

CHƯƠNG 2

TỔNG QUAN VỀ

HƯỚNG ĐỐI TƯỢNG

GV biên soạn: ThS. Trần Kim Hương

NỘI DUNG



Các tính chất cơ bản của HĐT.....●



Các khái niệm cơ bản của HĐT.....●



Sức mạnh của HĐT.....●

1. Các tính chất cơ bản của HĐT



Hướng Đối Tượng

Trừu tượng hóa

Abstraction

Tính đóng gói

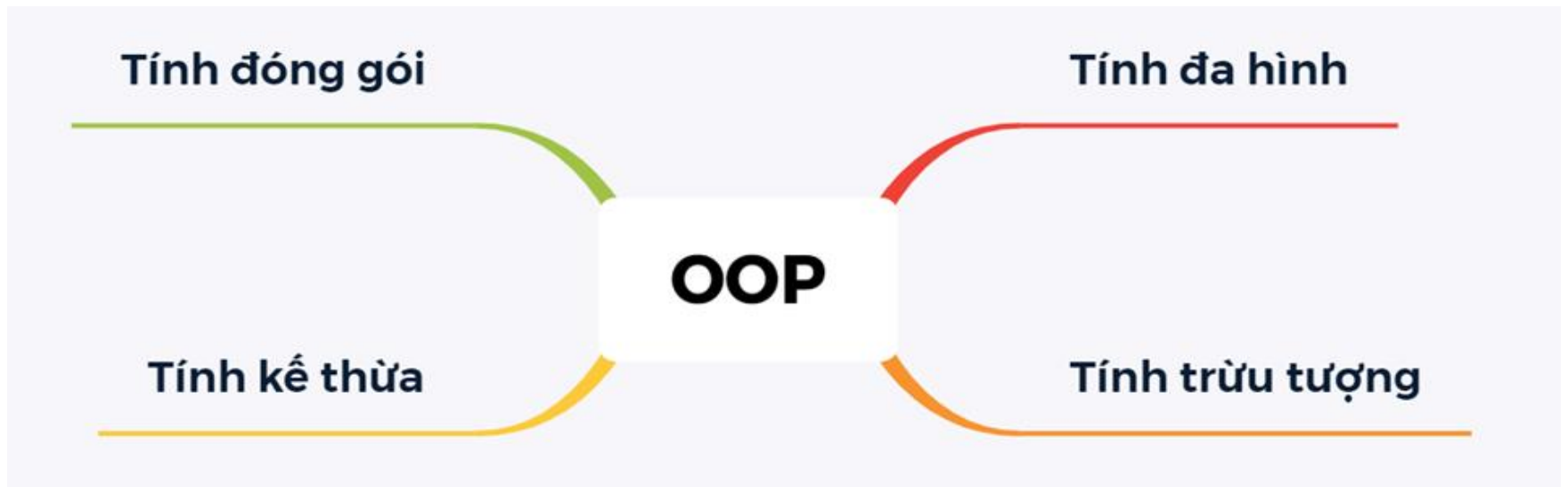
Encapsulation

Tính đa hình

Polymorphism

Tính kế thừa

Inheritance



1. Các tính chất cơ bản của HĐT



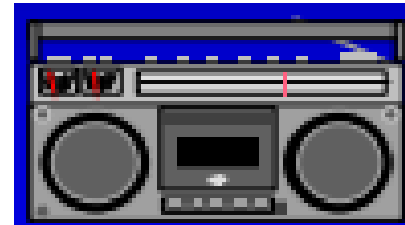
■ Thế nào là trừu tượng hóa?



Người bán hàng



Khách hàng



Sản phẩm

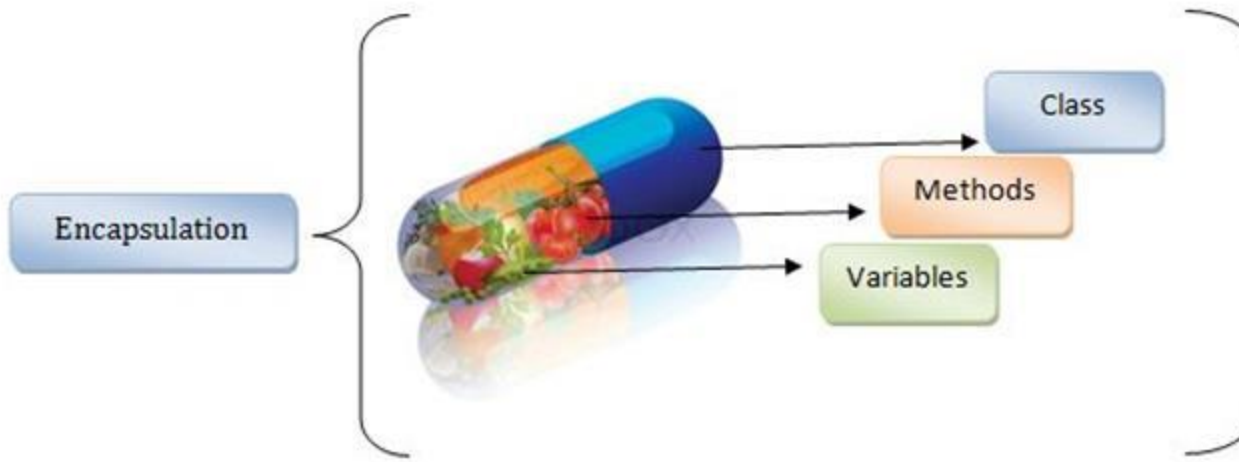
→ Quản lý được độ phức tạp

1. Các tính chất cơ bản của HĐT



■ Tính đóng gói (*Encapsulation*) là gì?

