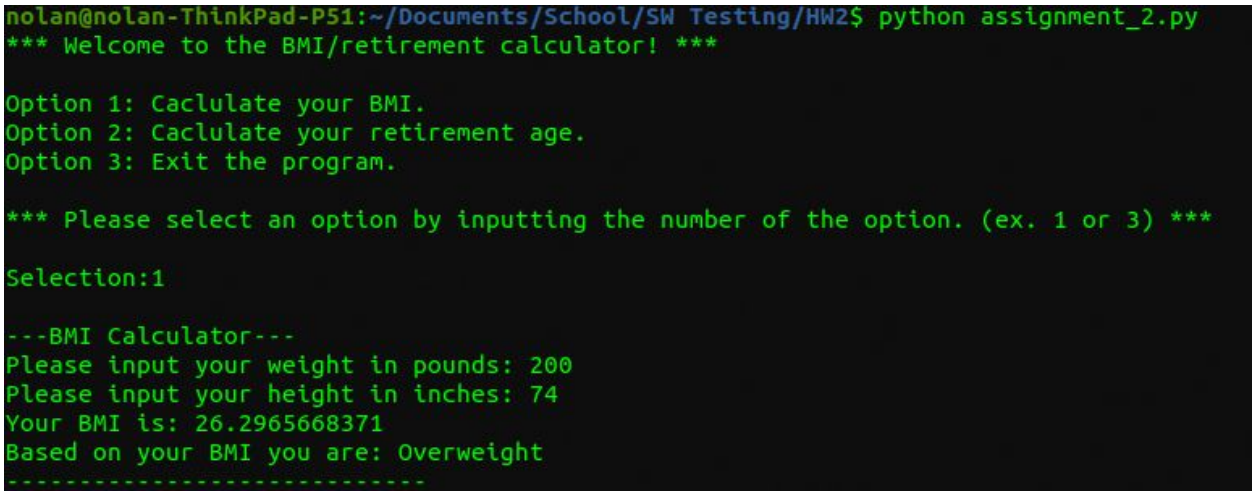
Test 2.3-
Test Case- successful inputs that lead to correct output of BMI type

Input- First Selection- Integer: 1 (to get to BMI Calculator)
Weight- Integer: 200
Height- Integer: 74

Desired Output- Overweight

Pass or Fail - Pass

Screenshot of Process-



Test 2.4-

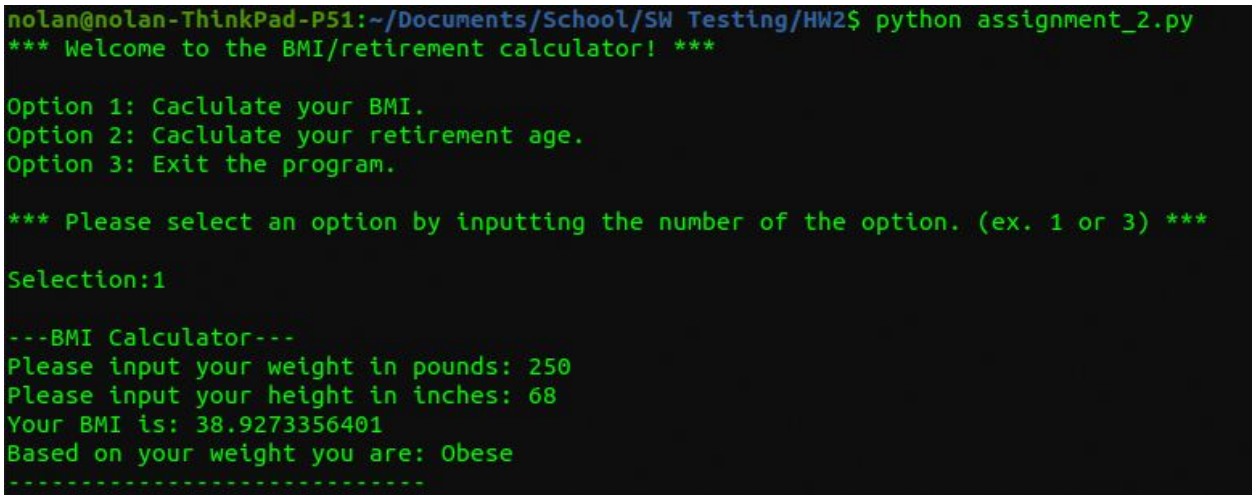
Test Case- successful inputs that lead to correct output of BMI type

Input- First Selection- Integer: 1 (to get to BMI Calculator)
Weight- Integer: 250
Height- Integer: 68

Desired Output- Obese

Pass or Fail - Pass

Screenshot of Process-



Test 2.5-

Test Case- incorrect inputs to perform error validation and boundary value checks on the weight

Input- First Selection- Integer: 1 (to get to BMI Calculator)

1. Weight- Integer: 4
2. Weight- Integer: 5
3. Weight- Integer: 6
4. Weight- Integer: 499
5. Weight- Integer: 500
6. Weight- Integer: 501

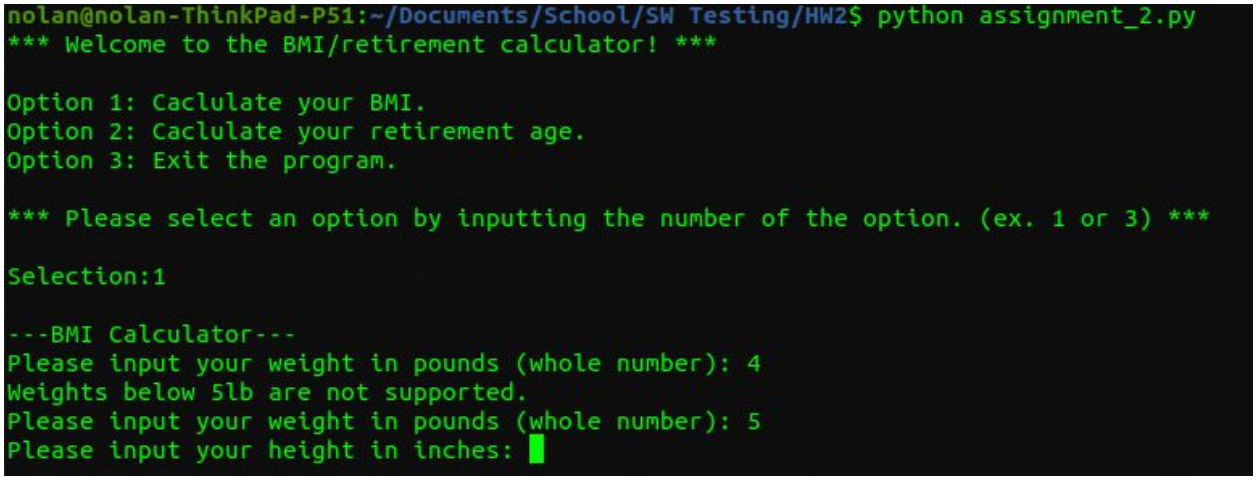
Desired Output- For 1 & 6- values not supported

