Reading Assignment

1. What are the benefits of Polymorphism?

Enhances code reusability by treating objects from different classes as objects of a common class.
Improves code readability and maintainability by reducing the amount of code that needs to be written and managed.
Supports dynamic binding, enabling the runtime invocation of the correct method based on the actual class of the object.
Facilitates treating objects as a single type, simplifying the creation of generic code capable of handling various object types.

2. How does Inheritance contribute to achieving Polymorphism in Java?

Polymorphism in Java involves performing a single action in different ways.
Inheritance is crucial for achieving Polymorphism in Java because it allows one class to acquire the properties and attributes of a parent class.
Java Inheritance enables the use of inherited properties to perform diverse tasks, thereby achieving the same action in multiple ways.

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

Polymorphism and inheritance are separate concepts, albeit interconnected.
Polymorphism utilizes the same name or interface for different actions or operations, while inheritance establishes new classes based on existing ones, inheriting their attributes and methods.
Polymorphism can be attained through method overloading or overriding, whereas inheritance is implemented using the extends or implements keywords.
Polymorphism can be static or dynamic, while inheritance can be single or multiple (excluding Java, which uses interfaces).
Polymorphism results in more concise and expressive code, while inheritance allows for the creation of a class hierarchy to organize code effectively.