



Kế thừa

Giới thiệu

- Kế thừa
 - Khả năng tái sử dụng
 - Tạo class mới từ class đã có
 - kế thừa thuộc tính và phương thức đã có
 - cải tiến bằng phương thức và thuộc tính mới
 - Class dẫn xuất kế thừa từ class cơ sở
 - Class dẫn xuất
 - Nhóm các đối tượng chi tiết hơn
 - phương thức kế thừa từ lớp cơ sở
 - có thể chỉnh sửa
 - Thêm phương thức

Giới thiệu

- Phân cấp lớp
 - Lớp cơ sở trực tiếp
 - Lớp cơ sở gián tiếp
 - Kế thừa đơn
 - kế thừa từ một lớp cơ sở
 - Kế thừa kép
 - kế thừa từ nhiều lớp cơ sở
 - các lớp cơ sở có thể không liên quan tới nhau

3 loại kế thừa

- 3 loại kế thừa

- **public**

- Mọi đối tượng của lớp cũng là đối tượng của lớp cơ sở

- đối tượng của lớp cơ sở không phải là đối tượng của lớp dẫn xuất

- Ví dụ: Ô tô là xe nhưng không phải tất cả các xe đều là ô tô

- **private**

- **protected**

3 loại kế thừa

Kiểu dẫn xuất

Tính chất ở lớp cơ sở

Tính chất ở lớp dẫn xuất

private

**private
protected
public**

**Không truy nhập được
private
private**

protected

**private
protected
public**

**Không truy nhập được
protected
protected**

public

**private
protected
public**

**Không truy nhập được
protected
public**

Cú pháp

```
class <Tên lớp dẫn xuất>: <Từ khóa dẫn xuất> <Tên  
lớp cơ sở> {  
... // Khai báo các thành phần lớp };
```

Trong đó:

Tên lớp dẫn xuất: là tên lớp được cho kế thừa từ lớp khác.

Tên lớp cơ sở: là tên lớp đã được định nghĩa trước đó để cho lớp khác kế thừa.

Từ khóa dẫn xuất: là từ khóa quy định tính chất của sự kế thừa. Có ba từ khóa dẫn xuất là `private`, `protected` và `public`.

Trừu tượng hoá

- Trừu tượng hoá
 - Tập trung vào những tính chất chung của các đối tượng trong hệ thống
- “is-a” so với “has-a”
 - “is-a”
 - kế thừa
 - Lớp dẫn xuất có thể coi là lớp cơ sở
 - Ví dụ: Ô tô *is a* xe
 - Tính chất/hành động (Thuộc tính/phương thức) của xe cũng là tính chất/hành động của ô tô
 - “has-a”
 - thành phần
 - ĐỐI tượng chứa một HAY NHIỀU đối tượng của lớp khác
 - Ví dụ: Car *has a* bánh lái

Lớp cơ sở và lớp dẫn xuất

Base Classes and Derived Classes

- Lớp cơ sở và lớp dẫn xuất
 - Đối tượng của một lớp “is an” đối tượng của lớp khác
 - Ví dụ: Hình chữ nhật là tứ giác
 - Class **Rectangle** kế thừa từ class **Quadrilateral**
 - **Quadrilateral**: lớp cơ sở
 - **Rectangle**: lớp dẫn xuất
 - Lớp cơ sở biểu diễn tập đối tượng lớn hơn tập đối tượng của lớp dẫn xuất
 - Ví dụ:
 - Lớp cơ sở: **Xe**
 - Ô tô, xe máy, xe đạp, ...
 - Lớp dẫn xuất: **Car**
 - Nhỏ hơn, tập con được mô tả chi tiết hơn của xe