For 2, 3, 4, & 5 - Values supported/ next prompt

Pass or Fail - Pass

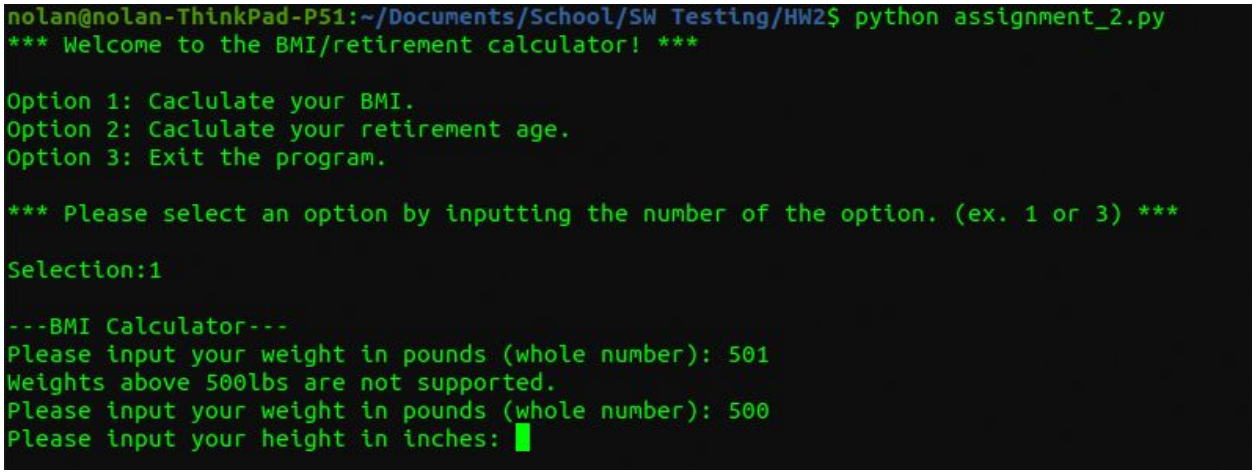
Screenshots of Process-

Testing inputs 1 and 2

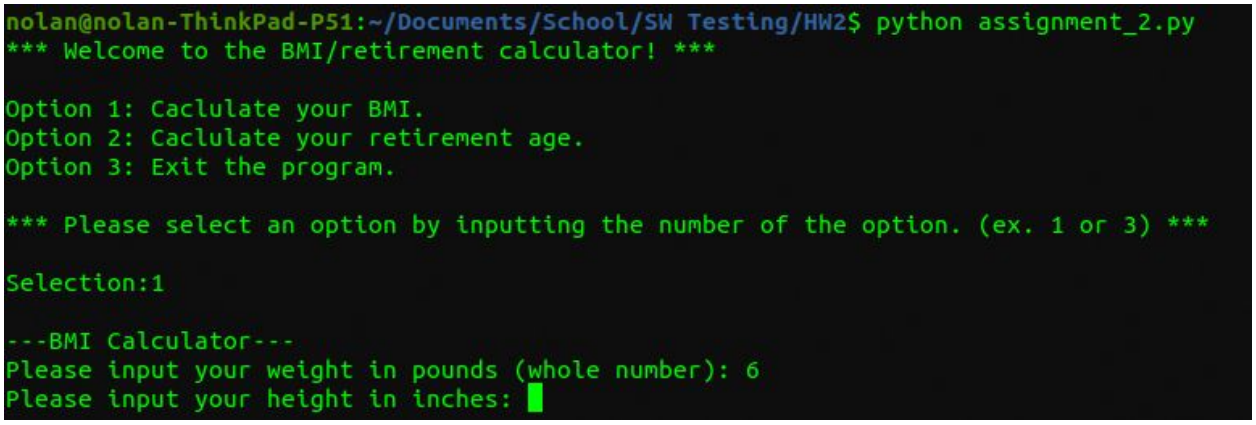


Test 2.5 cont. -

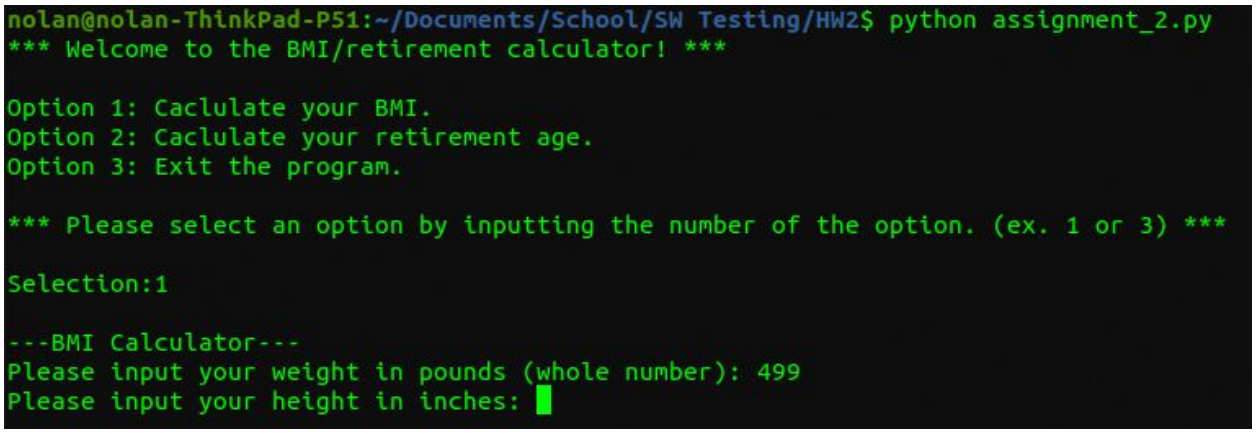
Testing inputs 5 & 6



Testing input 3



Testing input 4



Test 2.6-

Test Case- incorrect inputs to perform error validation and boundary value checks on the height