- ◆ Che dấu cài đặt bên trong với clients
 - Client phụ thuộc vào interface
- Tăng tính mềm dẻo

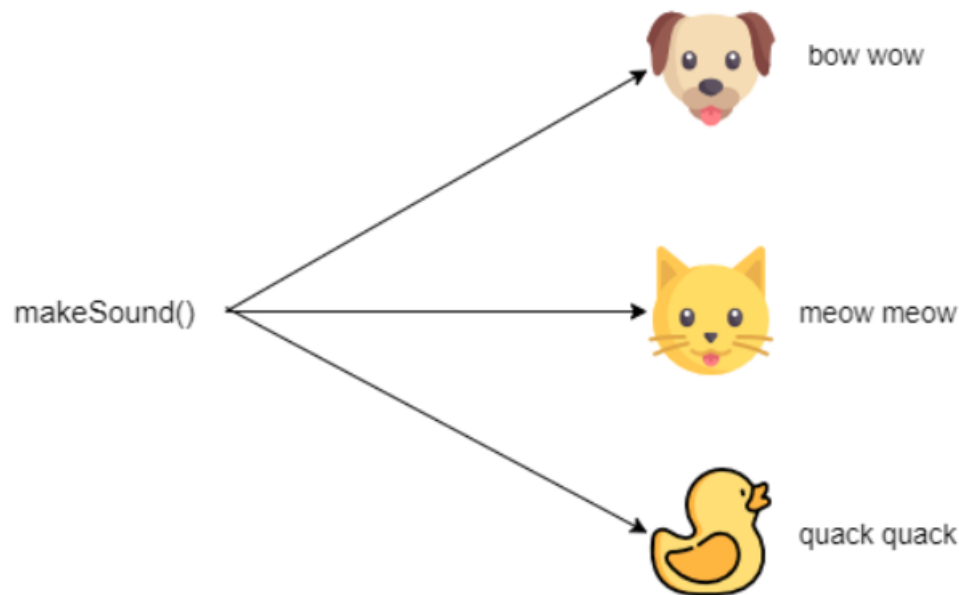


1. Các tính chất cơ bản của HĐT



■ Tính đa hình (Polymorphism) là gì?

- ◆ Đa hình cho phép một phương thức có thể thực hiện khác nhau dựa trên loại đối tượng gọi nó. Điều này giúp linh hoạt trong xử lý các đối tượng khác nhau

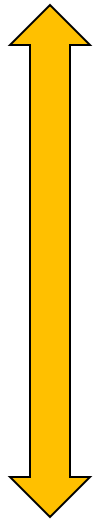


1. Các tính chất cơ bản của HĐT



- Tính kế thừa (Inheritance) là gì?
 - ◆ Mức độ trừu tượng hóa

Tăng mức độ
trừu tượng



BankAccount

Tài khoản
Tiết kiệm

Tài khoản
Thanh toán

Tài sản

Chứng khoán

Bất động sản

Cổ phiếu

Trái phiếu

Các phần tử trên cùng một mức phải có cùng mức độ trừu tượng

2. Các khái niệm cơ bản của HĐT



- Object
- Class
- Attribute
- Operation
- Relationships

2.1 Các KN của HĐT - Object



■ *Đối tượng (Object) là gì?*

- ◆ Một đối tượng là một khái niệm, sự trừu tượng, hoặc một vật với giới hạn rõ ràng và có ý nghĩa với một ứng dụng cụ thể.
- ◆ Một đối tượng có:
 - Trạng thái
 - Hành vi
 - Định danh

2.1 Các KN của HĐT - Object



■ Đối tượng (Object) là gì?

- ◆ Một đối tượng được biểu diễn bởi một hình chữ nhật với tên được gạch dưới.

:Professor

Chỉ có tên lớp

ProfessorClark

Chỉ có tên đối tượng

ProfessorClark:

