

'''

Experiment 3.1 : Implementing classes and objects in python.

Name: Khan Arshad Abdulla

Roll No: 20CO24

2021-2022

THEORY:

A class is a user-defined blueprint or prototype from which objects are created. Classes provide a means of bundling data and functionality together.

Creating a new class creates a new type of object, allowing new instances of that type to be made. Each class instance can have attributes attached to it for maintaining its state. Class instances can also have methods (defined by their class) for modifying their state.

An Object is an instance of a Class. A class is like a blueprint while an instance is a copy of the class with actual values. Itâ€™s not an idea anymore, itâ€™s an actual dog, like a dog of breed pug whoâ€™s seven years old. You can have many dogs to create many different instances, but without the class as a guide, you would be lost, not knowing what information is required.

Creating classes in Python

In Python, a class can be created by using the keyword class, followed by the class name.

'''

```
class Employee:
```

```
    '''
```

```
    An Employee class having employee attributes like empno, ename, and sal.
```

```
    It also has methods like setprop, display, etc.
```

```
    '''
```

```
    #class attributes
```

```
    empno=None
```

```
    ename=None
```

```
    sal=None
```

```
    dept=None
```

```
    loc=None
```

```
    def setprop(self,num,name,sal,dept,loc):
```

```
        self.empno=num
```

```
        self.ename=name
```

```
        self.sal=sal
```

```
        self.dept=dept
```

```
        self.loc=loc
```

```
    def getprop(self):
```

```
        return self.empno,self.ename,self.sal,self.dept,self.loc
```

```
    # in Python, static methods are simply used as utility functions
```

```
    @staticmethod          # decorator
```

```
    def retire(age):
```

```
        if age>=60:
```

```
            print("Employee Retires.")
```

```

        else:
            print("Employee can still work.")

#driver code
def main():
    '''
    Our main function of Exp301 having a driver code.
    '''
    e=Employee()
    e.setprop(1,"Sachin",30000,"Computer","Mumbai")
    en,name,sal,dept,l=e.getprop()
    #print(en,name,sal,dept,l)
    e1=Employee()
    e1.setprop(2,"Shamim",54000,"Computer","Navi Mumbai")
    e2=Employee()
    e2.setprop(3,"Khatib",50000,"Computer Testing","Navi Mumbai")

    el=[e,e1,e2]
    sal=0
    emp_hs=Employee()
    for i in el:
        if i.sal>sal:
            sal=i.sal
            emp_hs=i

    print("Employee with highest salary is",emp_hs.ename)
    Employee.retire(42)
    #print(el[0].ename)

if __name__=="__main__":
    print(main.__doc__)
    main()
'''

```

OUTPUT:

Our main function of Exp301 having a driver code.

Employee with highest salary is Shamim  
Employee can still work.

CONCLUSION:

In this experiment we have successfully implemented class and its objects.  
I understood that class is used for defining a particular type of object,  
Because  
Python objects can have both functions and data elements.