# Extension

Mặt khác, khi sử dụng mở rộng, chúng ta tận dung(utilize) kế thừ code từ code được viết cho lớp cha, nhưng chúng ta sẽ them những method mà không có không lớp cha, để mở rộng chứ năng của nó. Ví dụ, như Dog class, chúng ta tạo ra lớp con, BorderCollie, thừ kế những phương thức của lớp cha, nhưng thêm 1 số phương thức mới, BorderCollie có khả năng chăn thả gia súc(herding instinct) mà trong lớp cha không có phương thức này. Bằng cách thêm phương thức, chúng ta đang mở rộng các chức năng của lớp Dog.

Trong java, mỗi class có thêm mở rộng chính xác từ 1 lớp khác. Kể cả khi class không làm rõ rang việc sử dụng **Extends**, nó vẫn thừa kế chính xác từ lớp khác, trong trường hợp này là java.lang.Object. Bởi vì thuộc tính của nó, Java nói nó chỉ cho kế thừa 1 lớp từ các lớp.

# Types of Method loading

Trong khai báo 1 lớp mới, Java sử dụng 2 loại của phương thức overriding, **refinement** (sàng lọc) và **replacement** (thay thế). Loại thay thế, phương thức hoàn toàn thay thế phương thức của lớp cha mà nó overriding. Trong java, tất cả các phương thức của class sử dụng đặc điểm của overriding.

Trong loại sàng lọc, phương thức của lớp con không thay thế các phương thức của lớp cha, nhưng nó sẽ thêm code thêm vào các phương thức của lớp con. Trong java, phương thức khởi tạo sử dụng sự sàng lọc của overriding, sự sắp xếp này đươc gọi là khởi tạo chuỗi. Cụ thể, khởi tạo bắt đầu sự thực thi của nó bang cách gọi khỏi tạo của lớp cha. Điều này có thể làm rõ rang hoặc ngầm hiểu **(implicitly).** Để gọi phương thức khỏi tạo của lớp cha 1 cách rõ rang, chúng ta có thể sử dụng từ khóa super đê tham chiếu tới lớp cha. Nếu không có lệnh nào được thực thi trong phần khởi tạo. Tuy nhiên, trình biên dịch sẽ tự động tạo, ở dòng đầu tiên của khỏi tạo, gọi lênh super(). Tổng quan, trong java, khởi tạo sử dụng sàng lọc trong khi các phương thức thường sử dung replacement.

# Key word this

Trong 1 vài trường hợp, trong 1 lớp java, nó thuận tiện cho việc tham chiều tới thể hiện của lớp nó. Java cung cấp từ khóa, gọi là **This,** cho việc tham chiếu. **This** thật hữu dụng, ví dụ, nếu chúng ta muốn truyền đối tượng như là các tham số tới 1 số phương thức. Các ứng dụng của **This** là tham chiếu mảng trong 1 đối tượng mà có tên xung đột với các lớp hiện tại.