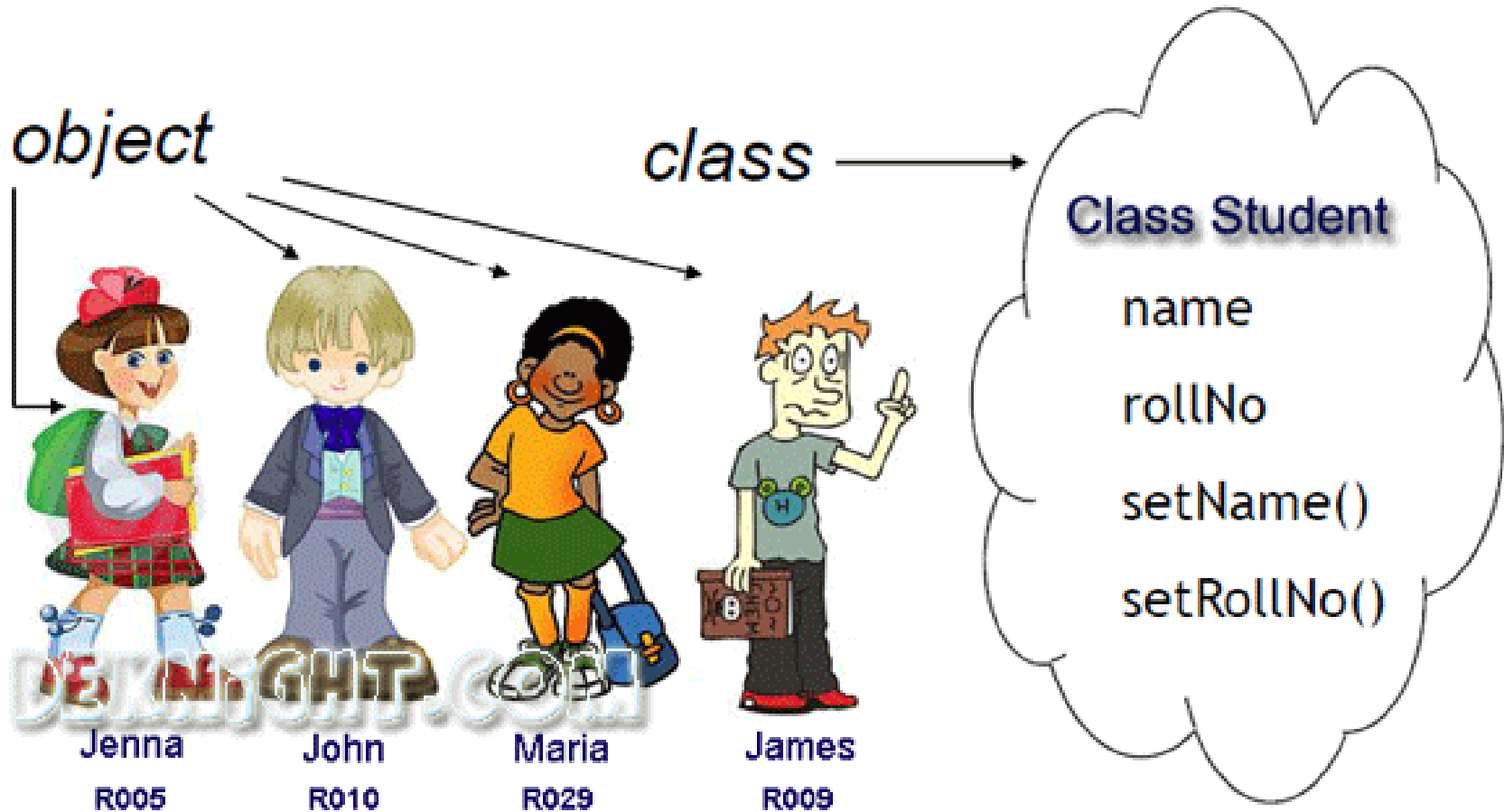
Professor

Tên lớp và tên đối tượng



ProfessorClark

2.2 Các KN của HĐT - Class



2.2 Các KN của HĐT - Class



■ *Class là gì?*

- ◆ Class là mô tả của một nhóm đối tượng có chung các thuộc tính, hành vi, các mối quan hệ và ngữ nghĩa.
 - Một đối tượng là một thể hiện của class
 - ◆ Một class là sự trừu tượng mà trong đó:
 - Nhấn mạnh các tính chất quan trọng
 - Bỏ qua các tính chất khác
- Nguyên tắc trừu tượng hóa

