Name: Nihar Shah
Stevens Institute of Technology
CWID: 20012583

HW05: Static Code Analysis

Course: SSW – 567

Summary:
The changes made to the original program after running the code analyzer and adding one more comment to the program made the code more readable and executable eradicating all the unnecessary indentation, spaces, variable renaming. Once this was done static value reached to a full. Hence ensuring in covering 81% coverage.

---

1. The GitHub URL of this code which is analyzed is:
   ➔ https://github.com/NehharShah/HW05-Static-Code-Analysis

---

2. The name and output of the static code analyzer tool you used:
   ➔ The tool used for static code analyzer is **Pylint** and **Coverage**

Initial Output (Before making the changes to code)

```
(base) niharshah@Nihars-MacBook-Pro HW01 % pylint triangle.py
************* Module triangle
triangle.py:8:3: C0303: Trailing whitespace (trailing-whitespace)
triangle.py:8:3: C0303: Trailing whitespace (trailing-whitespace)
triangle.py:19:0: C0301: Line too long (108/100) (line-too-long)
triangle.py:25:0: C0305: Trailing newlines (trailing-newlines)
triangle.py:10:0: C0103: Function name "classifyTriangle" doesn't conform to snake_case naming style (invalid-name)
triangle.py:10:0: C0103: Argument name "a" doesn't conform to snake_case naming style (invalid-name)
triangle.py:10:0: C0103: Argument name "b" doesn't conform to snake_case naming style (invalid-name)
triangle.py:10:0: C0103: Argument name "c" doesn't conform to snake_case naming style (invalid-name)
triangle.py:10:0: C0116: Missing function or method docstring (missing-function-docstring)
triangle.py:17:4: R1705: Unnecessary "elif" after "return" (no-else-return)

-----------------------------------
Your code has been rated at 1.67/10
```

Pylint Report:

```
Report
======
12 statements analysed.

Statistics by type
------------------

+---------+-------+-----------+-----------+------------+---------+
|type     |number |old number |difference |%documented |%badname |
+=========+=======+===========+===========+============+=========+
|module   |1      |NC         |NC         |100.00      |0.00     |
+---------+-------+-----------+-----------+------------+---------+
|class    |0      |NC         |NC         |0           |0        |
+---------+-------+-----------+-----------+------------+---------+
|method   |0      |NC         |NC         |0           |0        |
+---------+-------+-----------+-----------+------------+---------+
|function |1      |NC         |NC         |0.00        |100.00   |
+---------+-------+-----------+-----------+------------+---------+


Raw metrics
-----------

+----------+-------+------+---------+-----------+
|type      |number |%     |previous |difference |
+==========+=======+======+=========+===========+
|code      |15     |55.56 |NC       |NC         |
+----------+-------+------+---------+-----------+
|docstring |8      |29.63 |NC       |NC         |
+----------+-------+------+---------+-----------+
|comment   |0      |0.00  |NC       |NC         |
+----------+-------+------+---------+-----------+
|empty     |4      |14.81 |NC       |NC         |
+----------+-------+------+---------+-----------+
```

```
Duplication
-----------

+-------------------------+------+---------+-----------+
|                         |now   |previous |difference |
+=========================+======+=========+===========+
|nb duplicated lines      |0     |NC       |NC         |
+-------------------------+------+---------+-----------+
|percent duplicated lines |0.000 |NC       |NC         |
+-------------------------+------+---------+-----------+


Messages by category
--------------------

+-----------+-------+---------+-----------+
|type       |number |previous |difference |
+===========+=======+=========+===========+
|convention |9      |NC       |NC         |
+-----------+-------+---------+-----------+
|refactor   |1      |NC       |NC         |
+-----------+-------+---------+-----------+
|warning    |0      |NC       |NC         |
+-----------+-------+---------+-----------+
|error      |0      |NC       |NC         |
+-----------+-------+---------+-----------+


Messages
--------

+---------------------------+------------+
|message id                 |occurrences |
+===========================+============+
|invalid-name               |4           |
+---------------------------+------------+
|trailing-whitespace        |2           |
+---------------------------+------------+
|trailing-newlines          |1           |
+---------------------------+------------+
|no-else-return             |1           |
+---------------------------+------------+
|missing-function-docstring |1           |
+---------------------------+------------+
|line-too-long              |1           |
+---------------------------+------------+



--------------------------------------------------------------------
Your code has been rated at 1.67/10 (previous run: 1.67/10, +0.00)
```

---

3. The name and output of the code coverage tool you used:
   ➔ The tool used is coverage.py Initial: The initial coverage was 76%

```
(base) niharshah@Nihars-MacBook-Pro HW01 % coverage report -m
Name              Stmts   Miss  Cover   Missing
-----------------------------------------------
TestTriangle.py      13      0   100%
triangle.py          12      6    50%   7-9, 12, 15-16
-----------------------------------------------
TOTAL                25      6    76%
```

Final: The final coverage is 81%, covering all the test cases.

```
(base) niharshah@Nihars-MacBook-Pro HW01 % coverage report -m
Name              Stmts   Miss  Cover   Missing
-----------------------------------------------
TestTriangle.py      13      0   100%
triangle.py          14      5    64%   7-9, 12, 15
-----------------------------------------------
TOTAL                27      5    81%
```

Coverage HTML:

Coverage report: 81%

coverage.py v6.4.1, created at 2022-10-11 15:38 -0400

Module	*statements*	*missing*	*excluded*	*coverage*
TestTriangle.py	13	0	0	100%
triangle.py	14	5	0	64%
Total	**27**	**5**	**0**	**81%**

coverage.py v6.4.1, created at 2022-10-11 15:38 -0400

---

4. Identify both your original test cases and new test cases that you created to achieve at least 80% code coverage.
   ➔ As a part of the initial request our aim was to make the code 100%, we fixed our code to more than 80% efficiency and post that when we ran the test cases against the new code and was able to achieve coverage of more than 80%. We achieved an efficiency of 81%. I tested the program with a lot of test cases in the Assignment and there was no need to add more test cases. The thing that worked for me was to make the correction to code and post that everything was working well.

---

5. Attach screen shots of the output of the static code analyzer as well as code coverage. You should show a screenshot of the analysis results both before and after any changes that you make to your programs:
   ➔ **I have attached the screenshot of the static code analysis and code coverage above before and after already above. Also uploaded this assignment to the new repository as mentioned above with git URL.**