Question 1

Run styleLint.py in Codio.

What happens when the code is run? Can you modify this code for a more favourable outcome? What amendments have you made to the code?

Question 2

pip install pylint

Run
 pylint
on pylintTest.py
Review each of the code errors returned. Can you correct each of the errors identified by pylint? Before correcting the code errors, save the pylintTest.py file with a new name (it will be needed again in the next question).

Question 3

pip install flake8
Run
flake8
on pylintTest.py
Review the errors returned. In what way does this error message differ from the error message returned by pylint?
Run flake8 on metricTest.py. Can you correct each of the errors returned by flake8? What amendments have you made to the code?

Question 4

pip install mccabe

Run
mccabe
on sums.py. What is the result?
Run
mccabe
on sums2.py. What is the result?
What are the contributors to the cyclomatic complexity in each piece of code?

Question 1: Answers

- Executing stylelint on the below source code:

```
1
2    # CODE SOURCE: SOFTWARE ARCHITECTURE WITH PYTHON
3
4    def factorial(n):
5    """ Return factorial of n """
6    if n == 0:
7    return 1
8    else:
9    return n*factorial(n-1)
```

- Produced:

```
Last login: Thu Dec 16 05:04:43 2021 from 192.168.11.51
codio@target-cobalt:~/workspace$ python3 styleLint.py
  File "styleLint.py", line 5
    """ Return factorial of n """
                                 ^
IndentationError: expected an indented block
codio@target-cobalt:~/workspace$
```

- The output points out the indentation error of the docstring within the function
- The code was amended to correct the error by indenting the docstring correctly
- The below is the next error after the correction:

```
IndentationError: expected an indented block
codio@target-cobalt:~/workspace$ python3 styleLint.py
  File "styleLint.py", line 7
    return 1
           ^
IndentationError: expected an indented block
codio@target-cobalt:~/workspace$
```

Question 2: Answers

- Pylint was run on the below code:

```
1
2     # SOURCE OF CODE: https://docs.pylint.org/en/1.6.0/tutorial.html
3
4     import string
5
6     shift = 3
7     choice = raw_input("would you like to encode or decode?")
8     word = (raw_input("Please enter text"))
9     letters = string.ascii_letters + string.punctuation + string.digits
10    encoded = ''
11    if choice == "encode":
12      for letter in word:
13        if letter == ' ':
14          encoded = encoded + ' '
15        else:
16          x = letters.index(letter) + shift
17          encoded=encoded + letters[x]
18        if choice == "decode":
19          for letter in word:
20            if letter == ' ':
21              encoded = encoded + ' '
22            else:
23              x = letters.index(letter) - shift
24              encoded = encoded + letters[x]
25
26    print encoded
```

- Below is the first error:

```
Last login: Thu Dec 16 05:04:48 2021 from 192.168.11.51
codio@target-cobalt:~/workspace$ python3 pylintTest.py
  File "pylintTest.py", line 26
    print encoded
                ^
SyntaxError: Missing parentheses in call to 'print'. Did you mean print(encoded)?
codio@target-cobalt:~/workspace$
```

- The below errors were corrected to debug the entire program using pylint:

```
Last login: Thu Dec 16 12:55:10 2021 from 192.168.11.51
codio@target-cobalt:~/workspace$ python3 pylintTest.py
  File "pylintTest.py", line 26
    print encoded
                ^
SyntaxError: Missing parentheses in call to 'print'. Did you mean print(encoded)?
codio@target-cobalt:~/workspace$ python3 pylintTest.py
Traceback (most recent call last):
  File "pylintTest.py", line 7, in <module>
    choice = raw_input("would you like to encode or decode?")
NameError: name 'raw_input' is not defined
codio@target-cobalt:~/workspace$ python3 pylintTest.py
Traceback (most recent call last):
  File "pylintTest.py", line 7, in <module>
    choice = string.raw_input("would you like to encode or decode?")
AttributeError: module 'string' has no attribute 'raw_input'
codio@target-cobalt:~/workspace$ python3 pylintTest.py
would you like to encode or decode?encode
Traceback (most recent call last):
  File "pylintTest.py", line 8, in <module>
    word = (raw_input("Please enter text"))
NameError: name 'raw_input' is not defined
codio@target-cobalt:~/workspace$ python3 pylintTest.py
would you like to encode or decode?encode
Please enter texthello
khoor
codio@target-cobalt:~/workspace$
```