2.2 Các KN của HĐT - Class



■ VD Class

Class
Course

Properties

Tên
Địa điểm
Thời gian
Số tín chỉ
Giờ bắt đầu
Giờ kết thúc



Behavior

Thêm một sinh viên
Hủy một sinh viên
Lấy danh sách giáo sư
Xác định hết chỗ chưa

2.2 Các KN của HĐT - Class



■ *Biểu diễn Class*

- ◆ Một class biểu diễn bằng một hình chữ nhật gồm ba phần

Professor

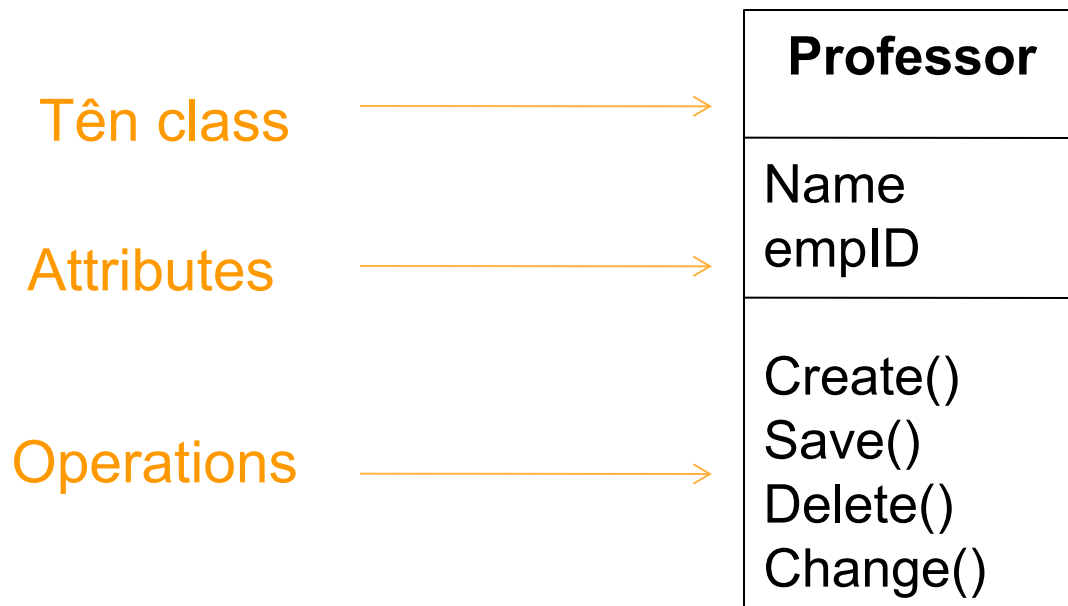


ProfessorClark

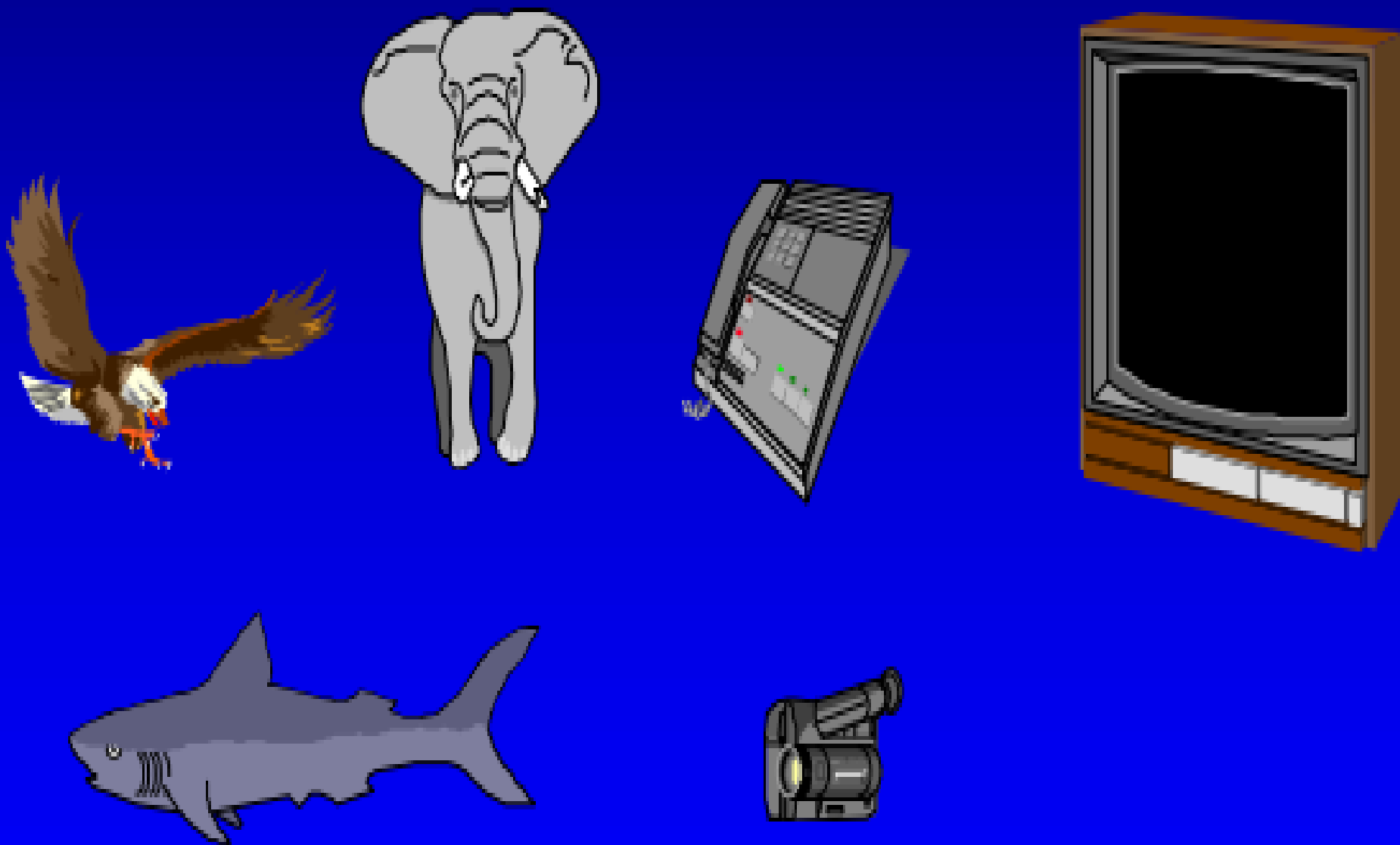
2.2 Các KN của HĐT - Class



- Các phần trong một class
 - ◆ Một class bao gồm ba phần
 - Phần đầu chứa tên class
 - Phần thứ hai cho thấy cấu trúc của lớp (attributes)
 - Phần thứ ba cho thấy các hành vi của lớp (operations)



2. Các khái niệm cơ bản của HĐT



2. Các khái niệm cơ bản của HĐT



■ Quan hệ giữa class và đối tượng

- ◆ Một class là một định nghĩa trừu tượng của một đối tượng
 - Nó định nghĩa cấu trúc và hành vi của mỗi đối tượng trong lớp
 - Nó được dùng như khuôn mẫu để tạo đối tượng
- ◆ Các đối tượng được nhóm thành các class



