

4 Lecture - CS506

Important Subjective

Certainly, here are 10 short subjective questions related to Object-Oriented Programming along with their answers:

****Question 1:**** What is the main goal of encapsulation in Object-Oriented Programming?

****Answer:**** Encapsulation aims to bundle data and methods that operate on the data into a single unit, known as a class. This concept provides data hiding and protection by restricting direct access to the data and allowing controlled interactions through methods.

****Question 2:**** Explain the concept of inheritance in Object-Oriented Programming.

****Answer:**** Inheritance allows a new class (subclass or derived class) to inherit properties and behaviors (attributes and methods) from an existing class (superclass or base class). It promotes code reuse and hierarchy by enabling a subclass to extend or override the functionality of the superclass.

****Question 3:**** What is method overloading in OOP?

****Answer:**** Method overloading refers to the ability to define multiple methods in a class with the same name but different parameter lists. The methods must have distinct parameter types or a different number of parameters. This allows for more flexible and intuitive method naming within a class.

****Question 4:**** How does polymorphism enhance code flexibility in OOP?

****Answer:**** Polymorphism allows objects of different classes to be treated as instances of a common superclass, facilitating dynamic method invocation. This enables code to be more adaptable and generic, as the same method name can be used for different implementations across various subclasses.

****Question 5:**** What is a constructor in OOP, and why is it important?

****Answer:**** A constructor is a special method that is used to initialize objects when they are created. It has the same name as the class and does not have a return type. Constructors ensure that the object is properly initialized before it is used and allow for setting initial values to object attributes.

****Question 6:**** Describe the concept of abstraction in Object-Oriented Programming.

****Answer:**** Abstraction involves presenting the essential features of an object while hiding its implementation details. It allows programmers to focus on what an object does rather than how it does it. Abstract classes and interfaces are used to achieve abstraction by defining a common set of methods that subclasses must implement.

****Question 7:**** What is the difference between instance variables and class variables?

****Answer:**** Instance variables (also called instance fields) belong to specific instances of a class and have different values for each instance. Class variables (also called static variables) are shared across all instances of a class and have the same value for all instances.

****Question 8:**** How does the "this" keyword work in OOP languages like Java?

****Answer:**** The "this" keyword refers to the current instance of the class. It is used to differentiate between instance variables and method parameters when they share the same name. "this" helps avoid ambiguity and ensures that the correct variable is accessed.

****Question 9:**** What is method overriding in OOP?

****Answer:**** Method overriding occurs when a subclass provides a specific implementation for a method that is already defined in its superclass. The method in the subclass must have the same name, return type, and parameters as the method in the superclass. This allows a subclass to customize or extend the behavior of the inherited method.

****Question 10:**** Explain the concept of composition in OOP.

****Answer:**** Composition involves building complex objects by combining simpler objects. It allows a class to have references to other objects as instance variables. This promotes reusability and

modularity, as changes to the composed objects don't directly affect the containing class.