Training On Python

Lecture – 6
Inheritance

Inheritance In Python

The concept of using properties of one class into another class without creating object of that class explicitly is known as inheritance.

- A class which is extended by another class is known as ‘super’ class.
- A class which is extending another class is known as ‘sub’ class.
- Both super class properties and sub class properties can be accessed through subclass reference variable.
- Super class properties directly we can use within the subclass.

Syntax:-

```
class A:
    #Class A Code
class B(A):
    #Class B Code
```

Types Of Inheritance In Python

The Python Programming Language provides five types of Inheritance:-

1. Single Inheritance:- In Single Inheritance there is a single base class and single derived class.
2. Multiple Inheritance:- In Multiple Inheritance there are multiple base classes and single derived class.
3. Hierarchical Inheritance:- In Hierarchical Inheritance there is a single base class and multiple derived class.
4. Multi-level Inheritance:- The concept of inheriting properties from multiple classes into single class with the concept of ‘one after another’ is known as a multilevel inheritance.
5. Hybrid Inheritance:- If you combine two or more inheritance then resultant inheritance is called Hybrid Inheritance.

Example Application Of Single Inheritance

```
#Example Application of single inheritance
class A:
    def showA(self):
        print("This message from base class")
class B(A):
    def showB(self):
        print("This message from derived class")
b=B() #Creation of object
b.showA() #This message from base class
b.showB() #This message from derived class
```

Example Application Of Multiple Inheritance

```
#Example Application Of Multiple Inheritance
class x:
    def m1(self):
        print('in m1 of x')
class y:
    def m2(self):
        print('in m2 of y')
class z(x,y):
    def m3(self):
        print('in m3 of z')
```

Example Application Of Multiple Inheritance (cont..)

```
z1=z()
z1.m1()
z1.m2()
z1.m3()
y1=y()
y1.m2()
x1=x()
x1.m1()
```

Output:-

```
in m1 of x
in m2 of y
in m3 of z
in m2 of y
in m1 of x
```

Example Application Of Hierarchical Inheritance

```
# Example Application Of Hierarchical Inheritance
class Shape:
    def setValue(self,s):
        self.s=s
class Square(Shape):
    def area(self):
        return self.s*self.s
class Cube(Shape):
    def volume(self):
        return self.s*self.s*self.s
```

Example Application Of Hierarchical Inheritance (cont..)

```
x=int(input("Enter side of square : "))
sq=Square()
sq.setValue(x)
print("Area of square:",sq.area())
x=int(input("Enter side of cube : "))
cu=Cube()
cu.setValue(x)
print("Area of square:",cu.volume())
```

Output:-

```
Enter side of square : 10
Area of square: 100
Enter side of cube : 10
Area of square: 1000
```

Example Application Of Multilevel Inheritance

```
# Example Application Of Multi-level Inheritance
class Employee:
    def setEmployee(self,empid,empname):
        self.empid=empid
        self.empname=empname
    def getEmployee(self):
        print("Employee Id=",self.empid)
        print("Employee Name=",self.empname)
```

Example Application Of Multilevel Inheritance (cont..)

```
class Payroll(Employee):
    def setPayroll(self,bs,hra,da):
        self.bs=bs
        self.hra=hra
        self.da=da
    def getPayroll(self):
        print("Basic Salary=",self.bs)
        print("House Rent Allownces=",self.hra)
        print("Dearness Allownces=",self.da)
```

Example Application Of Multilevel Inheritance (cont..)

```
class Payslip(Payroll):
    def netSalary(self):
        print("Net Salary=",(self.bs+self.hra+self.da))
eid=int(input("Enter Employee Id : "))
ename=input("Enter Employee Name : ")
b=int(input("Enter Basic Salary : "))
h=int(input("Enter House Rent Allownces : "))
d=int(input("Enter Dearness Allownces : "))
ps=Payslip()
ps.setEmployee(eid,ename)
ps.setPayroll(b,h,d)
```

Example Application Of Multilevel Inheritance (cont..)

```
class Payslip(Payroll):
    def netSalary(self):
        print("Net Salary=",(self.bs+self.hra+self.da))
eid=int(input("Enter Employee Id : "))
ename=input("Enter Employee Name : ")
b=int(input("Enter Basic Salary : "))
h=int(input("Enter House Rent Allownces : "))
d=int(input("Enter Dearness Allownces : "))
ps=Payslip()
ps.setEmployee(eid,ename)
ps.setPayroll(b,h,d)
print("**************PAY SLIP***************")
ps.getEmployee()
ps.getPayroll()
ps.netSalary()
```

Example Application Of Multilevel Inheritance (cont..)

Output:-

Enter Employee Id : 1001

Enter Employee Name : Brijesh Mishra

Enter Basic Salary : 35000

Enter House Rent Allownces : 15000

Enter Dearness Allownces : 10000

**********PAY SLIP**********

Employee Id= 1001

Employee Name= Brijesh Mishra

Basic Salary= 35000

House Rent Allownces= 15000

Dearness Allownces= 10000

Net Salary= 60000