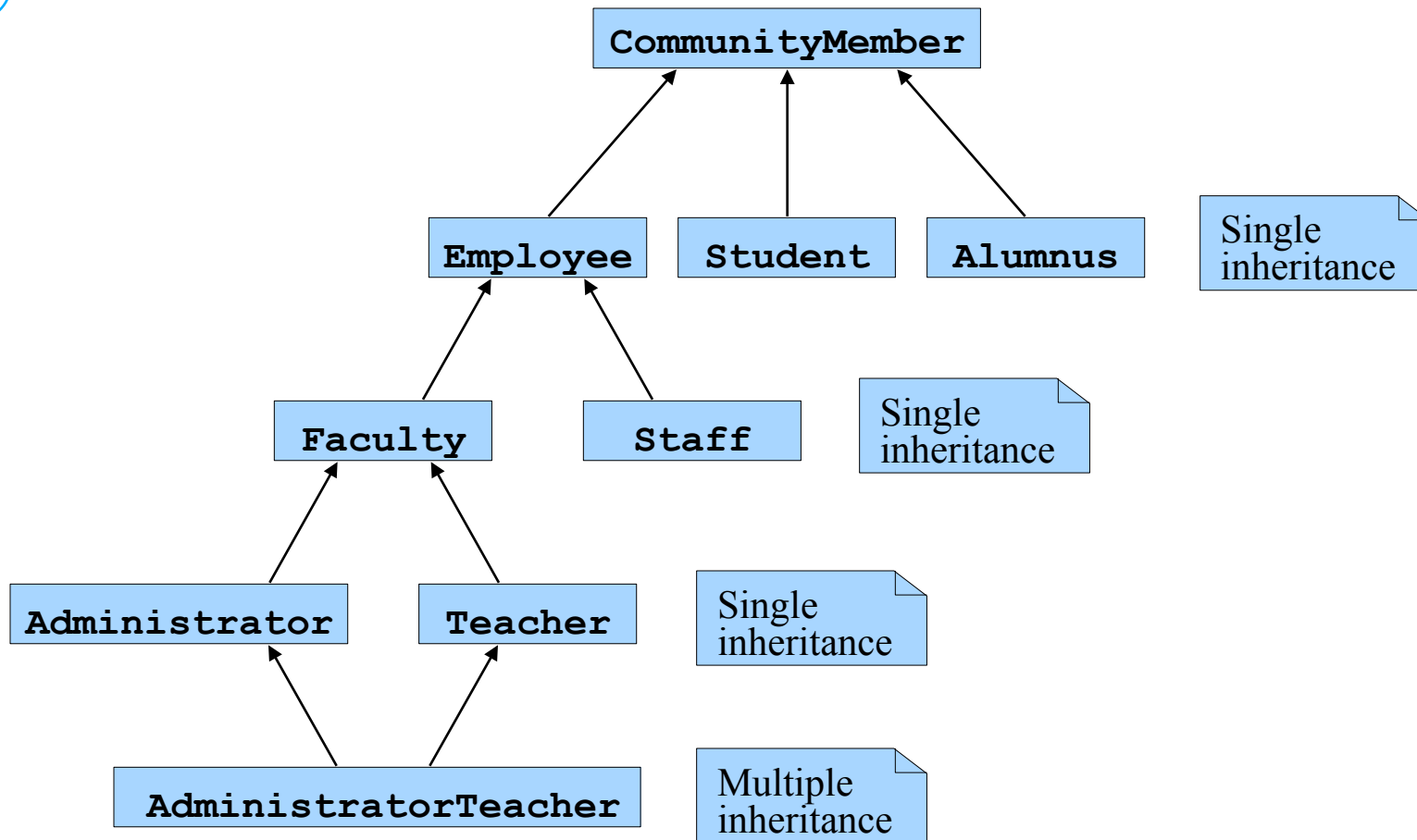
Ví dụ về kế thừa

Base class	Derived classes
Student	GraduateStudent UndergraduateStudent
Shape	Circle Triangle Rectangle
Loan	CarLoan HomeImprovementLoan MortgageLoan
Employee	FacultyMember StaffMember
Account	CheckingAccount SavingsAccount

