Q1. What is Abstraction in OOps? Explain with an example.

Abstraction in python is defined as a process of handling complexity by hiding unnecessary information from the user @abc.abstractclassmethod

Q2. Differentiate between Abstraction and Encapsulation. Explain with an example.

Abstraction is hiding the details and implementation of the code

Encapsulation is hiding the data and controlling the visibility of the code

Q3. What is abc module in python? Why is it used?

This module provides the infrastructure for defining abstract base classes (ABCs) in Python, as outlined in PEP 3119; see the PEP for why this was added to Python. (See also PEP 3141 and the numbers module regarding a type hierarchy for numbers based on ABCs.)

Q4. How can we achieve data abstraction?

In Python, abstraction can be achieved by having/using abstract classes and methods in our programs. Understanding Abstract Methods and Classes: An abstract method is a method that is declared, but does not contain implementation.

Q5. Can we create an instance of an abstract class? Explain your answer.

Abstract classes cannot be instantiated, but they can be subclassed. When an abstract class is subclassed, the subclass usually provides implementations for all of the abstract methods in its parent class. However, if it does not, then the subclass must also be declared abstract .

In [ ]: