Dò bài cũ

1. Static là gì? Ràng buộc khi sử dụng static

2. Các loại biến

Static là gì

Ràng buộc

- Là từ khóa dùng để khai báo thuộc tính và phương thức thuộc về class chứ không phải của đối tượng => Có thể truy cập các phương thức và biến static mà không cần tạo đối tượng.

- Truy xuất thành phần static (thuộc tính, phương thức) thông qua tên class, hoặc có thể thông qua đối tượng.

- Mục đính
  - Định nghĩa thuộc tính, phương thức chung của toàn bộ đối tượng
  - Tạo ra các lớp tiện ích.

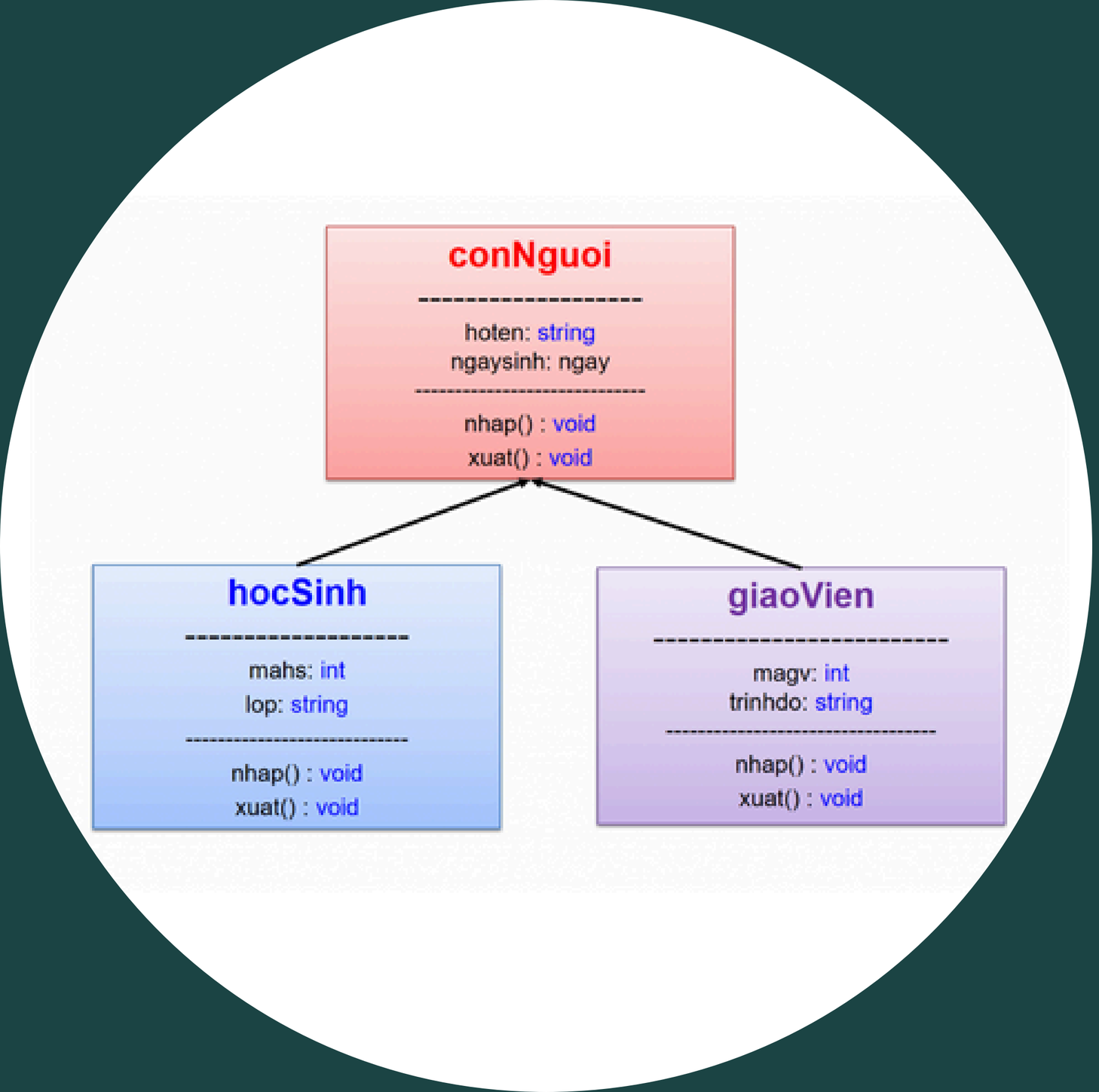
Ràng buộc khi sử dụng static

- Phương thức static
  - Chỉ có thể gọi phương thức và thuộc tính static.
  - Không thể gọi phương thức và thuộc tính non-static.
  - Không thể sử dụng this hoặc super
- Phương thức non-static
  - Có thể gọi phương thức và thuộc tính static.
- Có thể khởi tạo giá trị biến static thông qua khối static.