Objects



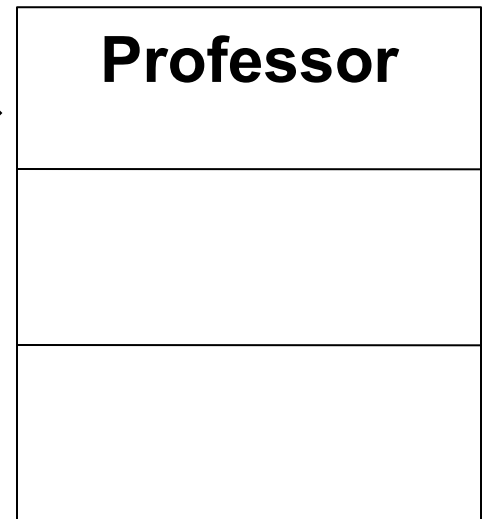
Professor

Professor Smith



Professor Mellon

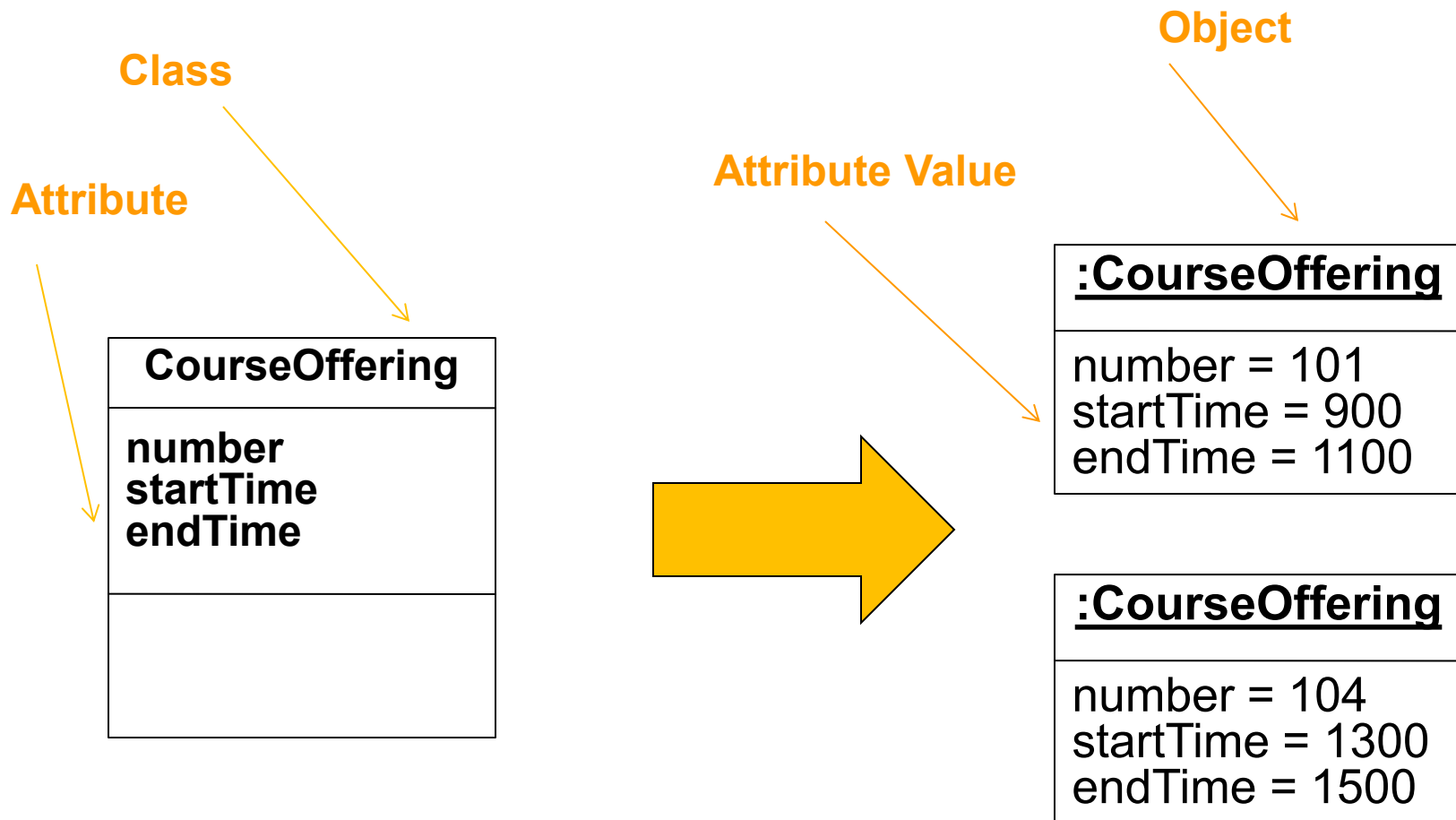
Professor Jones



2.3 Các KN của HĐT - Attribute



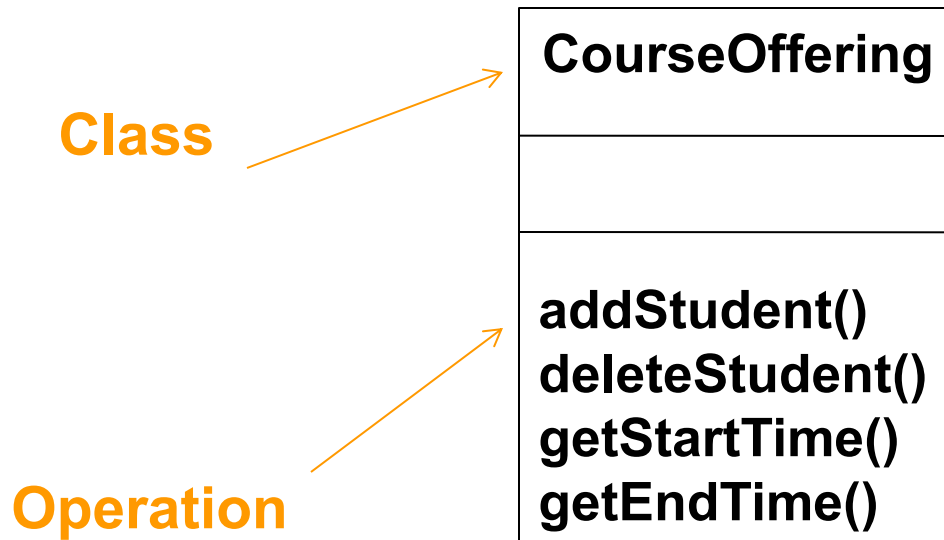
■ Thuộc tính (Attribute) là gì?



