

Reading Assignment

1. What are the advantages of Polymorphism?

- Đa hình cho phép một giao diện được sử dụng cho nhiều loại đối tượng, giúp mã linh hoạt và dễ mở rộng.
- Làm mã gọn gàng, tránh các câu lệnh if-else hoặc switch dài dòng để xử lý theo loại đối tượng.
- Giảm sự trùng lặp mã, tăng khả năng tái sử dụng bằng cách viết mã chung cho lớp cha, áp dụng được cho tất cả lớp con.
- (ví dụ: gọi toString() trên nhiều loại Media khác nhau).
- Dễ bảo trì và nâng cấp hệ thống (thêm loại Media mới mà không cần sửa mã hiện có).

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

- Kế thừa là nền tảng để đạt được tính đa hình: Trong Java, bạn chỉ có thể gọi các phương thức đa hình nếu lớp con kế thừa từ một lớp cha hoặc implement một interface.
- Khi các lớp con kế thừa cùng một lớp cha, ta có thể gán các đối tượng con vào biến kiểu lớp cha.
`Media m1 = new DigitalVideoDisc(...);`
`Media m2 = new Book(...);`
- Khi gọi phương thức như `m1.toString()` hay `m1.play()`, Java sẽ tự động gọi đúng phương thức tương ứng của đối tượng thực tế (DVD, Book, CD...) – đó chính là đa hình động (runtime polymorphism).

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

Đặc điểm	Kế thừa (Inheritance)	Đa hình (Polymorphism)
Định nghĩa	Cơ chế để một lớp kế thừa hành vi và thuộc tính từ lớp khác.	Khả năng các lớp khác nhau được xử lý như thể thuộc cùng một lớp cha qua giao diện chung.
Mục đích chính	Tái sử dụng mã; thiết lập mối quan hệ.	Linh hoạt mã; hành vi động tại runtime.
Mối quan hệ	Mối quan hệ giữa các lớp (cha – con)	Mối quan hệ giữa đối tượng và hành vi khi thực thi
Loại	Là khái niệm cấu trúc.	Là khái niệm hành vi.
Phụ thuộc	Kế thừa có thể tồn tại mà không cần đa hình.	Đa hình phụ thuộc vào kế thừa.
Thời điểm hoạt động	Chủ yếu ở thời điểm biên dịch	Thường xảy ra lúc chạy (runtime)