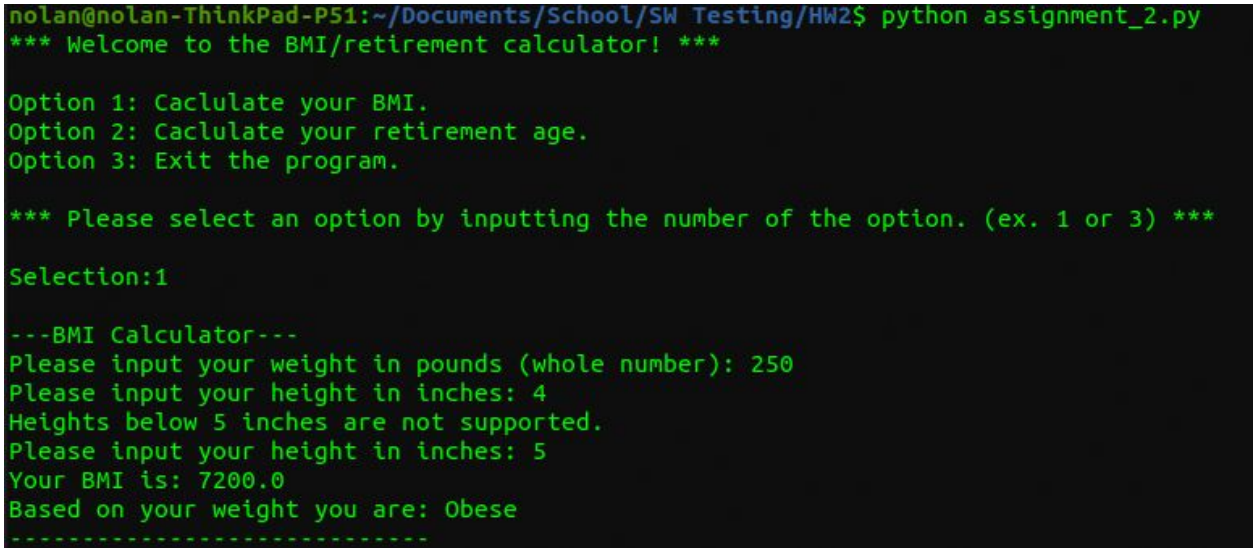
Input- First Selection- Integer: 1 (to get to BMI Calculator)
Weight- Integer: 250
1. Height- Integer: 4
2. Height- Integer: 5
3. Height- Integer: 6
4. Height- Integer: 99
` 5. Height- Integer: 100
6. Height- Integer: 101

Desired Output- For 1 & 6- values not supported
For 2, 3, 4, & 5 - Values supported/ next prompt

Pass or Fail - Pass

Screenshots of Process-

Testing inputs 1 & 2-



Test 2.6 cont-

Testing inputs 4 & 5-

```
nolan@nolan-ThinkPad-P51:~/Documents/School/SW Testing/HW2$ python assignment_2.py
*** Welcome to the BMI/retirement calculator! ***

Option 1: Caclulate your BMI.
Option 2: Caclulate your retirement age.
Option 3: Exit the program.

*** Please select an option by inputting the number of the option. (ex. 1 or 3) ***

Selection:1

---BMI Calculator---
Please input your weight in pounds (whole number): 250
Please input your height in inches: 101
Heights above 100 inches are not supported.
Please input your height in inches: 100
Your BMI is: 18.0
Based on your BMI you are: Underweight
----------------------------
```

Testing input 3

```
nolan@nolan-ThinkPad-P51:~/Documents/School/SW Testing/HW2$ python assignment_2.py
*** Welcome to the BMI/retirement calculator! ***

Option 1: Caclulate your BMI.
Option 2: Caclulate your retirement age.
Option 3: Exit the program.

*** Please select an option by inputting the number of the option. (ex. 1 or 3) ***

Selection:1

---BMI Calculator---
Please input your weight in pounds (whole number): 250
Please input your height in inches: 6
Your BMI is: 5000.0
Based on your weight you are: Obese
--------------------------------
```

Testing input 6

```
nolan@nolan-ThinkPad-P51:~/Documents/School/SW Testing/HW2$ python assignment_2.py
*** Welcome to the BMI/retirement calculator! ***

Option 1: Caclulate your BMI.
Option 2: Caclulate your retirement age.
Option 3: Exit the program.

*** Please select an option by inputting the number of the option. (ex. 1 or 3) ***

Selection:1

---BMI Calculator---
Please input your weight in pounds (whole number): 250
Please input your height in inches: 99
Your BMI is: 18.3654729109
Based on your BMI you are: Underweight
----------------------------
```

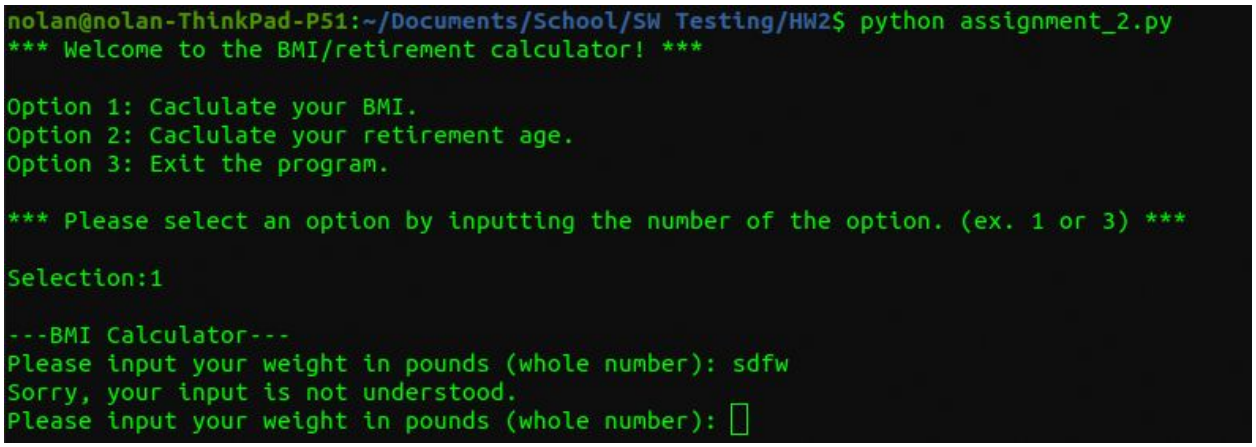
Test 2.7-
Test Case- incorrect inputs that check the error validation of weight input

Input- First Selection- Integer: 1 (to get to BMI Calculator)
Weight- Integer: sdfw

