191109020-PYTHON ASSESSMENT 02

191109020-PYTHON ASSESSMENT 02

Answer all the questions:

Q.No.01

a)

```
import math as m
print(f"Sin(0): {m.sin(0)}")
print(f"Cos(0): {m.cos(0)}")
print(f"Tan(0): {m.tan(0)}")
print(f"Sec(0): {m.asin(0)}")
print(f"Cosec(1): {m.acos(1)}")
print(f"Cot(0): {m.atan(0)}")
```

```
Sin(0): 0.0
Cos(0): 1.0
Tan(0): 0.0
Sec(0): 0.0
Cosec(1): 0.0
Cot(0): 0.0
```

b)

```
def add(b):
    return lambda a:a+b
sum = add(60)
print(sum(40))
```

```
100
```

c)

```
def findSum(n) :
    sum = 0
    x = 1
    while x <=n :
        sum = sum + x
        x = x + 1
    return sum
n = int(input("Enter a number: "))
```

```
print (findSum(n))
```

```
Enter a number: 7
28
```

Q.No.02

a)

```
import statistics
list=[10.5,11.5,21.5,26.5,30.0]
print(statistics.mean(list))
```

```
20.0
```

b)

```
def name(a,b):
    print(a+b)
x=input('Enter the first name:')
y=input('Enter the last name:')
name(x,y)
```

```
Enter the first name:Jebaz
Enter the last name:Prabhakaran
JebazPrabhakaran
```

Q.No.03

```
#Function to simulate a traffic light
#It is required to make 2 user defined functions trafficLight() and
#light().

def trafficLight():
 signal = input("Enter the colour of the traffic light: ")
 if (signal not in ("RED","YELLOW","GREEN")):
     print("Please enter a valid Traffic Light colour in CAPITALS")
 else :
   value = light(signal) #function call to light()
 if (value == 0):
     print("STOP, Your Life is Precious.")
 elif (value == 1):
     print("PLEASE GO SLOW.")
 else:
    print("GO!,Thank you for being patient.")
#function ends here
```

```
def light(colour):
 if (colour == "RED"):
      return(0);
 elif (colour == "YELLOW"):
      return (1)
 else:
      return(2)
#function ends here
trafficLight()
print("SPEED THRILLS BUT KILLS")
```

```
Enter the colour of the traffic light: YELLOW
PLEASE GO SLOW.
SPEED THRILLS BUT KILLS
```

Q.No.04

Opening a file

```
# Open function to open the file "MyFile1.txt"
# (same directory) in read mode and
file1 = open("MyFile.txt", "w")
```

Closing a file

```
# Opening and Closing a file "MyFile.txt"
# for object name file1.
file1 = open("MyFile.txt", "w")
file1.close()
```

Writing to file

```
# writing to file

# Opening a file
file1 = open('myfile.txt', 'w')
L = ["This is Kulithalai \n", "This is Karur \n", "This is Tamil naadu \n"]
s = "Hello\n"

# Writing a string to file
file1.write(s)

# Writing multiple strings
# at a time
file1.writelines(L)
```

```
# Closing file
file1.close()

# Checking if the data is
# written to file or not
file1 = open('myfile.txt', 'r')
print(file1.read())
file1.close()
```

```
     Hello
     This is Kulithalai
     This is Karur
     This is Tamil naadu
```

Appending to a file

```
# Append vs write mode
file1 = open("myfile.txt", "w")
L = ["This is Kulithalai \n", "This is Karur \n", "This is Tamil naadu \n"]
file1.writelines(L)
file1.close()

# Append-adds at last
file1 = open("myfile.txt", "a")  # append mode
file1.write("Today \n")
file1.close()

file1 = open("myfile.txt", "r")
print("Output of Readlines after appending")
print(file1.read())
print()
file1.close()

# Write-Overwrites
file1 = open("myfile.txt", "w")  # write mode
file1.write("Tomorrow \n")
file1.close()

file1 = open("myfile.txt", "r")
print("Output of Readlines after writing")
print(file1.read())
print()
file1.close()
```

```
     Output of Readlines after appending
     This is Kulithalai
     This is Karur
     This is Tamil naadu
     Today
```

```
Output of Readlines after writing
Tomorrow
```

With Statement

```
# write data to a file using with statement

L = ["This is Kulithalai \n", "This is Karur \n", "This is Tamil naadu \n"]

# Writing to file
with open("myfile.txt", "w") as file1:
    # Writing data to a file
    file1.write("Hello \n")
    file1.writelines(L)

# Reading from file
with open("myfile.txt", "r+") as file1:
    # Reading form a file
    print(file1.read())
```

```
Hello
This is Kulithalai
This is Karur
This is Tamil naadu
```

Q.No.05

a)

```
import re
def text_match(text):
        patterns = 'ab{2,3}'
        if re.search(patterns,  text):
                return 'Found a match!'
        else:
                return('Not matched!')
print(text_match("ab"))
print(text_match("aabbbbbc"))
```

```
Not matched!
Found a match!
```

b)

```
def text_match(text):
        patterns = '^[a-z]+_[a-z]+$'
        if re.search(patterns,  text):
                return 'Found a match!'
        else:
                return('Not matched!')
print(text_match("aab_cbbbc"))
print(text_match("aab_Abbbc"))
print(text_match("Aaab_abbbc"))
```

```
Found a match!
Not matched!
Not matched!
```

c)

```
patterns = [ 'fox', 'dog', 'horse' ]
text = 'The quick brown fox jumps over the lazy dog.'
for pattern in patterns:
    print('Searching for "%s" in "%s" ->' % (pattern, text),)
    if re.search(pattern,  text):
        print('Matched!')
    else:
        print('Not Matched!')
```

```
Searching for "fox" in "The quick brown fox jumps over the lazy dog." ->
Matched!
Searching for "dog" in "The quick brown fox jumps over the lazy dog." ->
Matched!
Searching for "horse" in "The quick brown fox jumps over the lazy dog." ->
Not Matched!
```