

Hướng đối tượng Quá trình suy nghĩ

Chương 1

Giới thiệu các khái niệm hướng đối tượng

Tiêu chí ngôn ngữ 00

§ Đóng gói § Kế thừa

§ Đa hình § Thành phần

Trình bao bọc đối tượng

Ngay cả khi có những lo ngại về di sản, vẫn có xu hướng bọc các hệ thống cũ trong các trình bao bọc đối tượng.

- Giấy gói được dùng để bọc:

- Mã kế thừa
- Mã không di động
- Thư viện của bên thứ 3
- vân vân, .

Đối tượng là gì?

Theo định nghĩa cơ bản, đối tượng là một thực thể chứa cả dữ liệu và hành vi.

- Đây là điểm khác biệt chính giữa phương pháp lập trình truyền thống hơn, lập trình thủ tục và lập trình OO.

Lập trình thủ tục

Trong lập trình thủ tục, mã được đặt vào phương pháp.

- Lý tưởng nhất là các thủ tục này sau đó trở thành "hộp đen", đầu vào đi vào và đầu ra đi ra.
- Dữ liệu được đặt vào các cấu trúc riêng biệt và được xử lý bằng các phương pháp này.

Không thể đoán trước?

Đầu tiên, điều này có nghĩa là việc truy cập dữ liệu không được kiểm soát và không thể đoán trước.

- Thứ hai, vì bạn không có quyền kiểm soát về việc ai có quyền truy cập vào dữ liệu, việc kiểm tra và gỡ lỗi khó khăn hơn nhiều.

Gói đẹp

Các đối tượng giải quyết những vấn đề này bằng cách kết hợp dữ liệu và hành vi thành một gói hoàn chỉnh, đẹp mắt.

- Đối tượng không chỉ là kiểu dữ liệu nguyên thủy, như số nguyên và chuỗi.

Dữ liệu và hành vi

Lập trình thủ tục tách dữ liệu của chương trình khỏi các thao tác thao tác dữ liệu.

- Ví dụ: nếu bạn muốn gửi thông tin qua mạng, chỉ dữ liệu liên quan được gửi với mong muốn chương trình ở đầu bên kia của đường ống biết phải làm gì với nó.

Lập trình 00

Ưu điểm cơ bản của lập trình 00 là dữ liệu và các thao tác thao tác dữ liệu đều được chứa trong đối tượng.

- Ví dụ: khi một đối tượng được vận chuyển qua mạng, toàn bộ đối tượng, bao gồm cả dữ liệu và hành vi, sẽ đi cùng với nó.

Dữ liệu đối tượng

Dữ liệu được lưu trữ trong một đối tượng thể hiện trạng thái của đối tượng.

- Trong thuật ngữ lập trình OO, dữ liệu này được gọi là thuộc tính.

Hành vi đối tượng

Hành vi của một đối tượng là những gì đối tượng có thể làm.

- Trong các ngôn ngữ thủ tục, hành vi được xác định bởi các thủ tục, hàm và chương trình con.

Hành vi đối tượng

Trong thuật ngữ lập trình OO, những hành vi này được chứa trong các phương thức và bạn gọi một phương thức bằng cách gửi tin nhắn tới nó.

Chính xác thì một lớp học là gì?

Một lớp là một bản thiết kế chi tiết cho một đối tượng.

- Khi bạn khởi tạo một đối tượng, bạn sử dụng một lớp làm cơ sở cho cách đối tượng được xây dựng.

Chính xác thì một lớp học là gì?

Một đối tượng không thể
được khởi tạo nếu không có
lớp.

- Các lớp có thể được coi
như các mẫu hoặc khuôn
cắt cookie cho các đối
tượng như trong hình tiếp
theo.



Các loại dữ liệu cấp cao hơn?

Một lớp có thể được coi là một loại kiểu dữ liệu cấp cao hơn.