Desired Output- Incorrect input/ prompt to re-enter

Pass or Fail - Pass

Screenshot of Process-



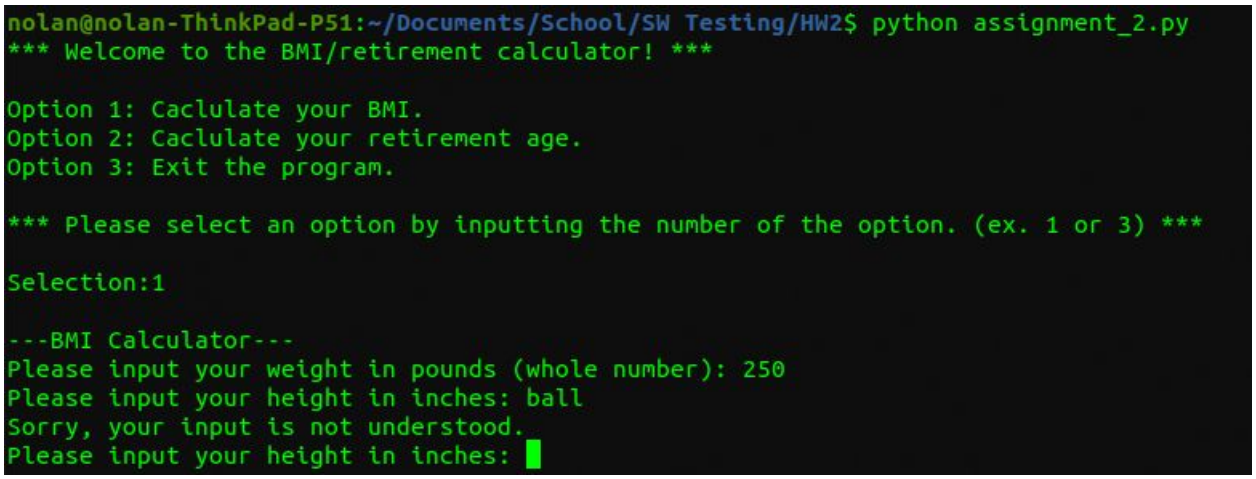
Test 2.8-
Test Case- incorrect inputs that check the error validation of height input

Input- First Selection- Integer: 1 (to get to BMI Calculator)
Weight- Integer: 250
Height- String: ball

Desired Output- Incorrect input/ prompt to re-enter

Pass or Fail - Pass

Screenshot of Process-



BMI testing summary-

All of the testing done on the BMI calculator are to determine if the output is correct, boundary values are operating correctly, and error validation is working correctly. This was proven to be successful.

Tests for Retirement Calculator-

The tests for the retirement calculator are going to test the function of the calculator to produce the correct results and if the error validation is correct. There are also going to be boundary tests for the set boundary values in the function. There are certain boundary values set because there are values that are realistic within the confines of the calculator.

Boundary Values for Retirement Calculator-

Age: Minimum Value - 18yo. This was set as the minimum value because most humans are not going to have a job or start saving until they are 18 years old.

Age: Maximum Value - 100yo. This was set as the maximum value for the saving because most people are going to be already saved by this point, but it was just for padding for exceptions.

Salary: Minimum Value - $10,000. This was set as the minimum value because most humans have more than a $10,000 annual salary.

Salary: Maximum Value - $15,000,000. This was set as the maximum value for salary because most people make much below this, but there are some exceptions.

Goal: Minimum Value - $100,000. This was set as the minimum value because most humans want to save at least $100,000 for their retirement money.

Goal: Maximum Value - $20,000,000. This was set as the maximum value because most of the general public will be below this, but there are some that would like to save to this point if possible.

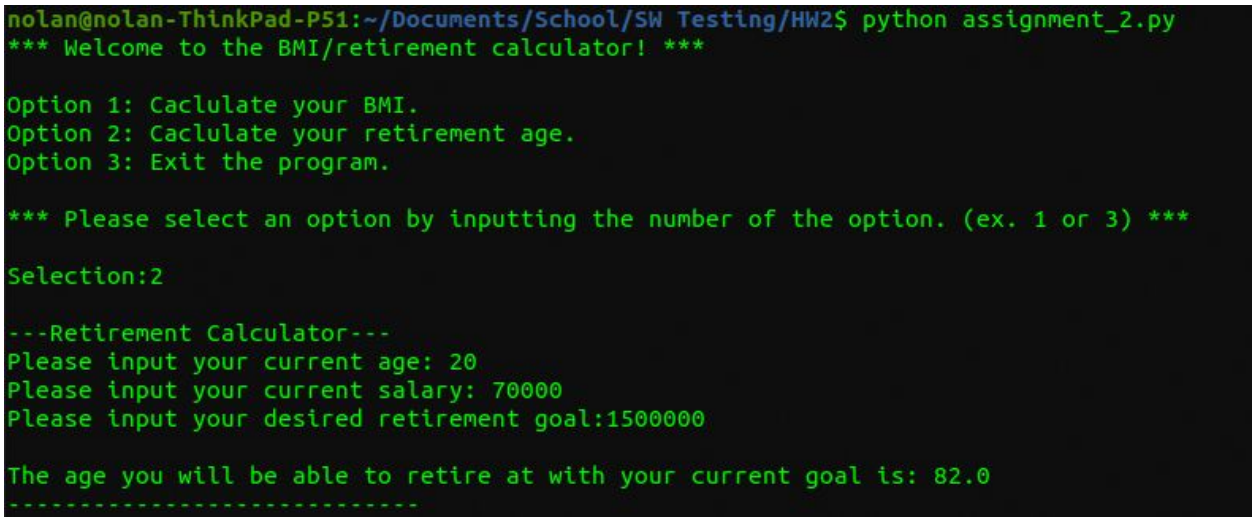
Test 3.1-
Test Case- correct inputs that result in an age of retirement under 100

Input- First Selection- Integer: 2 (to get to retirement Calculator)
Age- Integer: 20
Salary- Integer: 70,000
Goal- Integer: 1,500,000

Desired Output- Able to retire before the age of 100/ age: 82

Pass or Fail - Pass

Screenshot of Process-



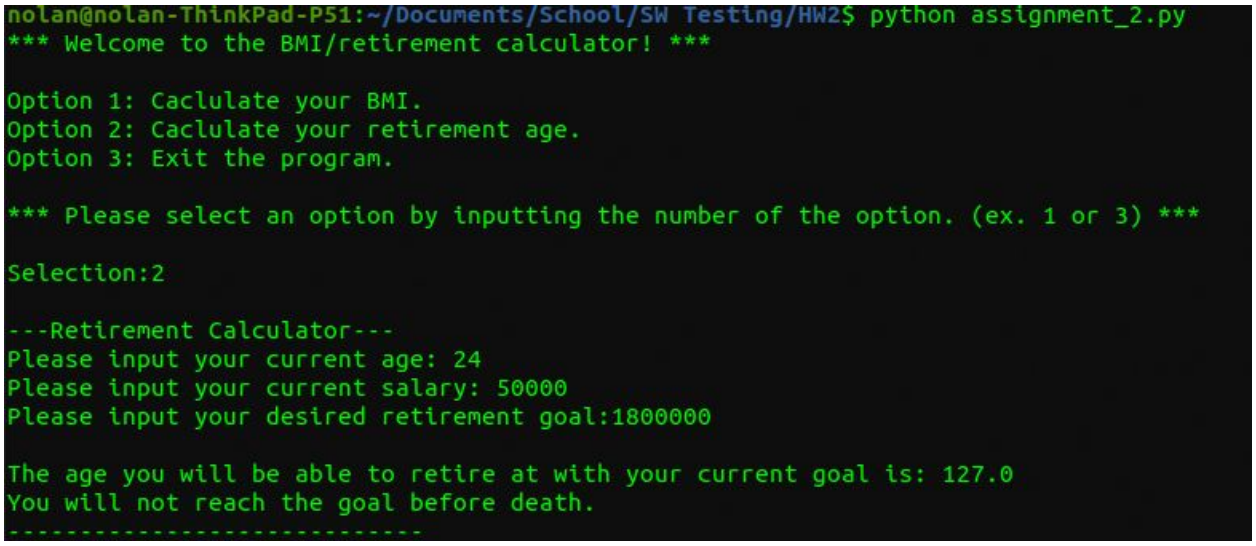
Test 3.2-
Test Case- correct inputs that result in an age of retirement under 100

