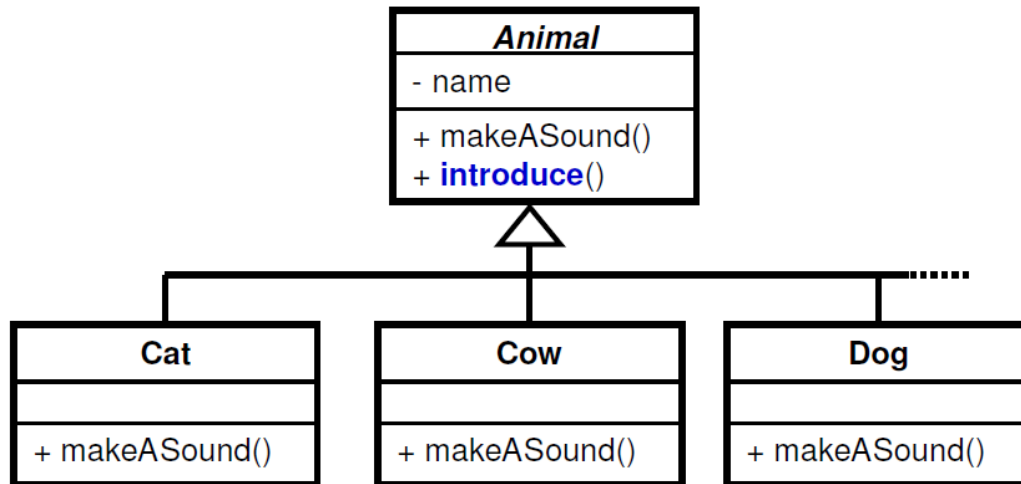


Bài thực hành số 08 – Ôn tập về kế thừa và đa hình

Cho sơ đồ như hình vẽ sau:



1. Cài đặt các lớp động vật **Animal**, **Cow**, **Cat**, **Dog** như mô tả trên. Trong đó lớp **Animal** là lớp trừu tượng. Khi gọi **introduce()** đối với một con vật, nó sẽ kêu một tiếng theo đúng kiểu của loài mình (**makeASound**) và tự giới thiệu tên (**name**). Viết lớp **AnimalTest** để thử nghiệm chức năng các lớp đã tạo (chú ý, tiếng kêu chỉ là in ra chữ theo tiếng kêu đó).
2. Chèn các lệnh **System.out.print()** vào **constructor** của các lớp trên để thông báo khi constructor của một lớp nào đó được gọi. Thử nghiệm tại **AnimalTest** để quan sát trình tự khởi tạo một đối tượng thuộc các lớp con Cat, Cow, Dog.
3. Bổ sung thuộc tính **population** (dân số) cho mỗi lớp để đếm số lượng các thể của mỗi loài. Giá trị này được khởi tạo bằng 0 và được tăng thêm 1 mỗi khi một đối tượng được tạo (tại constructor). Thuộc tính **population** của **Animal** cần lưu tổng số các con vật thuộc các loài khác nhau. Bổ sung (các) phương thức **getPopulation()** để trả về giá trị của **population** của từng loài. Lưu ý khi nào thì cần dùng loại static, khi nào có thể hoặc không thể kế thừa. Sửa lớp **AnimalTest** để thử thuộc tính **population**.
4. Bổ sung vào lớp **Animal** phương thức **giveBirth()** cho động vật con được sinh ra. Khi gọi phương thức này cho một đối tượng động vật (Cow/Cat/Dog) thì kết quả trả về là một đối tượng cùng loài với tên (**name**) của động vật mới sinh sẽ là tên của con mẹ cộng thêm **"s baby"** vào sau. Ví dụ con của Tom sẽ có tên là Tom's baby.
5. Thử sử dụng phương thức **finalize** xem có thể dùng nó để giảm số đếm **population** khi một con vật được cho là "chết" được hay không? Có cách nào khác để làm việc này. Viết các lớp và lớp **AnimalTest** để thử nghiệm.