Assignment 3

February 16, 2025

Computer Science Assignment 3

Kaung Ye Lin

Date - 16.2.25

1. Write a lambda expression to get the product of two numbers .

   Run test for expression(5,6)

   Output : 30

[178]:
```
Product_of_two_number = lambda x,y : x * y

Number = Product_of_two_number(5,6)
print("Output:", Number)
```

```
Output: 30
```

---

2. Write a function to get the area of a circle from the radius . Hint: remember to import the right module for being able to calculte the area of the circle.

Run test for function (10)

Output : 314.1592653589793

[181]:
```
import math

pi = math.pi

def function(radius):
    return  pi * (radius ** 2)

area_result = function(10)
print("Output:", area_result)
```

```
Output: 314.1592653589793
```

---

3. 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

[184]:
```
def function(x,y,operation):
    a = x + y
    m = x - y
    p = x * y
    d = x / y

    if operation == 'a' :
        return a
    elif operation == 'm' :
        return m
    elif operation == 'p' :
        return p
    elif operation == 'd' :
        return d
    else :
        print ("try again with correct function")


result = function(2,5,'d')
print("Output:", result)
```

Output: 0.4

---

4. 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

[188]:
```
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("Output:", r.area())
```

```
Output: 50
```

---

5. 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
```

[191]:
```
class Shape: # this is the class for general shape
    def __init__ (self, name, length):
        self.name = name
        self.length = length
    def area (self, name, length):
        return 0

class Square(Shape): # sub class for square
    def __init__ (self, name, length):
        super().__init__ (name,length)
    def area (self):
        return f"The area is: \n\t {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
```

[ ]:
```

```