2.4 Các KN của HĐT - Operation



■ *Hành vi (Operation) là gì?*



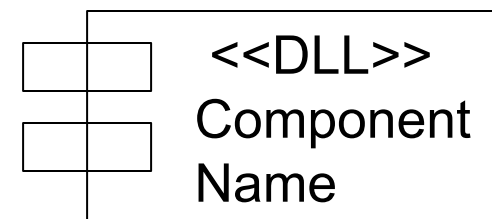
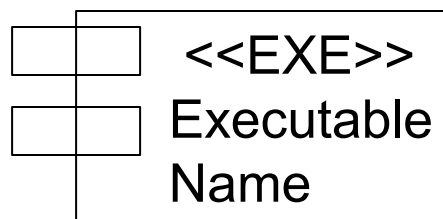
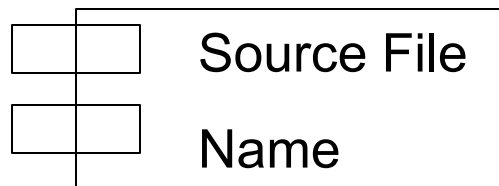
2.7 Các KN của HĐT - Component



■ *Component là gì?*

- ◆ Là một thành phần của hệ thống hoạt động độc lập và giữ một chức năng nhất định trong hệ thống
- ◆ Một component có thể là:
 - Một source code component
 - Một run time component
 - Một executable component

*Nguyên tắc OOP:
đóng gói*

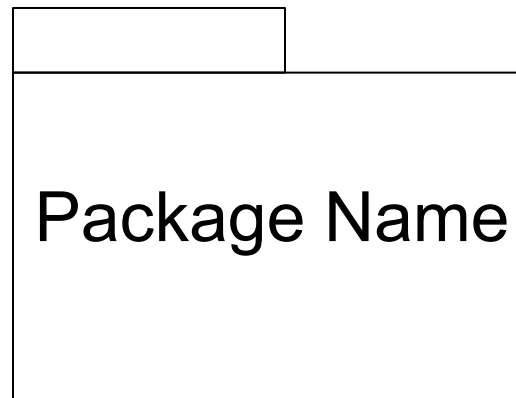


2.8 Các KN của HĐT - Package



■ *Package là gì?*

- ◆ Một package là một cơ chế để tổ chức các phần tử vào thành các nhóm
- ◆ Một phần tử trong mô hình có thể chứa các phần tử khác



*Nguyên tắc OOP:
Tính đơn thể*

- ◆ Dùng để:
 - Tổ chức mô hình đang phát triển
 - Một đơn vị trong quản trị cấu hình

2.9 Các KN của HĐT - Subsystem



■ *Subsystem là gì?*

- ◆ Subsystem là một phần của hệ thống lớn hơn, bao gồm nhiều package hoặc component liên quan, thực hiện một phần chức năng lớn của hệ thống.
- ◆ Nó có thể coi như một hệ thống con độc lập trong hệ thống tổng thể.