Input- First Selection- Integer: 2 (to get to retirement Calculator)
Age- Integer: 24
Salary- Integer: 50,000
Goal- Integer: 1,800,000

Desired Output- unable to retire before the age of 100/ age: 127

Pass or Fail - Pass

Screenshot of Process-



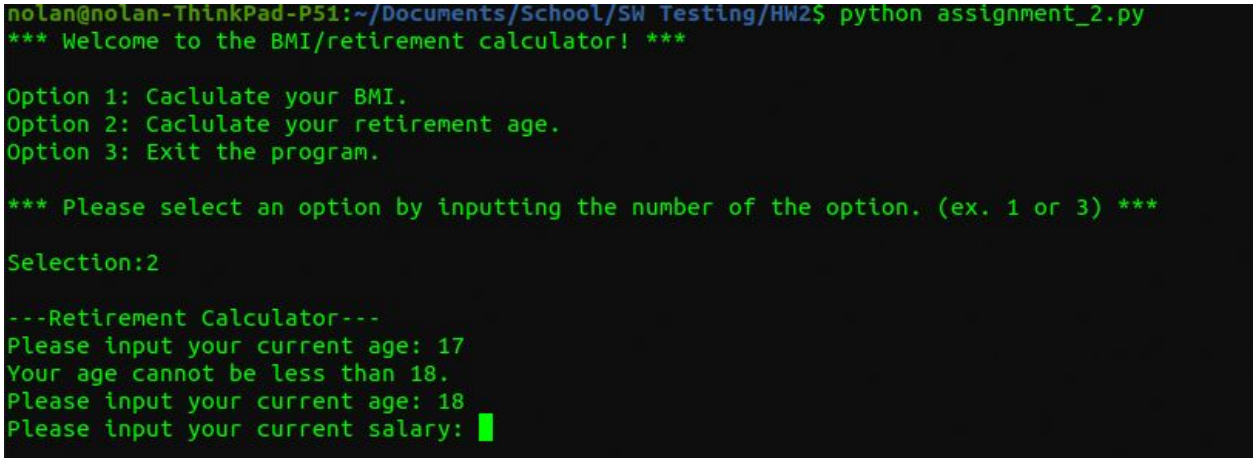
Test 3.3-
Test Case- boundary testing for the age input

Input- First Selection- Integer: 2 (to get to retirement Calculator)
1.Age- Integer: 17
2.Age- Integer: 18
3.Age- Integer: 19
4.Age- Integer: 99
5.Age- Integer: 100
6.Age- Integer: 101

Desired Output- For 1 & 6- values not supported
For 2, 3, 4, & 5 - Values supported/ next prompt

