

READING ASSIGNMENT

Câu 1: What are the advantages of Polymorphism?

Tính đa hình mang lại nhiều lợi ích quan trọng trong lập trình hướng đối tượng:

- Tái sử dụng mã: Giúp viết mã tổng quát để xử lý các đối tượng khác nhau mà vẫn dùng chung một cách gọi hàm.
- Mở rộng dễ dàng: Khi thêm lớp mới, không cần sửa đổi code cũ — chỉ cần đảm bảo lớp mới tuân theo interface hoặc kế thừa từ lớp cha.
- Tăng tính linh hoạt: Dễ dàng thay đổi cách triển khai của một phương thức mà không ảnh hưởng đến phần còn lại của chương trình.
- Tối ưu hóa cấu trúc chương trình: Giảm sự phụ thuộc lẫn nhau giữa các thành phần.

Câu 2: How is Inheritance useful to achieve Polymorphism in Java?

Kế thừa là cơ sở để thực hiện tính đa hình.

Trong Java, một lớp con kế thừa từ lớp cha có thể ghi đè (override) các phương thức của lớp cha.

Khi ta gọi phương thức từ đối tượng kiểu lớp cha nhưng thực tế đối tượng là lớp con, phương thức của lớp con sẽ được gọi — đây chính là đa hình động (runtime polymorphism).

Ví dụ:

```
Media m = new DigitalVideoDisc();
```

```
m.play(); // Gọi phương thức play() của DigitalVideoDisc nếu được override
```

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

Tiêu chí	Kế thừa (Inheritance)	Đa hình (Polymorphism)
Định nghĩa	Cơ chế một lớp kế thừa thuộc tính/phương thức từ lớp khác.	Cơ chế cho phép một phương thức có nhiều cách triển khai.
Mục đích chính	Tái sử dụng code, mở rộng lớp.	Cho phép gọi phương thức chung cho nhiều đối tượng khác nhau.

Quan hệ giữa các lớp	Quan hệ cha – con (is-a).	Quan hệ giao tiếp qua interface hoặc lớp cha.
Thực hiện	Dùng từ khóa extends, implements.	Thực hiện qua ghi đè (@Override) và interface.
Thời điểm thực hiện	Biên dịch (compile-time).	Chủ yếu là khi chạy (runtime).