

<b>Name of student: Abhay Omprakash Prajapati</b>		
<b>Roll no: 41</b>	<b>Tutorial No: 5</b>	
<b>Title of LAB Assignment:</b> To implement programs on OOP Concepts in python		
<b>DOP: 25-10-2023</b>	<b>DOS:</b>	
<b>CO Mapped:</b>	<b>PO Mapped:</b>	<b>Signature:</b>

### 1. Aim:

To demonstrate various Python programming concepts, including class creation, multiple inheritance, method overloading, operator overloading, and exception handling.

### 2. Theory:

Creating a class to compute the area and perimeter of a circle.

Implementing multiple inheritance in Python.

Defining a program with the same method name but multiple arguments.

Overloading operators in Python.

Handling various exceptions in Python.

### 3. Code:

#### A. Creating a Class to Compute the Area and Perimeter of a Circle:

```
import math

class Circle:
    def __init__(self, radius):
        self.radius = radius

    def area(self):
        return math.pi * self.radius**2

    def perimeter(self):
        return 2 * math.pi * self.radius

circle = Circle(5)
print("Circle Area:", circle.area())
print("Circle Perimeter:", circle.perimeter())
```

#### B. Implementing Multiple Inheritance in Python:

```
class Parent1:
    def show(self):
        print("This is from Parent1")

class Parent2:
    def show(self):
        print("This is from Parent2")

class Child(Parent1, Parent2):
    pass

child = Child()
child.show()
```

#### C. Defining a Program with the Same Method Name but Multiple Arguments:

```
class MyClass:
    def argsMethods(self, arg1, arg2=None):
        if arg2 is not None:
            print(f"Arguments: {arg1}, {arg2}")
        else:
            print(f"Argument: {arg1}")

obj = MyClass()
obj.example_method(10)
obj.example_method(20, 30)
```

## D. Overloading Operators in Python:

```
class ComplexNumber:
    def __init__(self, real, imag):
        self.real = real
        self.imag = imag

    def __add__(self, other):
        return ComplexNumber(self.real + other.real, self.imag + other.imag)

    def __str__(self):
        return f"{self.real} + {self.imag}i"

c1 = ComplexNumber(1, 2)
c2 = ComplexNumber(3, 4)
c3 = c1 + c2
print("Sum of Complex Numbers:", c3)
```

## E. Handling Various Exceptions in Python:

```
try:
    num1 = int(input("Enter a number: "))
    num2 = int(input("Enter another number: "))
    result = num1 / num2
    print("Result:", result)
except ZeroDivisionError:
    print("Division by zero is not allowed.")
except ValueError:
    print("Invalid input. Please enter a valid number.")
except Exception as e:
    print(f"An error occurred: {e}")
```

## 4. Conclusion:

In this practical, we demonstrated various Python programming concepts, including class creation, multiple inheritance, method overloading, operator overloading, and exception handling. These concepts are fundamental for building robust and versatile Python programs.

## 5. Output:

A.

The Run window displays the output of a Python script. The output consists of two lines: "Circle Area: 78.53981633974483" and "Circle Perimeter: 31.41592653589793". Below the output, it states "Process finished with exit code 0". A notification at the bottom right suggests the "Kubernetes" plugin is available, with buttons for "Configure plugins..." and "Don't suggest again". The status bar at the bottom indicates the file is "main.py" in the "pythonProject" directory, using Python 3.10 with UTF-8 encoding and 4 spaces for indentation.

```
Run main x
/home/approximator/PycharmProjects/pythonProject/pythonProject/venv/bin/python /home/approximator/PycharmProjects/pythonProject/pythonProject/main.py
Circle Area: 78.53981633974483
Circle Perimeter: 31.41592653589793

Process finished with exit code 0

Suggested plugin Kubernetes available.
Configure plugins... Don't suggest again

pythonProject > main.py 12:1 LF UTF-8 4 spaces Python 3.10 (pythonProject)
```

B.

The Run window displays the output of a Python script. The output is a single line: "This is from Parent1". Below the output, it states "Process finished with exit code 0". A notification at the bottom right suggests the "Kubernetes" plugin is available, with buttons for "Configure plugins..." and "Don't suggest again". The status bar at the bottom indicates the file is "main.py" in the "pythonProject" directory, using Python 3.10 with UTF-8 encoding and 4 spaces for indentation.

```
Run main x
/home/approximator/PycharmProjects/pythonProject/pythonProject/venv/bin/python /home/approximator/PycharmProjects/pythonProject/pythonProject/main.py
This is from Parent1

Process finished with exit code 0

Suggested plugin Kubernetes available.
Configure plugins... Don't suggest again

pythonProject > main.py 14:13 LF UTF-8 4 spaces Python 3.10 (pythonProject)
```

C.

The Run window displays the output of a Python script. The output shows "Argument: 10" and "Arguments: 20, 30". Below the output, it states "Process finished with exit code 0". A notification at the bottom right suggests the "Kubernetes" plugin is available, with buttons for "Configure plugins..." and "Don't suggest again". The status bar at the bottom indicates the file is "main.py" in the "pythonProject" directory, using Python 3.10 with UTF-8 encoding and 4 spaces for indentation.

```
Run main x
/home/approximator/PycharmProjects/pythonProject/pythonProject/venv/bin/python /home/approximator/PycharmProjects/pythonProject/pythonProject/main.py
Argument: 10
Arguments: 20, 30

Process finished with exit code 0

Suggested plugin Kubernetes available.
Configure plugins... Don't suggest again

pythonProject > main.py 8:16 LF UTF-8 4 spaces Python 3.10 (pythonProject)
```

D.

The Run window displays the output of a Python script. The output is a single line: "Sum of Complex Numbers: 4 + 6i". Below the output, it states "Process finished with exit code 0". A notification at the bottom right suggests the "Kubernetes" plugin is available, with buttons for "Configure plugins..." and "Don't suggest again". The status bar at the bottom indicates the file is "main.py" in the "pythonProject" directory, using Python 3.10 with UTF-8 encoding and 4 spaces for indentation.

```
Run main x
/home/approximator/PycharmProjects/pythonProject/pythonProject/venv/bin/python /home/approximator/PycharmProjects/pythonProject/pythonProject/main.py
Sum of Complex Numbers: 4 + 6i

Process finished with exit code 0

Suggested plugin Kubernetes available.
Configure plugins... Don't suggest again

pythonProject > main.py 12:18 (385 chars, 15 line breaks) LF UTF-8 4 spaces Python 3.10 (pythonProject)
```

E.