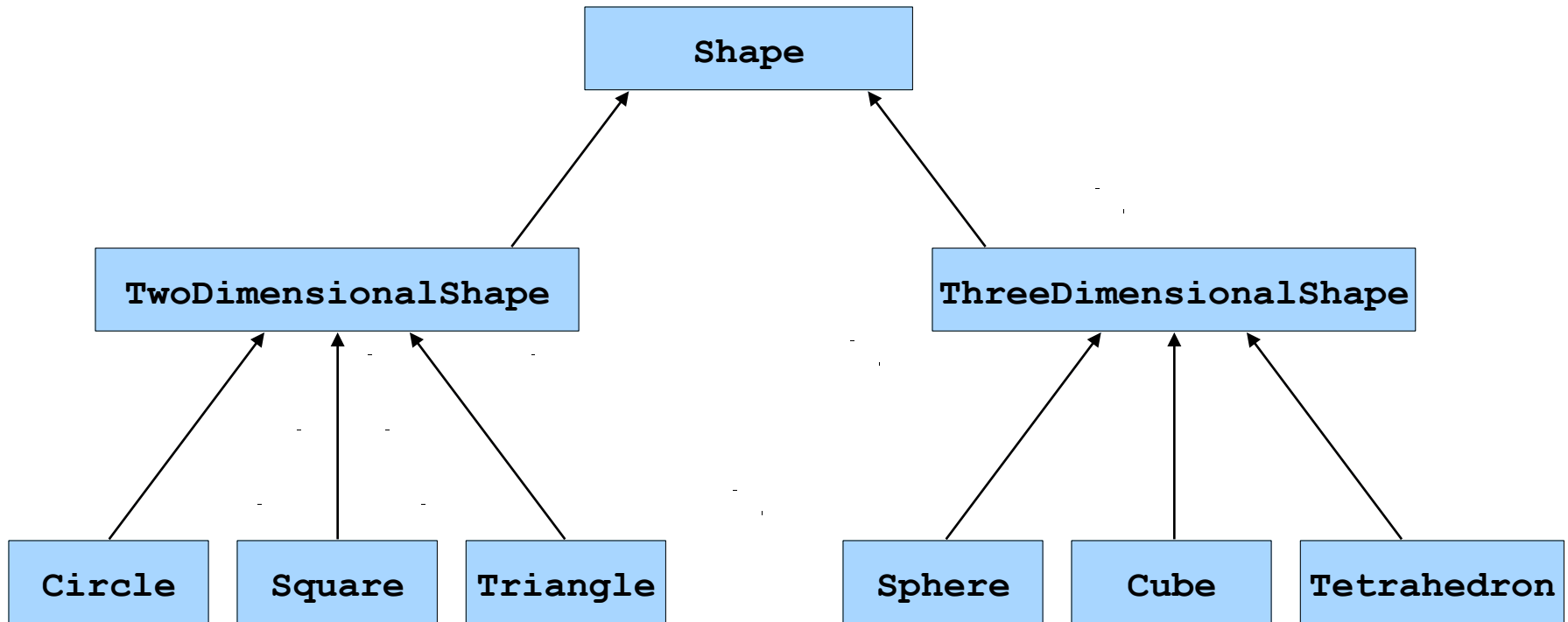
Lớp cơ sở và lớp dẫn xuất

- Sơ đồ phân cấp kế thừa
 - Quan hệ kế thừa: giống cấu trúc cây
 - Mỗi lớp trở thành
 - Lớp cơ sở
 - Cung cấp thuộc tính/phương thức cho các lớp khác
- HOẶC
- Lớp dẫn xuất
 - kế thừa thuộc tính/phương thức từ các lớp khác

Sơ đồ phân cấp kế thừa của đại học **CommunityMembers**.



