NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY, GREATER NOIDA
(AN AUTONOMOUS INSTITUTE)

SEMESTER-EVEN

Problem Solving Using Advance Python Lab

List of Experiments

1. Write a program illustrating class definition and accessing class members.
2. Write a program to implement default constructor, parameterized constructor, and destructor.
3. Create a Python class named Rectangle constructed by a length and width.
   a. Create a method called area which will compute the area of a rectangle.
4. Create a class called Numbers, which has a single class attribute called MULTIPLIER, and a constructor which takes the parameters x and y (these should all be numbers).
   a. Write an instance method called add which returns the sum of the attributes x and y.
   b. Write a class method called multiply, which takes a single number parameter a and returns the product of a and MULTIPLIER.
   c. Write a static method called subtract, which takes two number objects, b and c, and returns b - c.
   d. Write a method called value which returns a tuple containing the values of x and y.
5. Create a class named as Student to store the name and marks in three subjects. Use List to store the marks.
   a. Write an instance method called compute to compute total marks and average marks of a student.
   b. Write a method called display to display student information.
6. Create a class Employee that keeps a track of the number of employees in an organization and also stores their name, designation and salary details.
   a. Write a method called getdata to take input (name, designation, salary) from user.
   b. Write a method called average to find average salary of all the employees in the organization.
   c. Write a method called display to print all the information of an employee.

7. Create a Python class named Circle constructed by a radius. Use a class variable to define the value of constant PI.

a. Write two methods to be named as area and circum to compute the area and the perimeter of a circle respectively by using class variable PI.

b. Write a method called display to print area and perimeter.

8. Create a class called String that stores a string and all its status details such as number of uppercase letters, lowercase letters, vowels ,consonants and space in instance variables.

a. Write methods named as count_uppercase, count_lowercase, count_vowels, count_consonants and count_space to count corresponding values.

b. Write a method called display to print string along with all the values computed by methods in (a).

9. Write a program that has a class called Fraction with attributes numerator and denominator.

a. Write a method called getdata to enter the values of the attributes.

b. Write a method show to print the fraction in simplified form.

10. Write a program that has a class Numbers with a list as an instance variable.

a. Write a method called insert_element that takes values from user.

b. Write a class method called find_max to find and print largest value in the list.

11. Write a program that has a class Point with attributes x and y.

a. Write a method called midpoint that returns a midpoint of a line joining two points.

b. Write a method called length that returns the length of a line joining two points.

12. Create a class called Complex. Write a menu driven program to read, display, add and subtract two complex numbers by creating corresponding instance methods.

13. Write a Program to illustrate the use of __str__(), __repr__(), __new__, __doc__, __dict__, __name__ and __bases__ methods.

14. Create a BankAccount class. Your class should support the following methods:

a. __init__(self, account_no)

b. deposit (self, account_no, amount)

c. withdraw (self, account_no, amount)
d. get_balance (self, account_no)

15. Write a program to illustrate the use of following built-in methods:
a. hasattr(obj,attr)
b. getattr(object, attribute_name [, default])
c. setattr(object, name, value)
d. delattr(class_name, name)

16. Write a program to create class Employee. Display the personal information and salary details of 5 employees using single inheritance.

17. WAP that extends the class Employee. Derive two classes Manager and Team Leader from Employee class. Display all the details of the employee working under a particular Manager and Team Leader.

18. Write a program that has a class Point. Define another class Location which has two objects (Location and destination) of class Point. Also, define a function in Location that prints the reflection on the y-axis.

19. WAP that create a class Student having attribute as name and age and Marks class inheriting Students class with its own attributes marks1, marks2 and marks3 as marks in 3 subjects. Also, define the class Result that inherits the Marks class with its own attribute total. Every class has its own display() method to display the corresponding details. Use __init__() and super() to implement the above classes.

20. Write a program that create a class Distance with members km and metres. Derive classes School and office which store the distance from your house to school and office along with other details.

21. Write a program to create an abstract class Vehicle. Derive three classes Car, Motorcycle and Truck from it. Define appropriate methods and print the details of vehicle.

22. Write a program that has a class Polygon. Derive two classes Rectangle and triangle from polygon and write methods to get the details of their dimensions and hence calculate the area.

23. Write a program that extends the class Shape to calculate the area of a circle and a cone .(use super to inherit base class methods)

24. Write a program to demonstrate hybrid inheritance and show MRO for each class.
25. Write a program to overload + operator to multiply to fraction object of fraction class which contain two instance variable numerator and denominator. Also, define the instance method simplify() to simplify the fraction objects.
26. Write a program to compare two-person object based on their age by overloading > operator.
27. Write a program to overload inoperator.
28. WAP to create a Complex class having real and imaginary as it attributes. Overload the +,-,/,* and += operators for objects of Complex class.
29. Write a program to inspect the object using type() ,id(), isinstance(), issubclass() and callable() built-in function.
30. WAP to inspect the program code using the functions of inspect module.
31. Write a program to create a new list containing the first letters of every element in an already existing list.
32. Write a program using reduce() function to calculate the sum of first 10 natural numbers
33. Write a program that convert a list of temperatures in Celsius into Fahrenheit using map() function.
34. Write a program that creates an iterator to print squares of numbers.
35. Write a program that create a custom iterator to create even numbers.
36. Write a program to create a generator that starts counting from 0 and raise an exception when counter is equal to 10.
37. Write a program to create a generator to print the Fibonacci number.
38. Write a program to create an arithmetic calculator using tkinter.
39. Write a program to draw colored shapes (line, rectangle, oval) on canvas.
40. Write a program to create a window that disappears automatically after 5 seconds.
41. Write a program to create a button and a label inside the frame widget. Button should change the color upon hovering over the button and label should disappear on clicking the button.
42. Write a program to create radio-buttons (Male, Female, and Transgender) and a label. Default selection should be on Female and the label must display the current selection made by user.
43. Write a program to display a menu on the menu bar.
44. Write a NumPy program to create an array of (3, 4) shape, multiply every element value by 3 and display the new array.

45. Write a NumPy program to compute the multiplication of two given matrixes.

46. Write a Program to create a series from a list, numpy array and dict.

47. Write a Program to convert a numpy array to a dataframe of given shape.

48. Write a program to count number of missing values in each column.

49. Write a program to replace missing values in a column of a dataframe by the mean value of that column.

50. Write a Pandas program to create a line plot of the opening, closing stock prices of Alphabet Inc. between two specific dates. Use the alphabet_stock_data.csv file to extract data.