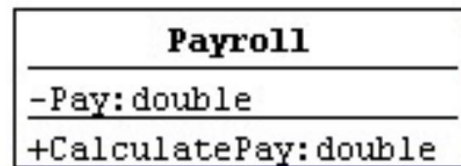
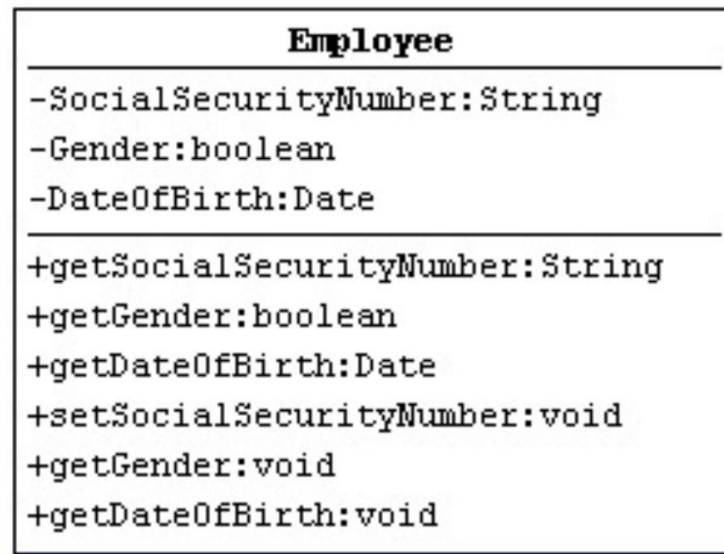
- Ví dụ như khi bạn tạo một số nguyên

hoặc một cái phao:

int x;

thả nổi y;

Lập mô hình một lớp trong UML

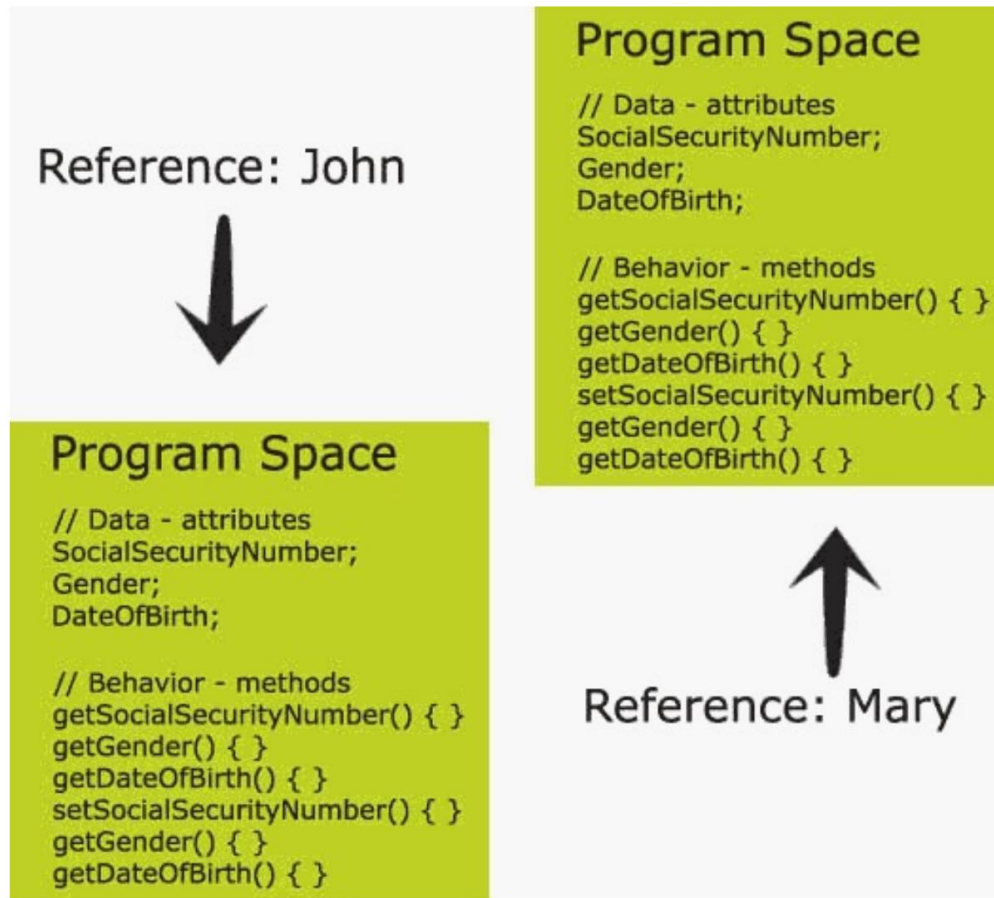


Chữ ký phương thức

Sau đây là tất cả những gì người dùng cần biết để sử dụng hiệu quả các phương pháp:

