

Practical No.10 - OOPS-II

DATE:

Q1.] Write a Python program to implement abstract classes in Python.

Code:

```
from abc import ABC, abstractmethod
```

```
class AbsBaseClass(ABC):
```

```
    def __init__(self):
```

```
        print("I am the constructor of the  
        abstract class")
```

```
    @abstractmethod
```

```
    def abstrmeth(self):
```

```
        ("but pass")
```

```
    def concremeth(self):
```

```
        print("I am concrete method")
```

```
class Derived(AbsBaseClass):
```

```
    def abstrmeth(self):
```

```
        print("I am redefined abstract method")
```

```
o = Derived()
```

```
o.abstrmeth()
```

```
o.concremeth()
```

Output:

I am the constructor of the abstract class

~~I am a concrete meth~~

I am redefined abstract method

I am concrete method.

2)

Write a python program to implement the concept of abstract method in abstract class.

Code :

```
from abc import ABC, abstractmethod

class AbsBaseClass(ABC):
    def __init__(self):
        print("I am the constructor of abstract method")
    @abstractmethod
    def abstmeth(self):
        print("I am abstract method")
    def concremeth(self):
        print("I am concrete method")

class Derived(AbsBaseClass):
    def abstmeth(self):
        super().abstmeth()
        print("I am redefined abstract method")

D = Derived()
D.abstmeth()
D.concremeth()
```

Output :

```
I am redefined abstract method
I am the constructor of abstract class
I am abstract method
I am redefined abstract method
I am concrete method.
```


Q) Write a python program to implement the concept of interface using abstract class.

Code:

```
from abc import ABC, abstractmethod

class Interface(ABC):
    @abstractmethod
    def abstractMeth1(self):
        pass
    @abstractmethod
    def abstractMeth2(self):
        pass

class Derived(Interface):
    def abstractMeth1(self):
        print("Abstract method 1 defined")
    def abstractMeth2(self):
        print("Abstract method 2 defined")

o = Derived()
print("Interface methods defined")
o.abstractMeth1()
o.abstractMeth2()
```

Output:

Interface method defined.

~~abstract method 1 defined~~
26/10/20 abstract method 2 defined.