Bài 6

|  |  |
| --- | --- |
| 1. Kế thừa là gì? Mục đích? | * Kế thừa (Inheritance) là cơ chế cho phép một lớp (class) con kế thừa các thuộc tính và phương thức từ lớp cha. * Mục đích: Tái sử dụng mã nguồn, tăng tính mở rộng và bảo trì. |
| 1. Đặc điểm kế thừa trong Java: | * Java hỗ trợ kế thừa đơn (một lớp chỉ kế thừa từ một lớp cha). * Sử dụng từ khóa extends để kế thừa. * Lớp con có thể ghi đè phương thức của lớp cha. * Constructor không được kế thừa. |
| 1. Vì sao Java không hỗ trợ đa kế thừa? | * Để tránh xung đột phương thức (diamond problem). * Thay vào đó, Java hỗ trợ đa kế thừa thông qua interface. |
| 1. Từ khóa super, final dùng để làm gì? | * super: Gọi constructor hoặc phương thức của lớp cha. * final:   + Với biến: không thay đổi được giá trị.  + Với phương thức: không cho ghi đè.  + Với lớp: không được kế thừa. |
| 1. class Object là gì? Một số method cơ bản của Object? | * Object là lớp cha của tất cả các lớp trong Java. * Một số phương thức phổ biến:   + toString()  + equals(Object obj)  + hashCode()  + clone()  + getClass()  + finalize() |
| 1. Tính đa hình là gì? Thể hiện ở đâu trong Java? | * Đa hình là khả năng đối tượng có thể hành xử theo nhiều cách khác nhau. * Thể hiện qua:   + Ghi đè phương thức (Overriding)  + Gọi phương thức thông qua biến tham chiếu lớp cha |
| 1. Overriding là gì? Điều kiện để có Overriding? | * Ghi đè phương thức lớp cha trong lớp con. * Điều kiện:   + Cùng tên, kiểu trả về, tham số  + Không hạ cấp phạm vi truy cập  + Không ghi đè phương thức final, static, private |
| 1. Phân biệt Overloading và Overriding? | * Overloading   + Cùng tên, khác tham số  + Trong cùng một lớp  + Xảy ra lúc biên dịch  + Có thể khác kiểu trả về   * Overriding   + Cùng tên, cùng tham số  + Giữa lớp cha và lớp con  + Xảy ra lúc chạy chương trình  + Phải cùng kiểu trả về |
| 1. Ép kiểu là gì? Vì sao phải ép kiểu? | * Ép kiểu là chuyển kiểu dữ liệu này sang kiểu khác. * Cần thiết khi muốn:   + Chuyển giữa kiểu dữ liệu nguyên thủy  + Gán đối tượng kiểu con cho biến kiểu cha (hoặc ngược lại) |
| 10.Lưu ý khi ép kiểu | * Ép kiểu xuống (downcasting) cần chắc chắn kiểu thực sự của đối tượng. * Có thể gây lỗi ClassCastException nếu ép sai. * Dùng instanceof để kiểm tra trước khi ép. |