Revision 3

October 9, 2024

1 Exercise 3

Solve the following exercises in Jupyter Notebook, run all cells and download as pdfdocument. Return your pdf in Assignment 3 created under module Python Part I

1.Write a lambda expression to get the product of two numbers.Run test for expression(5,6) Output:30

[5]:
```
x = lambda num1,num2: (num1*num2)
x(5,6)
```

[5]:
```
30
```

2.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 function(10)Output:314.1592653589793

[7]:
```
from math import pi
radius = float(input("Enter the radius of your circle: "))
area = pi*radius**2
print("Area of your circle is: ",area)
```

```
Enter the radius of your circle:  10
```

```
Area of your circle is:  314.1592653589793
```

[10]:
```
from math import pi
def area_of_circle(r):
    return pi*r**2
area_of_circle(10)
```

[10]:
```
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

[16]:
```
def calculator(num1, num2, operation):
    if operation == 'a':  # Addition
        return num1 + num2
    elif operation == 's':  # Subtraction
```

```
        return num1 - num2
    elif operation == 'm':  # Multiplication
        return num1 * num2
    elif operation == 'd':  # Division
        return num1 / num2 if num2 != 0 else "Error: Division by zero"
    else:
        return "Error: Invalid operation"
result_calculator = calculator(2, 5, 'd')
result_calculator
```

[16]: 0.4

[13]:
```
def calculator(num1,num2,op):
    if op=='a':
        return num1+num2
    elif op=='s':
        return num1-num2
    elif op=='m':
        return num1*num2
    elif op=='d':
        return num1/num2
calculator(2,5,'d')
```

[13]: 0.4

[1]:
```
##defining operations
print("1 - Add")
print("2 - Subtract")
print("3 - Multiply")
print("4 - Divide")
option = int(input("Choose operation from above numbers: ")) # ask tomchoose an␣
 ↪operation
result = 0

if(option in [1,2,3,4]):
    num1 = float(input("Enter first number: "))     #float to have decimals
    num2 = float(input("Enter second number: "))
    if(option == 1):
        result = num1 + num2
    elif(option == 2):
        result = num1 - num2
    elif(option == 3):
        result = num1 * num2
    elif(option == 4):
        result = num1 / num2

else:
```

```
    print("Invalid operation entered")
print("The result of the operation is: ",(result))
```

```
1 - Add
2 - Subtract
3 - Multiply
4 - Divide

Choose operation from above numbers:  4
Enter first number:  2
Enter second number:  5

The result of the operation is:  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

[4]:
```
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)
r.area()
```

[4]: 50

[ ]:

[3]:
```
class Rectangle():
    def __init__(self,length,width):
        self.length = length
        self.width = width
    def area(self):
        return self.length*self.width

l=int(input("Enter the length of the Rectangle: "))
w=int(input("Enter the width of the Rectangle: "))

r=Rectangle(l,w)
print("Area of the Rectangle is: ",r.area())
```

```
Enter the length of the Rectangle:  10
Enter the width of the Rectangle:  5

Area of the Rectangle is:  50
```

[6]:
```
# Define the Rectangle class
class Rectangle:
    def __init__(self, length, width):
        self.length = length
        self.width = width

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

# Test the class with the given input
r = Rectangle(5, 10)
result_rectangle_area = r.area()
result_rectangle_area
```

[6]: 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

[5]:
```
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 describe(self):
        print('This is a:',self.name)
    def area(self):
        print('The area is: ')
        return self.length**2
s = Square('square',5)
s.area()
```

```
The area is:
```

[5]: 25

[6]:
```
s.describe()
```

```
This is a: square
```

[8]:
```
print (s.area())
s.describe()
```

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

[9]:
```
# Define the base class Shape
class Shape:
    def __init__(self, name, length):
        self.name = name
        self.length = length

    def area(self):
        return 0

# Define the subclass Square
class Square(Shape):
    def __init__(self, name, length):
        super().__init__(name, length)

    def area(self):
        return self.length ** 2

    def describe(self):
        return f"This is a: {self.name}"

# Test the Square class with the given input
s = Square('square', 5)
print(f"The area is: {s.area()}")  # Output: The area is: 25
print(s.describe())  # Output: This is a: square
```

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

[ ]:
```

```