Assignment 03

February 15, 2025

Name : Sajith Lakmal Liyanage

Date : 15 February 2025

(Q1) Write a lambda expression to get the product of two numbers.

`Run test for expression(5,6) Output : 30`

[114]:
```
x=lambda a1,a2:a1*a2
# Call the function
x(5,6)
```

[114]: 30

(Q2) Write a function to get the area of a circle from the radius . Hint: remember to import the right modul for being able to calculte the area of the circle.

Run test for functio n (10)

Output : 314.1592653589793

[116]:
```
import math

def Cal_circle_area(radius):
    return math.pi*radius**2

print(Cal_circle_area(10))
```

```
314.1592653589793
```

(Q3) Build a simple calculator which can: add, subtract, multiply , divide.

Hint: solve by writing a function that takes as argument two numbers and the operation and returns the desired output.

Run test for function(2,5,'d')

Output: 0.4

[118]:
```
def Calculator(num1,num2,operator):
    if operator == 'a':
        return num1+num2
    elif operator == 's':
        return num1-num2
```

```
    elif operator == 'd':
        return num1/num2
    elif operator == 'm':
        return num1*num2

print(Calculator(2,5,'d'))
```

```
0.4
```

(Q4) Define a class named Rectangle which can be constructed by a length and width. The Rectangle class has a method which can compute the area.

Run test for r = Rectangle(5,10) r.area()

Output: 50

[120]:
```
class Rectangle():
    def __init__(self,length,width):
        self.length = length
        self.width = width

    def area(self):
        return self.length * self.width

r=Rectangle(5,10)
print(r.area())
```

```
50
```

(Q5) Define a class named Shape and its subclass Square.

Shape objects can be consrtucted by name and length has an area function wich return 0

Square subclass has an init function which take a length and name as argument and has an area method and a describe method what prints the name of the Shape.

Print the area from Square class. Run test for : s = Square('square',5) print(s.area()) print(s.describe())

Output: The area is: 25

This is a: square

[122]:
```
class Shape():
    def __init__(self,name,length):
        self.name = name
        self.length = length
    def area(self):
        return 0

class Square(Shape):
```

```
    def __init__(self,name,length):
        super().__init__(name,length)

    def area(self):
        return f"the area is: {self.length ** 2}"
    def describe(self):
        return f"This is a: {self.name}"

s = Square('square',5)
print(s.area())
print(s.describe())
```

```
the area is: 25
This is a: square
```

[ ]: