1. What is the concept of an abstract superclass?

i. Abstract means partially implemented.

ii. A class is called as abstract class if it contains at-least one abstract method. An abstract method is a method which is only declared but not implemented. It’s next level child class is responsible for implementation.

iii. Abstract class may not be instantiated.

Eg :

class ABC:

def m(self):

pass

class XYZ(ABC):

def m(self,a,b):

return a+b

1. What happens when a class statement's top level contains a basic assignment statement?

If the class statement’s top level contains a basic assignment stamen then it can be usable in further codes if we are using the same variable without re-assigning any other value. In case we are re-assigning any value PVM will consider the recent assigned value.

1. Why does a class need to manually call a superclass's \_\_init\_\_ method?

When we will create an object of the child class it will only execute the constructor present in particular class. If a class extends to a super class and we want to execute super class constructor then we need to call super class constructor manually from child class constructor with respwective arguments.

Eg :

class Parent:

def \_\_init\_\_(self,B,C):

print(B, C)

self.B = B

self.C = C

class Child(Parent):

def \_\_init\_\_(self, B, C):

super().\_\_init\_\_(B,C)

c = Child(10,20)

1. How can you augment, instead of completely replacing, an inherited method?

We can augment a method instead of completely replacing it by performing different operations.

Eg :

class ABC:

def m(self):

global a

a = 10

class XYZ(ABC):

def m(self):

b = 20

c = 30

return b+c

x = XYZ()

print(x.m())

5. How is the local scope of a class different from that of a function?

i. Local variables have the scope with in the function, method. It will get destroyed by destructor once the execution completes. If we will try to access them out side of that block we will get error.

ii. If we are creating a function then we can call it within the module, as it is provides functionality all across the module.