Dhirubhai Ambani
Institute of Information and Communication Technology

Software Engineering

(IT - 314)

Lab 5:

Static Analysis

Name : Parth Prajapati
ID : 202001232

Language : Python
Analysis Tool : Pylint

S.No	Message Object	Expansion	Explanation
1.	C	Convention	It is displayed when the program is not following the standard rules.
2.	R	Refactor	It is displayed for bad code smell
3.	W	Warning	It is displayed for python specific problems
4.	E	Error	It is displayed when that particular line execution results some error
5.	F	Fatal	It is displayed when pylint has no access to further process that line.

Code 1 :

Git-hub link :
https://github.com/thartshorne/factorial/blob/master/factorial.py

Code:

```
# hello
import sys



def recur_factorial(n):
    """Function to return the factorial
    of a number using recursion"""
    if n == 1:
        return n
    else:
        return n*recur_factorial(n-1)


if __name__ == '__main__':
    arg = sys.argv
    print(recur_factorial(int(arg[1])))
```

Errors detected:

```
PS C:\Users\student\Desktop\Lab 6\python-mini-project\Testfiles> py -m pylint tmp.py
************* Module tmp
tmp.py:16:0: C0304: Final newline missing (missing-final-newline)
tmp.py:1:0: C0114: Missing module docstring (missing-module-docstring)
tmp.py:5:20: C0103: Argument name "n" doesn't conform to snake_case naming style (invalid-name)
tmp.py:8:4: R1705: Unnecessary "else" after "return", remove the "else" and de-indent the code inside it (no-else-return)

-------------------------------------------------------------------
Your code has been rated at 5.00/10 (previous run: 0.00/10, +5.00)

PS C:\Users\student\Desktop\Lab 6\python-mini-project\Testfiles>
```

Analysis :

- First 3 messages are conventional (C) suggesting it has not followed some standard rules for writing.

- The last message is R type - Refactor. It suggests that there is no meaning using the else statement after the return statement

Code : 2

Link :
https://github.com/anish-rajan/sum_of_array/blob/master/python/a3/q6.py
Fibonacci series:

Code:

```
def fib(n,prev=1,count=0,next=0):
    #in fibonacci difference of terms are in ap
    if count==n:
        return next
    else:
        next=prev+next
        prev=next-prev
        count=count+1
        return fib(n,prev,count,next)
n=input("enter a number")
fib(n)
```

Error detected:

```
PS C:\Users\student\Desktop\Lab 6\python-mini-project\Testfiles> py -m pylint tmp.py
************* Module tmp
tmp.py:10:0: C0303: Trailing whitespace (trailing-whitespace)
tmp.py:12:0: C0304: Final newline missing (missing-final-newline)
tmp.py:1:0: C0114: Missing module docstring (missing-module-docstring)
tmp.py:1:0: C0116: Missing function or method docstring (missing-function-docstring)
tmp.py:1:8: C0103: Argument name "n" doesn't conform to snake_case naming style (invalid-name)
tmp.py:1:8: W0621: Redefining name 'n' from outer scope (line 11) (redefined-outer-name)
tmp.py:1:25: W0622: Redefining built-in 'next' (redefined-builtin)
tmp.py:3:4: R1705: Unnecessary "else" after "return", remove the "else" and de-indent the code inside it (no-else-return)

------------------------------------------------------------------
```

Analysis :

- First 5 messages are conventional (C) suggesting it has not followed some standard rules for writing.
- For example message two indicates that there is no final empty line as per the standard convention.
- The 6th message is the warning type. And it says that you are using same named variable ‘n’ globally and locally as well.
- The last message is R type - Refactor. It suggests that there is no meaning using the else statement after the return statement.

Code :3

Link:
https://github.com/anish-rajan/sum_of_array/blob/master/python/a3/q5.py

Code:

```
#This is a recursive function that uses the property between 2 numbers to
calculate GCD
def gcd(x,z):
    if(z==0):
        return x
    else:
        return gcd(z,x%z)
x=input()
y=input()
print(gcd(x,y))
```

```
PS C:\Users\student\Desktop\Lab 6\python-mini-project\Testfiles> py -m pylint q3.py
************* Module q3
q3.py:3:0: C0325: Unnecessary parens after 'if' keyword (superfluous-parens)
q3.py:9:0: C0304: Final newline missing (missing-final-newline)
q3.py:1:0: C0114: Missing module docstring (missing-module-docstring)
q3.py:2:0: C0116: Missing function or method docstring (missing-function-docstring)
q3.py:2:8: C0103: Argument name "x" doesn't conform to snake_case naming style (invalid-name)
q3.py:2:10: C0103: Argument name "z" doesn't conform to snake_case naming style (invalid-name)
q3.py:2:8: W0621: Redefining name 'x' from outer scope (line 7) (redefined-outer-name)
q3.py:3:4: R1705: Unnecessary "else" after "return", remove the "else" and de-indent the code inside it (no-else-return)

-----------------------------------
Your code has been rated at 0.00/10
```

Analysis :

- First 6 messages are conventional (C) suggesting it has not followed some standard rules for writing.
- For example, the 2nd message indicates that there is no final empty line as per the standard convention.
- The second last message is the warning type. And it says that you are using the same named variable ‘x’ globally and locally as well.
- The last message is R type - Refactor. It suggests that there is no meaning using the else statement after the return statement.

Code : 4

Link :
https://github.com/anish-rajan/sum_of_array/blob/master/python/a3/q4.py

Code :

```
import random
def factorial(m):
    if m==0 or m==1: #Added 1 to the base case so that the number of times
the recursion call takes places decreses by 1.
        return 1
    else:
        return m*factorial(m-1)
m=input()
print(factorial(m))
```

```
PS C:\Users\student\Desktop\Lab 6\python-mini-project\Testfiles> py -m pylint q4.py
************* Module q4
q4.py:3:0: C0301: Line too long (121/100) (line-too-long)
q4.py:8:0: C0304: Final newline missing (missing-final-newline)
q4.py:1:0: C0114: Missing module docstring (missing-module-docstring)
q4.py:2:0: C0116: Missing function or method docstring (missing-function-docstring)
q4.py:2:14: C0103: Argument name "m" doesn't conform to snake_case naming style (invalid-name)
q4.py:2:14: W0621: Redefining name 'm' from outer scope (line 7) (redefined-outer-name)
q4.py:3:4: R1705: Unnecessary "else" after "return", remove the "else" and de-indent the code inside it (no-else-return)
q4.py:3:7: R1714: Consider merging these comparisons with 'in' by using 'm in (0, 1)'. Use a set instead if elements are hashable. (consider-using-in)
q4.py:1:0: W0611: Unused import random (unused-import)

--------------------------------------
Your code has been rated at 0.00/10
```

Analysis :

- First 5 messages are conventional (C) suggesting it has not followed some standard rules for writing.
- For example, the 1st message points out that the 3rd line is too long (>100 words).
- The second last message is R type - Refactor. It suggests that we should use ‘in’ keyword instead of using two if conditions.
- The last message is the warning type. And it indicates that we have not used the library which we imported in the first line.

Code : 5

Link:
https://github.com/Twiggecode/Integer-Sequences/blob/main/Divisor%20Function/divisor_function.py

Code

```
power_of_function, dividend = input(
    "In the Function σ(n) base x enter the base and n respictively : "
).split()
divisor = []
for i in range(1, int(int(dividend) / 2) + 1):
    if (int(dividend) % i) == 0:
        divisor.append(i)
divisor.append(int(dividend))

sum = 0
for i in divisor:
    sum += i ** int(power_of_function)
print(
    "sum of positive divisors function σ x(n) where x is",
    power_of_function,
    "and n is",
    dividend,
    "is :",
    sum,
)
print("aliquot sum s(n) where n is", dividend, "is :", sum -
int(dividend))
```

Error detected:

```
PS C:\Users\student\Desktop\Lab 6\python-mini-project\Testfiles> py -m pylint q5.py
************* Module q5
q5.py:21:0: C0304: Final newline missing (missing-final-newline)
q5.py:1:0: C0114: Missing module docstring (missing-module-docstring)
q5.py:10:0: W0622: Redefining built-in 'sum' (redefined-builtin)
q5.py:10:0: C0103: Constant name "sum" doesn't conform to UPPER_CASE naming style (invalid-name)

-----------------------------------
Your code has been rated at 6.36/10
```

Analysis :

- First 2 messages are conventional (C) suggesting it has not followed some standard rules for writing.
- The second last message is the warning type. And it indicates that we are using the same named variable ‘sum’ globally and locally as well.
- The last message says that the variable name “sum” is not uppercase naming style.