1. Tên phương pháp
2. Các tham số truyền vào phương thức
3. Kiểu trả về của phương thức

Không gian chương trình



Đóng gói

Một trong những ưu điểm chính của việc sử dụng đối tượng là đối tượng không cần phải bộc lộ tất cả các thuộc tính và hành vi của nó.

§ Trong thiết kế OO tốt (ít nhất là những gì thường được chấp nhận là tốt), một đối tượng chỉ nên hiển thị các giao diện cần thiết để tương tác với nó.

Đóng gói

Các chi tiết không liên quan đến việc sử dụng đối tượng sẽ được ẩn khỏi các đối tượng khác.

- Điều này được gọi là đóng gói.

Giao diện

Bất kỳ hành vi nào mà đối tượng cung cấp phải được được gọi bằng một tin nhắn được gửi bằng một trong các giao diện được cung cấp.

- Giao diện phải mô tả đầy đủ cách người dùng của lớp tương tác với lớp.
- các phương thức là một phần của giao diện được chỉ định là công khai.

Giao diện

Các giao diện thường không bao gồm các phương thức chỉ thuộc tính.

§ Như đã thảo luận trước đó trong mô-đun này, nếu người dùng cần quyền truy cập vào một thuộc tính thì một phương thức sẽ được tạo để trả về thuộc tính đó.

Triển khai

Chỉ các thuộc tính và phương thức công khai mới được coi là giao diện.

- Người dùng sẽ không nhìn thấy bất kỳ phần nào của triển khai tương tác với một đối tượng chỉ thông qua các giao diện.
- Ở ví dụ trước là Nhân viên

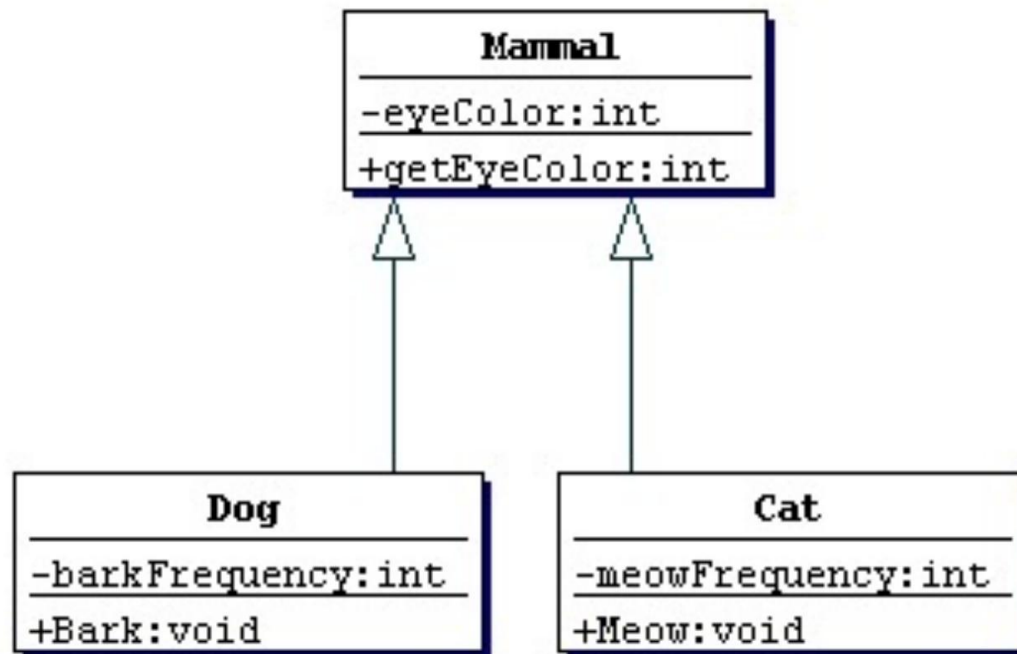
lớp, chỉ có các thuộc tính bị ẩn.
- Do đó, việc triển khai có thể thay đổi và sẽ không ảnh hưởng đến mã của người dùng.

