**Module 8: Discussion Forum**

Briefly describe the difference between a class and an instance of a class. In a Python class, how do you hide an attribute from code outside the class? Provide code examples demonstrating the usage of classes in Python.

**Class** in python is a blue print or a container, consist of functions and attriutes that describes the behavior of the object

**Instance** is a actual object contains a copy of class functions and attributes. We can use instance to access the attributes and call functions of the class.

**Example of Class and Instance:**

|  |
| --- |
| **class Book:**      def \_\_init\_\_(self, name, author, price):  **# Class attributes**          self.name = name          self.author = author          self.price = price      **# class functtion**      def get\_bookdetails(self):          return f"BookName: {self.name}, Author: {self.author}, price: {self.price}"  if \_\_name\_\_ == "\_\_main\_\_":  **# instance of a class**     book1 = Book("English Book", "James", 10)     book2 = Book("Maths Book", "Toney", 20)  # calling class function using **instance**    print(book1.get\_bookdetails())  print(book1.name) |

As we observe, we can access the functions and class attributes using the instance name.

Sometimes we may have to hide the class attributes and enable access them only using **get/set** methods to access and modify the attribute values. We can use ‘**\_\_**’ to achieve this in python.

**Example to make the attributes hide from access it directly**

|  |
| --- |
| class Book:      def \_\_init\_\_(self, name, author, price):          self.name = name          self.author = author  **self.\_\_price = price**    **def setprice(self, price):**  **self.\_\_price = price**    **def getprice(self):**  **return self.\_\_price**    if \_\_name\_\_ == "\_\_main\_\_":     book1 = Book("English Book", "James", 10)     book2 = Book("Maths Book", "Toney", 20)       print(f"BookName: {book1.name}, Author: {book1.author}, price: {book1.**getprice**()}")     book1.**setprice**(12)     print(f"BookName: {book1.name}, Author: {book1.author}, price: {book1.getprice()}")       print(f"BookName: {book2.name}, Author: {book2.author}, price: {book2.getprice()}") |

In the above example ‘price’ attribute made hidden, accessing it using the instance will throw **‘AttributeError’ (**AttributeError: 'Book' object has no attribute 'price'**).** We need to use getter and setter to get and modify the hidden attributes