Question 3: Answers

- Running the same code as above through the Linter of flake8 produces the below output:

```
Last login: Thu Dec 16 13:21:39 2021 from 192.168.11.51
codio@target-cobalt:~/workspace$ flake8 Pylint2.py
Pylint2.py:5:1: W293 blank line contains whitespace
Pylint2.py:12:3: E111 indentation is not a multiple of 4
Pylint2.py:14:7: E111 indentation is not a multiple of 4
Pylint2.py:16:7: E111 indentation is not a multiple of 4
Pylint2.py:17:7: E111 indentation is not a multiple of 4
Pylint2.py:17:14: E225 missing whitespace around operator
Pylint2.py:19:7: E111 indentation is not a multiple of 4
Pylint2.py:23:11: E111 indentation is not a multiple of 4
Pylint2.py:24:11: E111 indentation is not a multiple of 4
Pylint2.py:26:14: W292 no newline at end of file
```

- The output is focussed on stylistic errors
- The below code was tested too:

```
#CODE SOURCE: SOFTWARE ARCHITECTURE WITH PYTHON

"""
2 Module metricTest.py
3
4 Metric example - Module which is used as a testbed for static
checkers.
5 This is a mix of different functions and classes doing
different things.
6
7 """
8 import random
9
10 def fn(x, y):
11 """ A function which performs a sum """
12 return x + y
13
14 def find_optimal_route_to_my_office_from_home(start_time,
15 expected_time,
16 favorite_route='SBS1K',
17 favorite_option='bus'):
18
19
20 d = (expected_time - start_time).total_seconds()/60.0
21
22 if d<=30:
23 return 'car'
24
25 # If d>30 but <45, first drive then take metro
26 if d>30 and d<45:
27 return ('car', 'metro')
28
29 # If d>45 there are a combination of optionsWriting Modifiable a
[ 68 ]
30 if d>45:
31 if d<60:
32 # First volvo,then connecting bus
33 return ('bus:335E','bus:connector')
34 elif d>80:
35 # Might as well go by normal bus
36 return random.choice(('bus:330','bus:331',':'.
join((favorite_option,
37 favorite_route))))
38 elif d>90:
39 # Relax and choose favorite route
40 return ':'.join((favorite_option,
41 favorite_route))
42
43
44 class C(object):
45 """ A class which does almost nothing """
46
47 def __init__(self, x,y):
48 self.x = x
49 self.y = y
50
51 def f(self):
52 pass
53
```

```
61      54 def g(self, x, y):
62      55
63      56 if self.x>x:
64      57 return self.x+self.y
65      58 elif x>self.x:
66      59 return x+ self.y
67      60
68      61 class D(C):
69      62 """ D class """
70      63
71      64 def __init__(self, x):
72      65 self.x = x
73      66
74      67 def f(self, x,y):
75      68 if x>y:
76      69 return x-y
77      70 else:Chapter 2
78      [ 69 ]
79      71 return x+y
80      72
81      73 def g(self, y):
82      74
83      75 if self.x>y:
84      76 return self.x+y
85      77 else:
86      78 return y-self.x
```

- The below output was produced:

codio@target-cobalt:~/workspace$ flake8 metricTest.py
metricTest.py:2:1: E265 block comment should start with '# '
metricTest.py:2:48: W291 trailing whitespace
metricTest.py:13:8: E999 SyntaxError: invalid syntax
metricTest.py:16:1: E112 expected an indented block
metricTest.py:20:1: E128 continuation line under-indented for visual indent
metricTest.py:21:1: E128 continuation line under-indented for visual indent
metricTest.py:22:1: E128 continuation line under-indented for visual indent
metricTest.py:23:1: E112 expected an indented block
metricTest.py:27:8: E225 missing whitespace around operator
metricTest.py:28:1: E112 expected an indented block
metricTest.py:30:3: E261 at least two spaces before inline comment
metricTest.py:31:8: E225 missing whitespace around operator
metricTest.py:31:17: E225 missing whitespace around operator
metricTest.py:32:1: E112 expected an indented block
metricTest.py:34:3: E261 at least two spaces before inline comment
metricTest.py:34:80: E501 line too long (83 > 79 characters)
metricTest.py:35:2: E201 whitespace after '['
metricTest.py:35:5: E202 whitespace before ']'
metricTest.py:36:8: E225 missing whitespace around operator
metricTest.py:37:1: E112 expected an indented block
metricTest.py:37:8: E225 missing whitespace around operator
metricTest.py:38:1: E112 expected an indented block
metricTest.py:38:3: E261 at least two spaces before inline comment
metricTest.py:39:22: E231 missing whitespace after ','
metricTest.py:40:10: E225 missing whitespace around operator
metricTest.py:41:1: E112 expected an indented block
metricTest.py:41:3: E261 at least two spaces before inline comment
metricTest.py:42:35: E231 missing whitespace after ','
metricTest.py:42:45: E231 missing whitespace after ','

metricTest.py:43:1: E128 continuation line under-indented for visual indent
metricTest.py:44:1: E128 continuation line under-indented for visual indent
metricTest.py:45:10: E225 missing whitespace around operator
metricTest.py:46:1: E112 expected an indented block
metricTest.py:46:3: E261 at least two spaces before inline comment
metricTest.py:48:1: E128 continuation line under-indented for visual indent
metricTest.py:52:1: E112 expected an indented block
metricTest.py:54:24: E231 missing whitespace after ','
metricTest.py:55:1: E112 expected an indented block
metricTest.py:59:1: E112 expected an indented block
metricTest.py:62:1: E112 expected an indented block
metricTest.py:63:13: E225 missing whitespace around operator
metricTest.py:64:1: E112 expected an indented block
metricTest.py:65:10: E225 missing whitespace around operator
metricTest.py:66:1: E112 expected an indented block
metricTest.py:66:12: E225 missing whitespace around operator
metricTest.py:69:1: E112 expected an indented block
metricTest.py:72:1: E112 expected an indented block
metricTest.py:74:17: E231 missing whitespace after ','
metricTest.py:75:1: E112 expected an indented block
metricTest.py:75:8: E225 missing whitespace around operator
metricTest.py:76:1: E112 expected an indented block
metricTest.py:77:8: E231 missing whitespace after ':'
metricTest.py:78:2: E201 whitespace after '['
metricTest.py:78:5: E202 whitespace before ']'
metricTest.py:82:1: E112 expected an indented block
metricTest.py:83:13: E225 missing whitespace around operator
metricTest.py:84:1: E112 expected an indented block
metricTest.py:86:1: E112 expected an indented block
metricTest.py:86:19: W292 no newline at end of file

Question 4 - Answers

- Running the below code through the McCabe linter:

```
1
2     # SOURCE OF CODE: https://realpython.com/python-testing/
3
4 ▾   def test_sum():
5         assert sum([1, 2, 3]) == 6, "Should be 6"
6
7 ▾   if __name__ == "__main__":
8         test_sum()
9         print("Everything passed")
```

- Creates the below output:

```
codio@target-cobalt:~/workspace$ python -m mccabe sums.py
("4:0: 'test_sum'", 1)
('If 7', 2)
```

- Running the below code through the mcabe linter:

```
1     |
2     # SOURCE OF CODE: https://realpython.com/python-testing/
3
4 ▾   def test_sum():
5         assert sum([1, 2, 3]) == 6, "Should be 6"
6
7 ▾   def test_sum_tuple():
8         assert sum((1, 2, 2)) == 6, "Should be 6"
9
10 ▾  if __name__ == "__main__":
11        test_sum()
12        test_sum_tuple()
13        print("Everything passed")
```

- Creates the below output:

```
codio@target-cobalt:~/workspace$ python -m mccabe sums2.py
("7:0: 'test_sum_tuple'", 1)
("4:0: 'test_sum'", 1)
('If 10', 2)
```

- The cyclomatic complexity of a portion of code is determined by the number/depth of conditional control statements