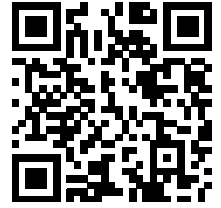


Object-Oriented Concepts



Object-oriented programming (OOP) is a programming _____ that uses "objects" to design applications and _____ programs. It utilizes several key concepts including _____, encapsulation, polymorphism, and abstraction. These _____ empower developers to create more flexible, modular, and reusable code. In OOP, an _____ is an instance of a class, which can combine both _____ and methods to perform operations. _____ allows a class to inherit properties and methods from another class, promoting code reuse and reducing _____. Encapsulation hides the internal state of an object from the outside world and restricts _____ to it, which can prevent the object's data from being directly modified. _____ enables objects of different classes to be treated as objects of a common super class, mainly for the purpose of allowing different _____ to be accessed through the same interface. Abstraction simplifies complex _____ by modeling classes appropriate to the problem, omitting the irrelevant details from the user. OOP languages, such as _____, C++, and Python, facilitate the implementation of real-world _____ in programming, making it easier to handle complex systems and maintain _____. The modular approach of OOP makes it particularly useful in large-scale software engineering _____, where maintaining and upgrading software can be highly challenging. It also supports better data _____ and modeling, as it is closely aligned with the real-world scenario of objects and interactions. The _____ development process benefits from OOP by improving code readability, scalability, and efficiency, making it a _____ of modern software engineering practices.

code Inheritance Java access software inheritance concepts computer
 data classes handling paradigm scenarios Polymorphism redundancy object
 reality projects cornerstone