1. **What are the advantages of Polymorphism?**
   * **Tăng tính linh hoạt:**Cho phép sử dụng cùng một giao diện cho các hành vi khác nhau.
   * **Giảm thiểu trùng lặp mã:** Một phương thức có thể được sử dụng cho nhiều đối tượng khác nhau, giúp mã nguồn trở nên gọn gàng và dễ bảo trì hơn.
   * **Dễ dàng mở rộng và bảo trì:** Các lớp mới có thể được thêm vào hệ thống mà không cần thay đổi mã hiện tại, miễn là chúng tuân theo giao diện đã định nghĩa trước đó.
   * **Tái sử dụng mã:** Tăng khả năng tái sử dụng mã nguồn thông qua các phương thức và lớp cơ sở.
2. **How is Inheritance useful to achieve Polymorphism in Java?**

Kế thừa là nền tảng để thực thi tính đa hình trong Java:

* Lớp con kế thừa hưởng những thuộc tính từ lớp cha, từ đó có thể ghi đè (Overide) các phương thức và cung cấp các hành vi khác nhau tùy vào kiểu đối tượng.
* Cho phép tham chiểu đến 1 lớp cha (hoặc interface), ta có thể gọi các phương thức của lớp con.
* Cơ chế liên kết động (dynamic binding) tại thời điểm chạy đảm bảo rằng phương thức bị ghi đè trong lớp con sẽ được gọi, dù biến tham chiếu có kiểu lớp cha.

1. **What are the differences between Polymorphism and Inheritance in Java**

|  |  |  |
| --- | --- | --- |
| **Tiêu chí** | **Kế thừa** | **Đa hình** |
| Định nghĩa | Một lớp kế thừa thuộc tính và phương thức lớp khác | Một phương thức có nhiều cách thực thi khác nhau |
| Mục đích | Tái sử dụng mã, giảm trùng lặp, tổ chức hệ thống phân cấp | Tăng tính linh hoạt khi xử lý đối tượng, tập trung vào hành vi tại runtime |
| Cơ chế | Lớp con tự động có thuộc tính/phương thức lớp cha | Gọi đúng phương thức của đối tượng thực tế tại runtime |