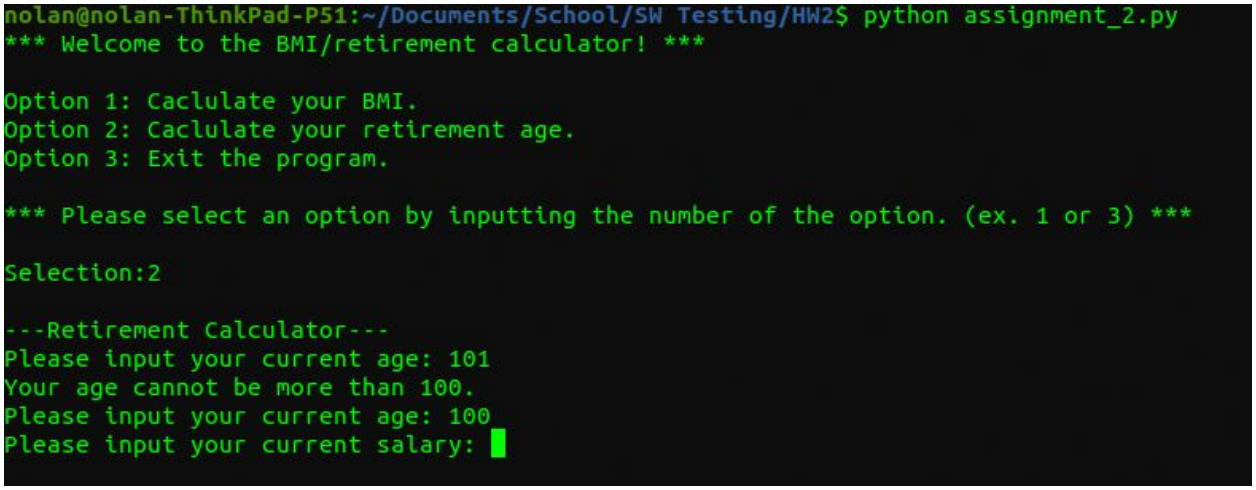
Pass or Fail - Pass

Screenshot of Process-
Testing inputs 1 & 2-

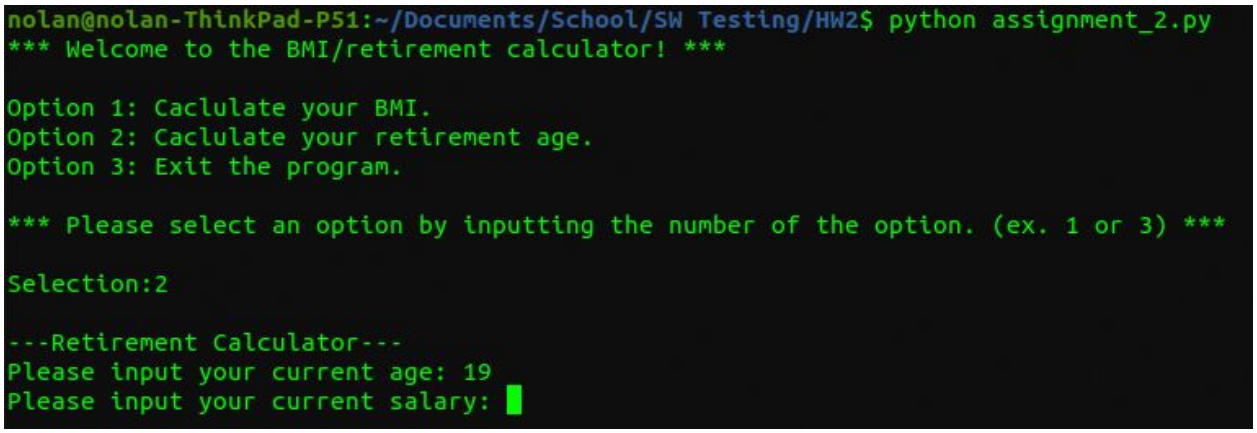


Test 3.3 cont-

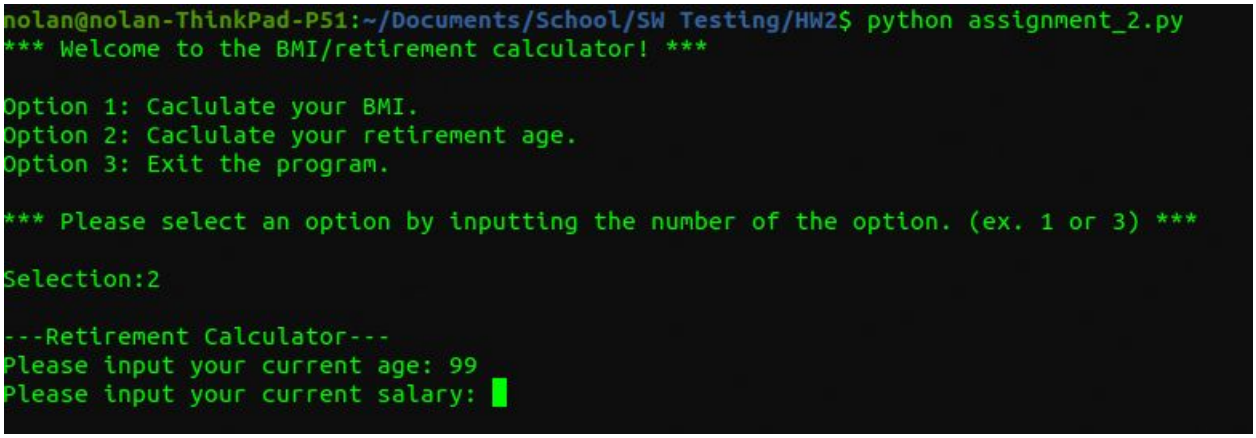
Testing inputs 5 & 6-



Testing input 3-



Testing input 4-



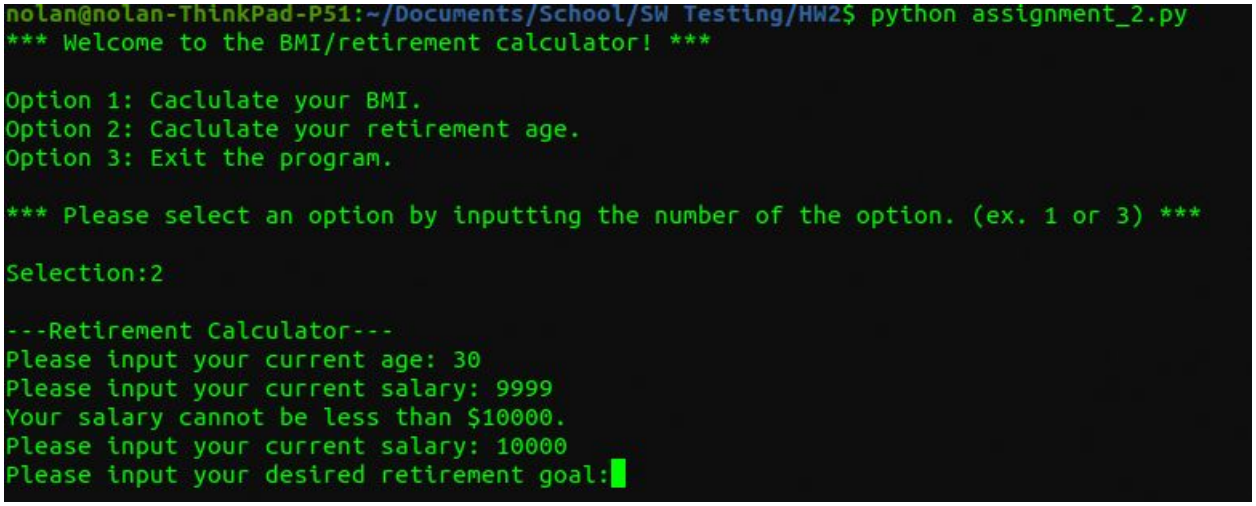
Test 3.4-
Test Case- boundary testing for the salary input

Input- First Selection- Integer: 2 (to get to retirement Calculator)
Age- Integer: 30
1.Salary- Integer: 9,999
2.Salary- Integer: 10,000
3.Salary- Integer: 10,001
4.Salary- Integer: 14,999,999
5.Salary- Integer: 15,000,000
6.Salary- Integer: 15,000,001

Desired Output- For 1 & 6- values not supported
For 2, 3, 4, & 5 - Values supported/ next prompt

Pass or Fail - Pass

Screenshot of Process-
Testing inputs 1 & 2-



Test 3.4 cont.-

Testing inputs 5 & 6-

```
nolan@nolan-ThinkPad-P51:~/Documents/School/SW Testing/HW2$ python assignment_2.py
*** Welcome to the BMI/retirement calculator! ***

Option 1: Caclulate your BMI.
Option 2: Caclulate your retirement age.
Option 3: Exit the program.

*** Please select an option by inputting the number of the option. (ex. 1 or 3) ***

Selection:2

---Retirement Calculator---
Please input your current age: 30
Please input your current salary: 15000001
Your salary cannot be more than $15,000,000.
Please input your current salary: 15000000
Please input your desired retirement goal:
```

Testing input 3-

```
nolan@nolan-ThinkPad-P51:~/Documents/School/SW Testing/HW2$ python assignment_2.py
*** Welcome to the BMI/retirement calculator! ***

Option 1: Caclulate your BMI.
Option 2: Caclulate your retirement age.
Option 3: Exit the program.

*** Please select an option by inputting the number of the option. (ex. 1 or 3) ***

Selection:2

---Retirement Calculator---
Please input your current age: 30
Please input your current salary: 10001
Please input your desired retirement goal:
```

Testing input 4-

```
nolan@nolan-ThinkPad-P51:~/Documents/School/SW Testing/HW2$ python assignment_2.py
*** Welcome to the BMI/retirement calculator! ***

Option 1: Caclulate your BMI.
Option 2: Caclulate your retirement age.
Option 3: Exit the program.

*** Please select an option by inputting the number of the option. (ex. 1 or 3) ***

Selection:2

---Retirement Calculator---
Please input your current age: 30
Please input your current salary: 14999999
Please input your desired retirement goal:
```