2.6 Các KN của HĐT - Relationships



■ *Relationships*

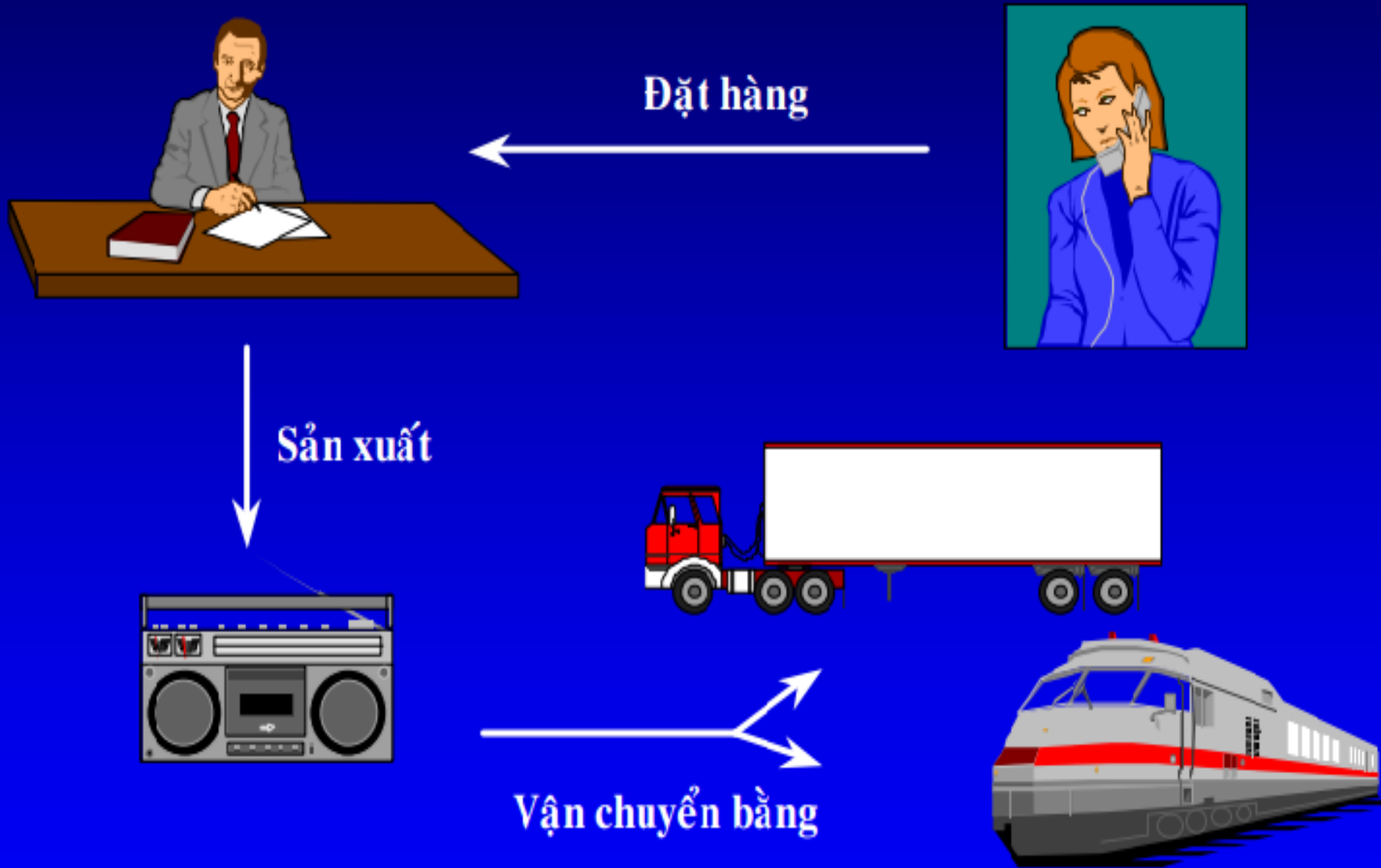
- ◆ Association (kết hợp)
 - Aggregation (thu nạp)
 - Composition (cấu thành)
- ◆ Dependency (phụ thuộc)
- ◆ Generalization (tổng quát hóa)
- ◆ Realization (hiện thực hóa)