Di sản

Kế thừa cho phép một lớp kế thừa các thuộc tính và các phương thức của lớp khác.

- Điều này cho phép bạn tạo các lớp hoàn toàn mới bằng cách trừu tượng hóa các thuộc tính và hành vi phổ biến.

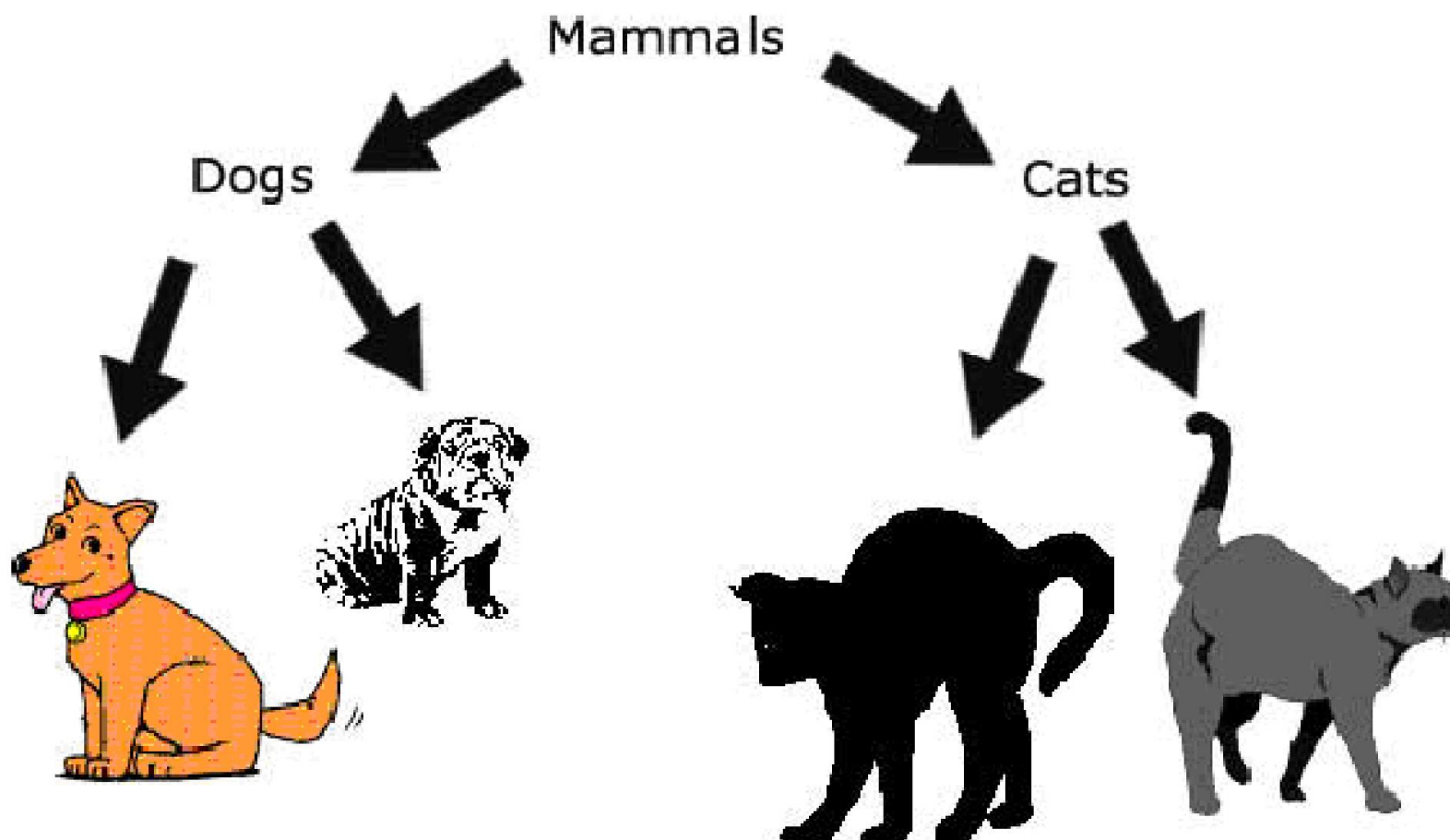
Hệ thống phân cấp động vật có vú



Siêu lớp và lớp con

