Inheritance & Polymorphism in Python OOPAuthor: Madhu DadiTags: pythonSource: https://madhudadi.in/blog/pos

- The Customer class is initialized with a name, gender, and an Address object,demonstrating composition. - The A

Explanation- The Customer class initializes with a name, gender, and an Address object, allowingfor structured cus

- Defines a parent class User with an initializer that sets a name attribute and alogin method. - Introduces a child cl

- Constructor- Non Private Attributes- Non Private MethodsThis code defines a Phone class and a SmartPhone sul

[code block]Output[code block]Understanding Private Member Access in Inheritance with Python ClassesExplanati

Output[code block]This code snippet invokes a method to perform a check on an object or system.Explanation- Th

[code block]Demonstrating inheritance and encapsulation in Python classesExplanation- The Parent class initialize

[code block]Demonstrating inheritance and encapsulation in Python classesExplanation- The Parent class initialize

Explanation- The code defines two classes, A and B, where B inherits from A. - Class A has an initializer that sets a

- The code defines a base Phone class with a constructor that initializes price,brand, and camera attributes while p

Explanation- The code defines a base class Phone with an initializer that sets price, brand, andcamera attributes. -

Understanding Inheritance and Constructor Initialization in Python ClassesExplanation- The Phone class is defined

- The Parent class initializes a private attribute num and provides a method getnum()to access its value. - The Chil

with a value of 10. - The show method in the Child class calls the show method of the Parent class todisplay num,

smartphoneExplanation- The Phone class is defined with an initializer that sets the price, brand, andcamera specifi

specifications. - The buy method in the Phone class allows for simulating the purchase of a phone. - SmartPhone a

includes a method to simulate buying a phone. - The Product class has a method for customer reviews, which take

includes a method to simulate buying a phone.- The Product class contains a method for reviewing a product base

from B. - Class A has a method m1 that returns the integer 20. - Class B overrides the m1 method to return 30 and

- Class C inherits from B and further overrides the m1 method, adding 20 to the resultof B’s m1, yielding a final res

[code block]Understanding Method Overloading Behavior in Python ClassesExplanation- The Shape class defines

[code block]This code snippet demonstrates string concatenation using operator overloading inPython.Explanation

- The result of the operation is a new list that contains all elements from both listsin the order they were added. - Th