3. Sức mạnh của HĐT

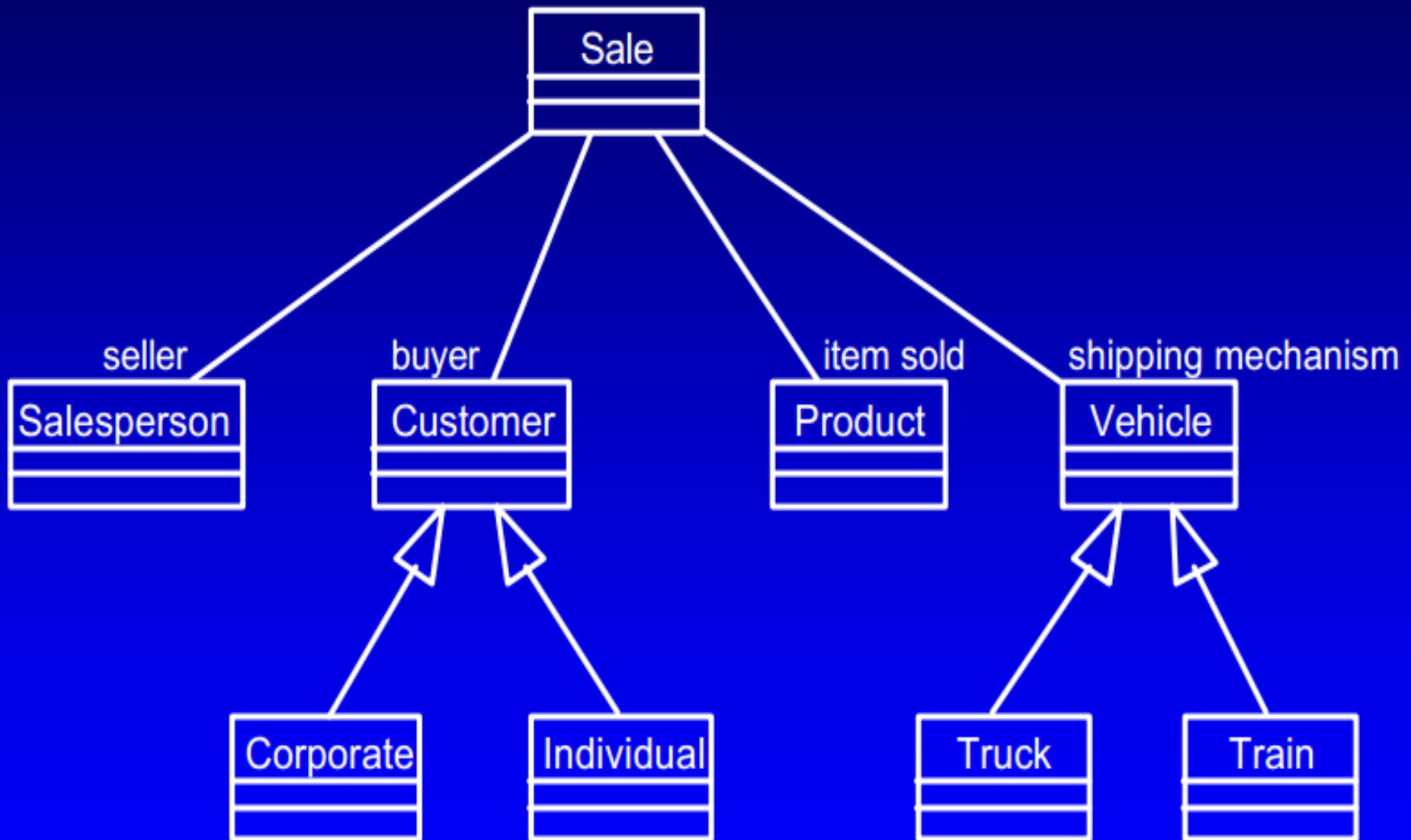


- Một mô hình chung
- Có tính dễ dùng lại
- Mô hình phản ánh chính xác thế giới thực
 - ◆ Mô tả chính xác hơn các tập dữ liệu và các xử lý
 - ◆ Được phân rã dựa trên các phân chia tự nhiên
 - ◆ Dễ hiểu và dễ bảo trì
- Tính ổn định
 - ◆ Một thay đổi nhỏ trong yêu cầu không gây ra sự thay đổi lớn trong hệ thống đang phát triển

3. Ví dụ hệ thống bán hàng

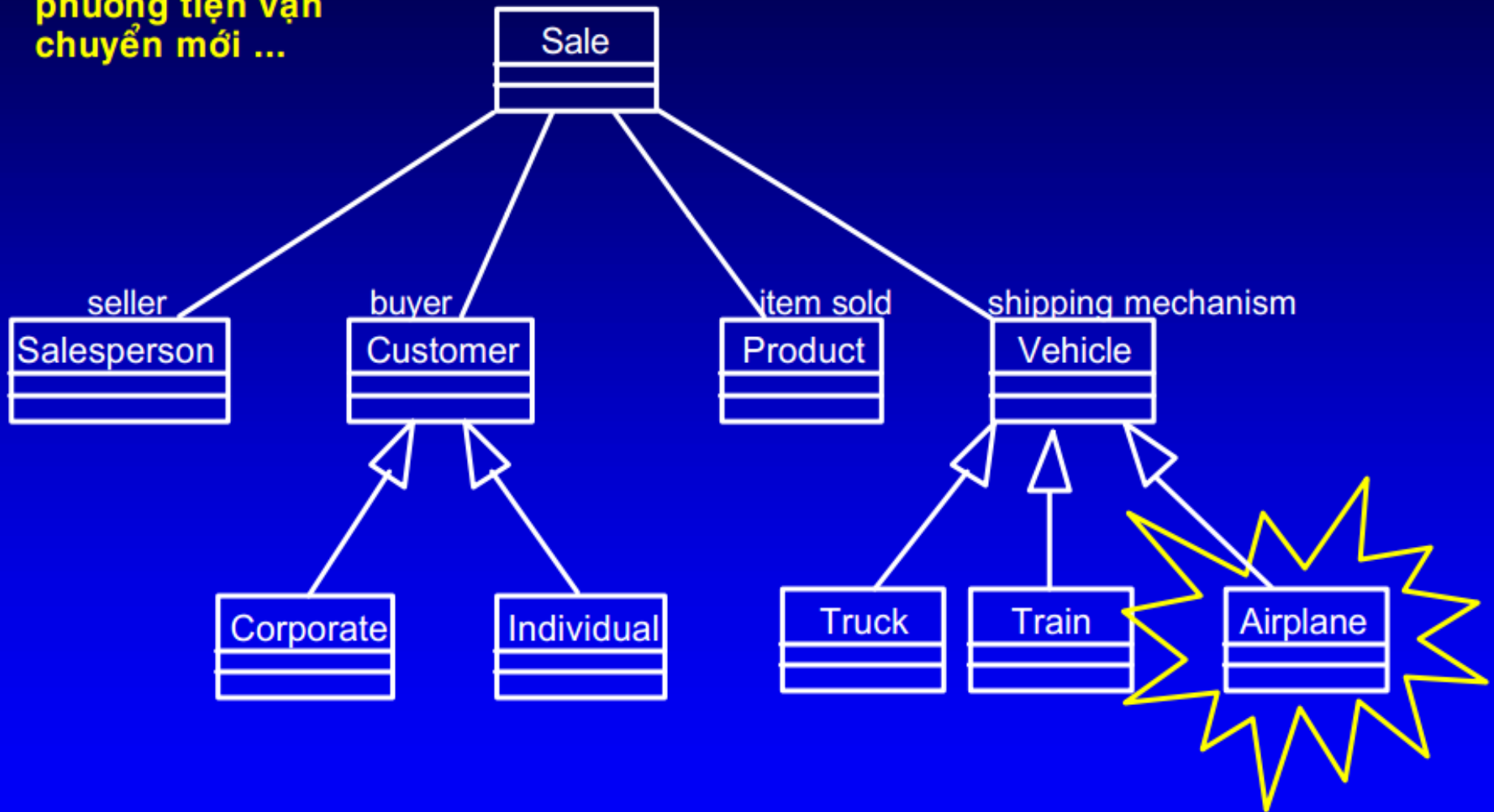


3. VD Sơ đồ lớp hệ thống bán hàng



3. Hiệu ứng của sự thay đổi yêu cầu

Giả sử bạn cần
phương tiện vận
chuyển mới ...



Việc thay đổi liên quan đến việc thêm 1 subclass mới

Câu hỏi ôn tập



1. Bốn tính chất cơ bản của OO là gì? Mô tả về mỗi tính chất
2. Object là gì? Class là gì? Những điểm khác nhau giữa chúng?
3. Attribute là gì?
4. Operation là gì?
5. Polymorphism là gì?

Câu hỏi thảo luận