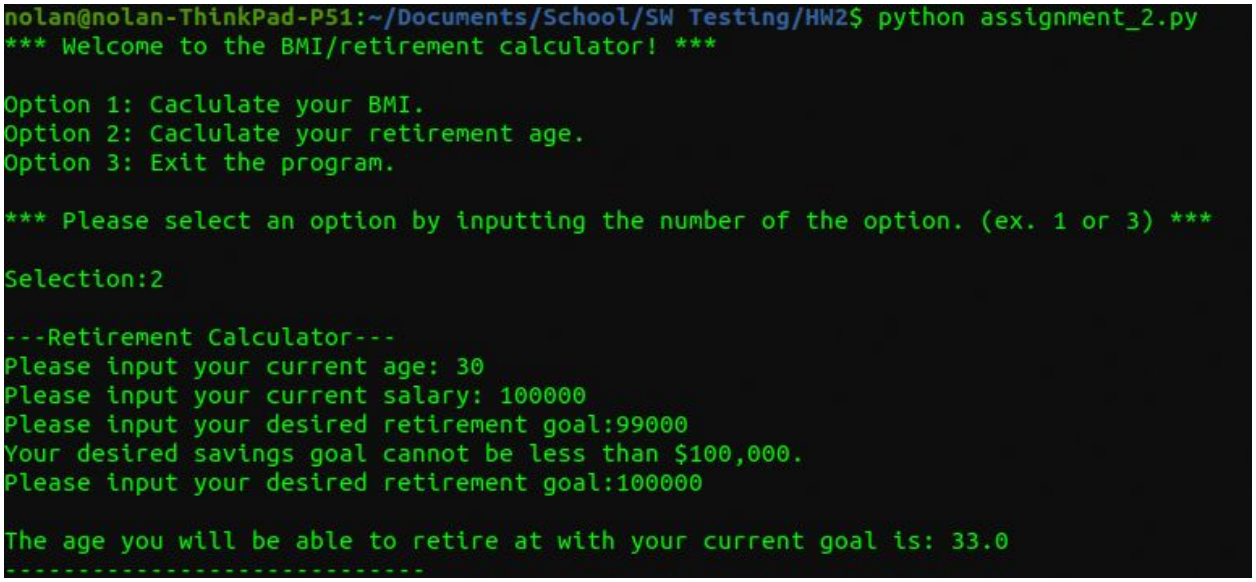
Test 3.5-
Test Case- boundary testing for the goal input

Input- First Selection- Integer: 2 (to get to retirement Calculator)
Age- Integer: 30
Salary- Integer: 10000
1.Goal- Integer: 99,000
2.Goal- Integer: 100,000
3.Goal- Integer: 100,001
4.Goal- Integer: 19,999,999
5.Goal- Integer: 20,000,000
6.Goal- Integer: 20,000,001

Desired Output- For 1 & 6- values not supported
For 2, 3, 4, & 5 - Values supported/ next prompt

Pass or Fail - Pass

Screenshot of Process-
Testing inputs 1 & 2-



Test 3.5 cont-

Testing inputs 5 & 6-

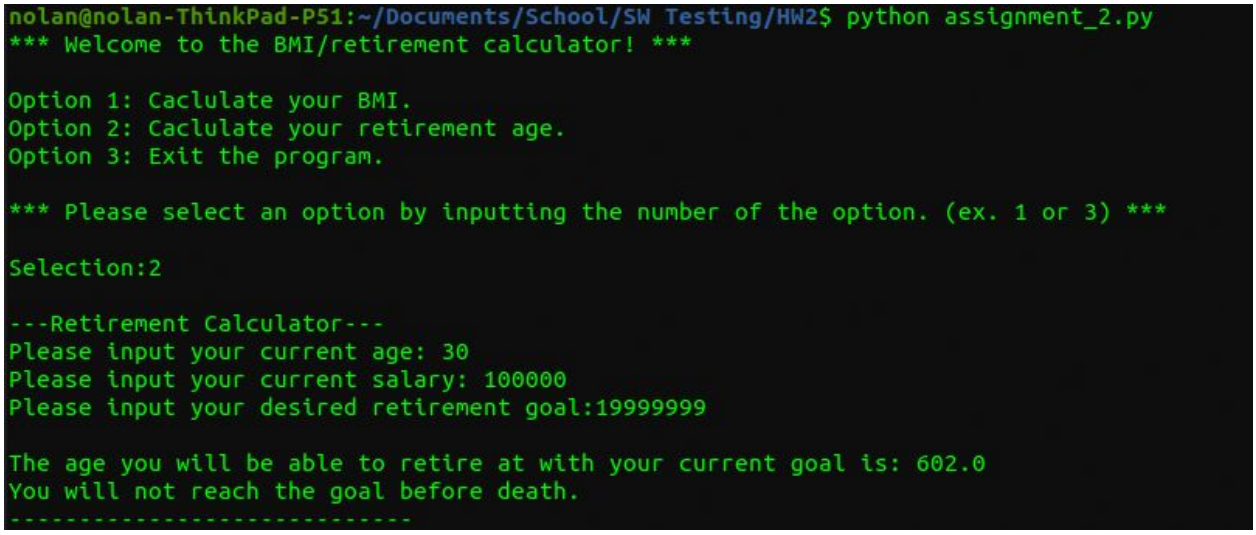
```
nolan@nolan-ThinkPad-P51:~/Documents/School/SW Testing/HW2$ python assignment_2.py
*** Welcome to the BMI/retirement calculator! ***

Option 1: Caclulate your BMI.
Option 2: Caclulate your retirement age.
Option 3: Exit the program.

*** Please select an option by inputting the number of the option. (ex. 1 or 3) ***

Selection:2

---Retirement Calculator---
Please input your current age: 30
Please input your current salary: 100000
Please input your desired retirement goal:20000001
Your desired savings goal cannot be more than $20,000,000.
Please input your desired retirement goal:20000000

The age you will be able to retire at with your current goal is: 602.0
You will not reach the goal before death.
----------------------------
```

Testing input 3-

```
nolan@nolan-ThinkPad-P51:~/Documents/School/SW Testing/HW2$ python assignment_2.py
*** Welcome to the BMI/retirement calculator! ***

Option 1: Caclulate your BMI.
Option 2: Caclulate your retirement age.
Option 3: Exit the program.

*** Please select an option by inputting the number of the option. (ex. 1 or 3) ***

Selection:2

---Retirement Calculator---
Please input your current age: 30
Please input your current salary: 100000
Please input your desired retirement goal:100001

The age you will be able to retire at with your current goal is: 33.0
----------------------------
```

Test 3.5 cont-

Testing input 4-



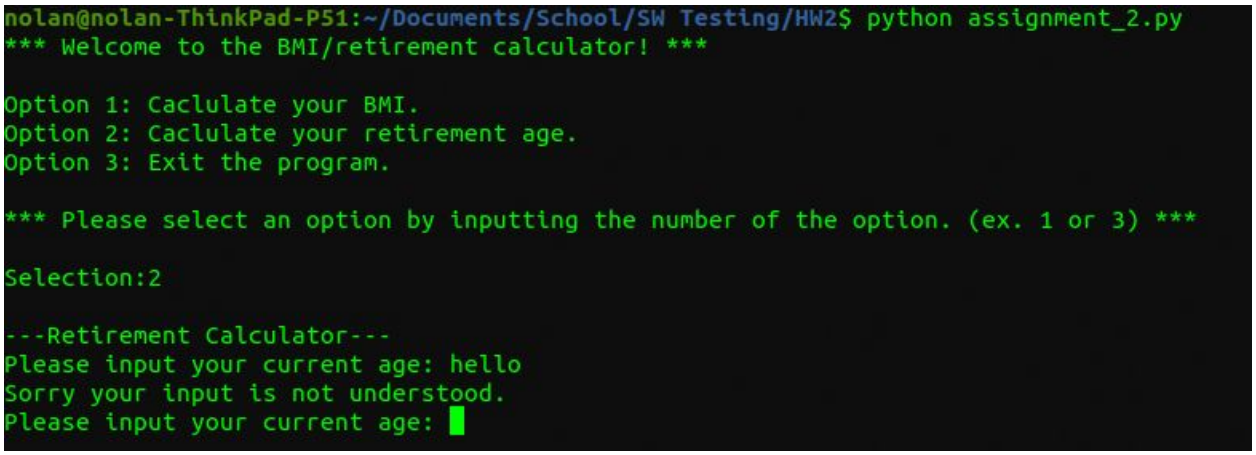
Test 3.6-
Test Case- incorrect inputs that check the error validation of age input

Input- First Selection- Integer: 2 (to get to retirement Calculator)
Age- string: hello

Desired Output- Incorrect input/ prompt to re-enter

Pass or Fail - Pass

Screenshot of Process-



Test 3.7-

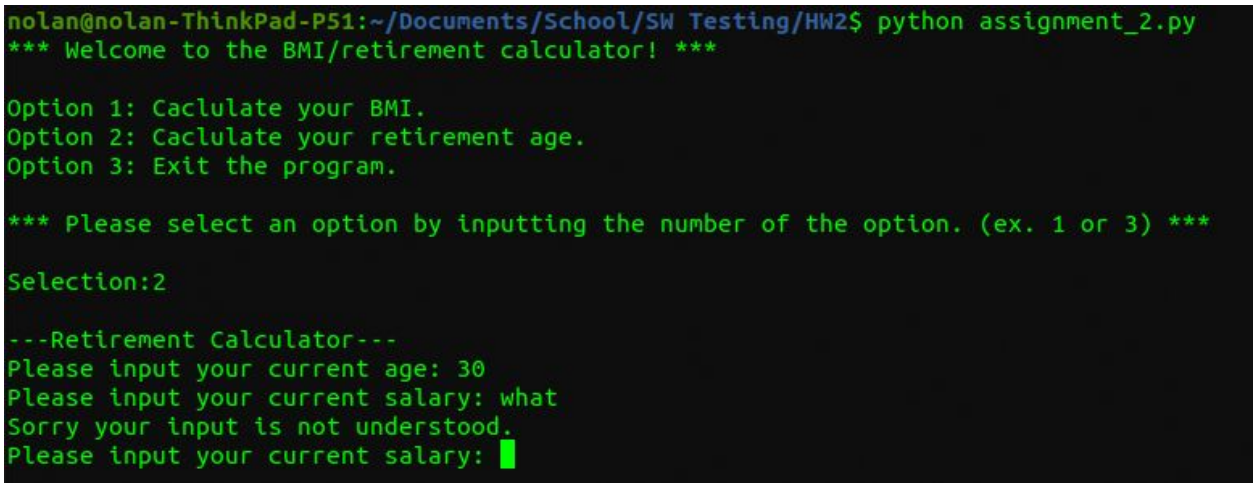
Test Case- incorrect inputs that check the error validation of salary input

Input- First Selection- Integer: 2 (to get to retirement Calculator)
Age- Integer: 30
Salary- String: what

Desired Output- Incorrect input/ prompt to re-enter

Pass or Fail - Pass

Screenshot of Process-



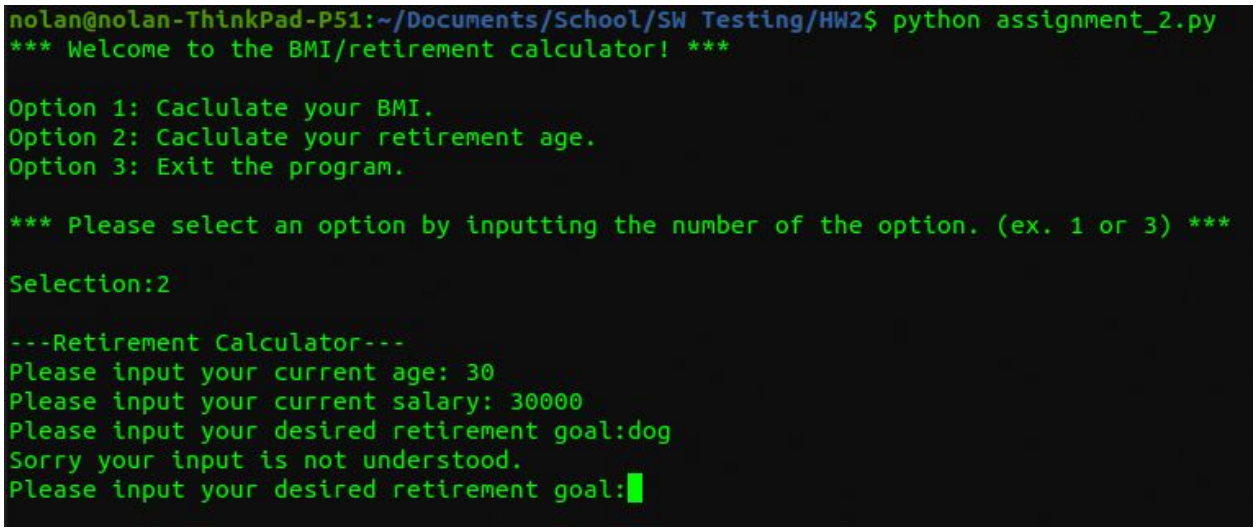
Test 3.8-
Test Case- incorrect inputs that check the error validation of desired goal input

Input- First Selection- Integer: 2 (to get to retirement Calculator)
Age- Integer: 30
Salary- Integer: 30,000
Goal- String: dog

Desired Output- Incorrect input/ prompt to re-enter

Pass or Fail - Pass

Screenshot of Process-



Retirement testing summary-

All of the testing done on the retirement calculator are to determine if the output is correct, boundary values are operating correctly, and error validation is working correctly. This was proven to be successful.