§ Lớp cha, hay lớp cha, chứa tất cả các thuộc tính và hành vi chung cho các lớp kế thừa từ nó.

Hệ thống phân cấp động vật có vú



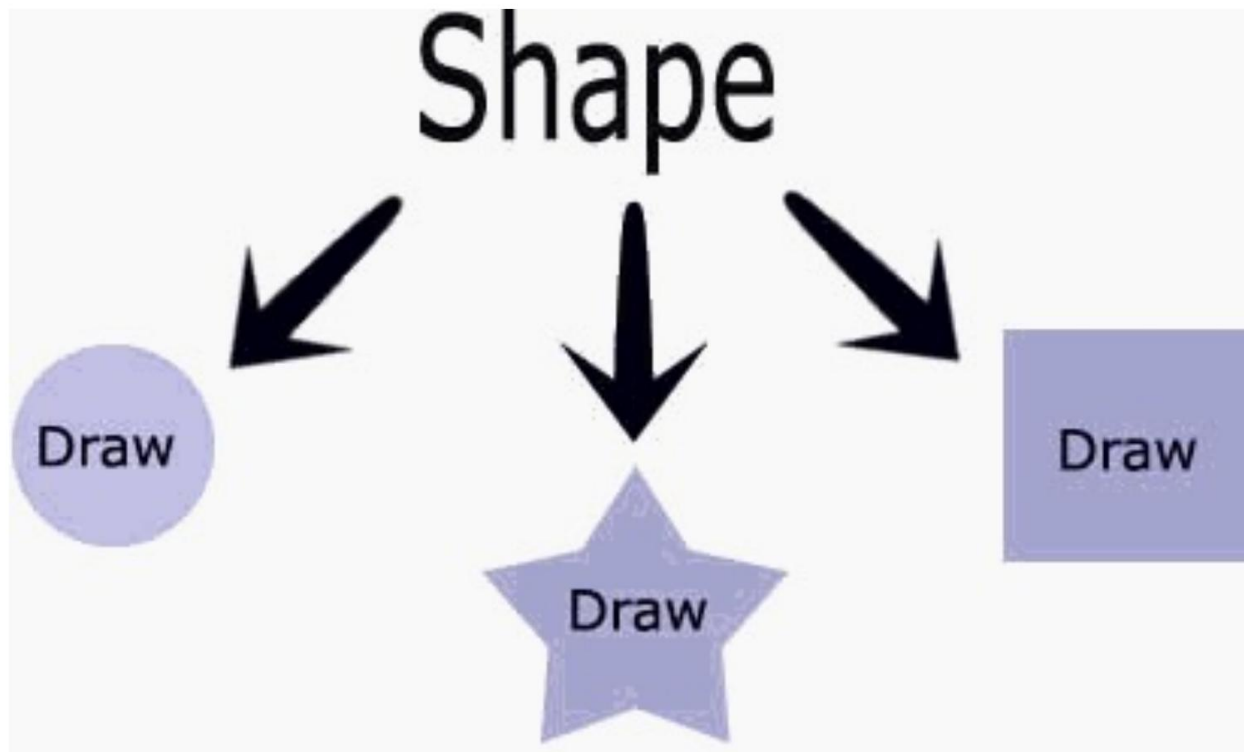
Mối quan hệ là một

Trong ví dụ về Hình dạng, Hình tròn, Hình vuông và Ngôi sao đều kế thừa trực tiếp từ Hình dạng.