Các loại biến

- Biến toàn cục (global, instance, đối tượng)
  - Khai báo trong một class, bên ngoài constructor, method và block {}

- Biến cục bộ (local)
  - Khai báo bên trong constructor, method hoặc block {}
-
- Biến static
  - Được khai báo bằng từ khóa static


conNguoi
hoten: string
ngaysinh: ngay
nhap() : void
xuat() : void
hocSinh
mahs: int
lop: string
nhap() : void
xuat() : void
giaoVien
magv: int
trinhdo: string
nhap() : void
xuat() : void

Kế thừa

NỘI DUNG CHÍNH

1. Kế thừa là gì
2. Mối quan hệ và đặc điểm
3. Overload & Override
4. Từ khóa super
5. Từ khóa final

Kế thừa

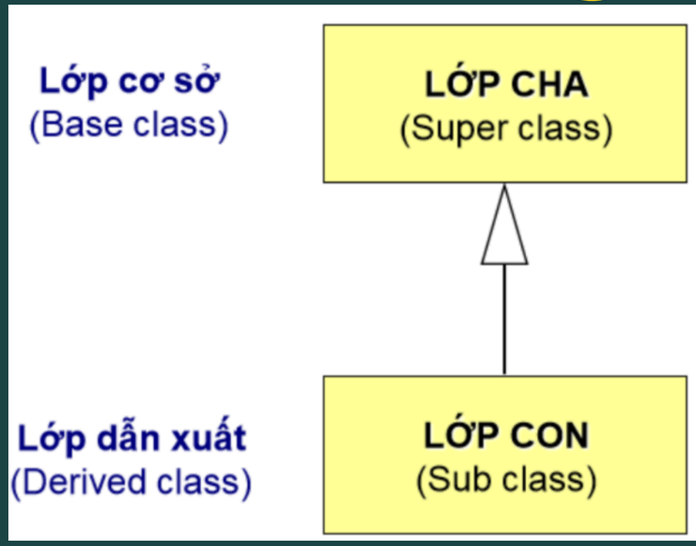
- **Là cơ chế cho phép lớp con có thể sử dụng lại các đặc điểm và hành vi đã được định nghĩa ở lớp cha.**

- Mục đích
  - Tái sử dụng mã nguồn

Mối quan hệ và đặc điểm

is-a
has-a

- Mối quan hệ giữa lớp con và lớp cha là : is – a
  - Lớp cha (super class, parent class, base class)
  - Lớp con (subclass, child class, derived class)



- Đặc điểm
  - Lớp con không thể kế thừa constructor của lớp cha
  - Lớp con không thể truy cập tài sản private của lớp cha
  - Java không hỗ trợ đa kế thừa.

is-a & has-a

- Mối quan hệ is-a
  - Được xác định thông qua từ khóa extends
  - Cat extends Animal
    - Cat is-an Animal
    - Trường hợp sai: Animal is an Cat

- Mối quan hệ has-a
  - Được xác định thông qua sự sở hữu.

```
class Person {
    Heart heart = new Heart();
    // code
}
```

    - Person has-a Heart
    - Trường hợp sai: Heart has-a Person

Overloading & Overriding

- Overloading - Nạp chồng
  - Một lớp có nhiều phương thức cùng tên nhưng khác nhau về số lượng và kiểu dữ liệu của tham số
  - Làm tăng khả năng đọc hiểu của chương trình
  - Xảy ra trong cùng một class

- Overriding - Ghi đè
  - Lớp con có phương thức cùng tên, cùng tham số, cùng kiểu trả về với lớp cha.
  - Định nghĩa lại phương thức của cha sao cho phù hợp với phương thức của con.
  - Xảy ra trong quan hệ kế thừa

Super

- Từ khóa super đại diện cho đối tượng của lớp cha
- Dùng để gọi constructor, phương thức, thuộc tính của lớp cha

Final

- Được sử dụng trong rất nhiều ngữ cảnh
  - Đi với biến
    - Hằng số
  - Đi với phương thức
    - Không cho ghi đè
  - Đi với class
    - Không cho kế thừa (Vô sinh)

DATABASE

THANK YOU