Sơ đồ phân cấp kế thừa của Shape



Lớp cơ sở và lớp dẫn xuất

kế thừa **public**

– Ví dụ:

Class TwoDimensionalShape : public Shape

- Class **TwoDimensionalShape** kế thừa từ class **Shape**

– Thành phần **private** của lớp cơ sở

- Không truy cập trực tiếp được
- Vẫn được kế thừa

– Truy cập thông qua các phương thức được kế thừa

– Thành phần **public** và **protected** của lớp cơ sở

- Được kế thừa với quyền truy cập như lớp cơ sở

– **Hàm friend**

- Không kế thừa

Quan hệ giữa lớp cơ sở và lớp dẫn xuất

- Quan hệ giữa lớp cơ sở và lớp dẫn xuất
 - Ví dụ: Điểm/Hình tròn
 - Điểm
 - Tọa độ x-y, hàm tạo, hiển thị, nhập
 - Hình tròn
 - Tâm Tọa độ x-y kế thừa điểm
 - Bán kính
 - hiển thị: hiển thị tâm (gọi lại hiển thị điểm), hiển thị bán kính
 - Nhập
 - tính chu vi
 - tính diện tích
 - Hình tam giác:
 - tọa độ 3 đỉnh
 - tính chu vi, diện tích

Quan hệ giữa lớp cơ sở và lớp dẫn xuất

- Dùng thuộc tính **protected**

- Ưu

- Lớp dẫn xuất có thể thay đổi thuộc tính trực tiếp
 - Tăng hiệu năng đôi chút
 - không phải gọi hàm get/set

- Nhược

- Không kiểm tra hợp lệ
 - lớp dẫn xuất có thể gán giá trị không hợp lệ
 - Cài đặt bị phụ thuộc
 - phương thức của lớp dẫn xuất phụ thuộc vào cài đặt của lớp cơ sở
 - thay đổi ở lớp cơ sở có thể dẫn tới phải chỉnh sửa lớp dẫn xuất
 - » Phần mềm dễ lỗi

Ví dụ: sơ đồ kế thừa 3 mức

- điểm/hình tròn/ hình trụ (point/circle/cylinder)
 - Point
 - toạ độ x-y
 - Circle
 - toạ độ x-y
 - bán kính
 - Cylinder
 - toạ độ x-y
 - bán kính
 - chiều cao

(xem code mẫu đã gửi [KiemTraKeThua.zip](#))

Hàm tạo và hàm huỷ ở lớp dẫn xuất

- Khởi tạo đối tượng ở lớp dẫn xuất
 - Chuỗi gọi hàm tạo
 - Hàm tạo lớp dẫn xuất gọi tới hàm tạo lớp cơ sở
 - ẩn hoặc trực tiếp
 - Hàm tạo của Lớp cơ sở
 - được gọi cuối cùng
 - thân hàm tạo kết thúc thực hiện đầu tiên
 - Ví dụ: **Point3/Circle4/Cylinder**
 - **Point3** constructor được gọi cuối cùng
 - **Point3** constructor body kết thúc thực hiện đầu tiên
 - Khởi tạo thuộc tính
 - Mỗi hàm tạo của lớp cơ sở khởi tạo thuộc tính của nó
 - Được kế thừa bởi lớp dẫn xuất

Hàm tạo và hàm huỷ ở lớp dẫn xuất

- Hàm huỷ
 - Chuỗi lời gọi hàm huỷ
 - Ngược với hàm tạo
 - Hàm huỷ của lớp dẫn xuất được gọi trước
 - Hàm huỷ của lớp cơ sở kế được gọi tiếp
 - Cho tới khi gọi tới hàm huỷ của lớp cơ sở đầu tiên
 - Sau khi hàm huỷ của lớp cơ sở đầu tiên được gọi, đối tượng sẽ bị xoá khỏi bộ nhớ

Hàm tạo và hàm huỷ ở lớp dẫn xuất

- Hàm tạo, hàm huỷ, phép gán của lớp cơ sở
 - Không được kế thừa
 - Hàm tạo của lớp dẫn xuất, phép gán có thể gọi
 - Hàm tạo của lớp cơ sở
 - Phép gán