

Reading Assignment

1. What are the advantages of Polymorphism?

Polymorphism is a fundamental concept in OOP that allows objects of different types to be treated as objects of a common type. This provides several advantages:

- It enhances code reusability and readability by allowing a single interface to be used to represent different types.
- It provides a way to implement dynamic method dispatch, which is a mechanism by which a call to an overridden method is resolved at runtime, rather than compile-time. This makes your code more flexible and capable of handling new types that are added later.

2. How is Inheritance useful to achieve Polymorphism in Java?

- Inheritance is another key concept in OOP that allows one class to inherit the properties (methods and fields) of another class.
- In Java, polymorphism is achieved through inheritance. A superclass reference variable can refer to a subclass object. This allows Java to select the appropriate method to call at runtime based on the actual object referred to by the reference variable, not the type of the reference variable. This is known as dynamic method dispatch and is a key aspect of polymorphism.

3. What are the differences between Polymorphism and Inheritance in Java?

- While both polymorphism and inheritance are fundamental concepts in OOP, they serve different purposes.
- + Inheritance is a mechanism that allows one class to acquire the properties and behaviors of another class, promoting code reusability and organization.
- + On the other hand, polymorphism is a concept that allows objects of different types to be treated as objects of a common type, enhancing flexibility and extensibility.
- In Java, inheritance is a prerequisite for polymorphism, but they are not the same thing.