- Mối quan hệ này thường được gọi là mối quan hệ is-a vì hình tròn là một hình dạng.

Đa hình

Đa hình theo nghĩa đen có nghĩa là nhiều hình dạng.



Đa hình

Đối tượng Shape không vẽ được hình, nó quá trừu tượng (thực tế là phương thức Draw() trong Shape _____ không chứa phần thực hiện).

- Bạn phải xác định hình dạng cụ thể.
- Để làm được điều này, bạn cung cấp cách triển khai thực tế trong Circle.

Đa hình

Nói tóm lại, mỗi lớp có thể phản hồi khác nhau với cùng một phương thức Draw và tự vẽ.

- Đây chính là ý nghĩa của tính đa hình.

Đa hình

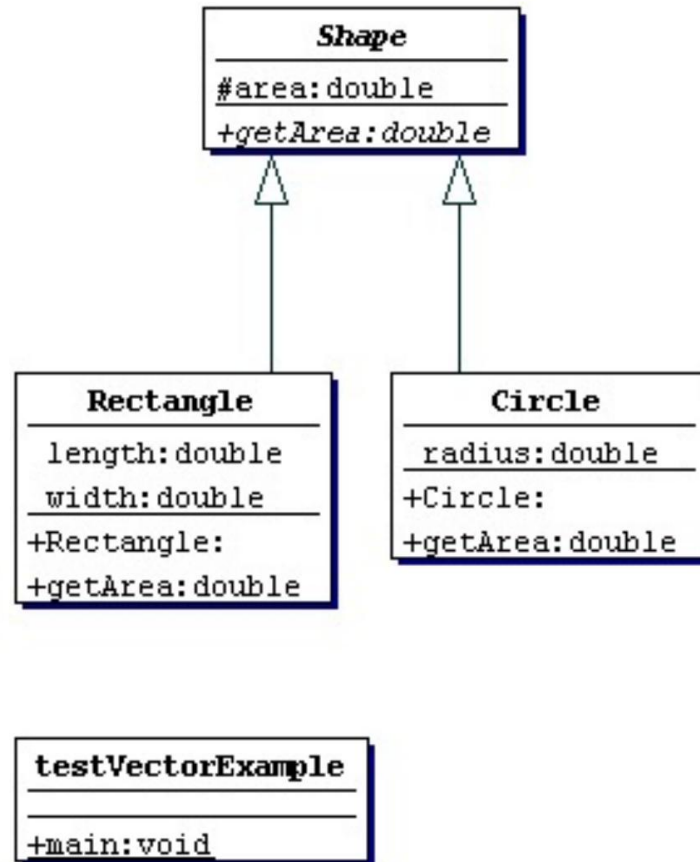
Khi một phương thức được định nghĩa là trừu tượng, một lớp con phải cung cấp cách triển khai cho phương thức này.

- Trong trường hợp này, Shape yêu cầu các lớp con cung cấp cách triển khai getArea().

Đa hình

- § Nếu một lớp con kế thừa một phương thức trừu tượng từ một siêu lớp, thì nó phải cung cấp cách triển khai cụ thể của phương thức đó, nếu không nó sẽ là một lớp trừu tượng (xem hình sau để biết sơ đồ UML).

Hình dạng sơ đồ UML



Thành phần

Điều tự nhiên là nghĩ các vật thể chứa đựng các vật thể khác.

- Một chiếc tivi có bộ thu sóng và màn hình video.

Thành phần

Một máy tính có card màn hình, bàn phím và ổ đĩa.

- Mặc dù máy tính có thể được coi là một đối tượng nhưng ổ đĩa cũng được coi là một đối tượng hợp lệ.

Thành phần

Trên thực tế, bạn có thể mở máy tính, tháo ổ đĩa và cầm nó trên tay.

- Cả máy tính và ổ đĩa đều được coi là đối tượng.
Chỉ là máy tính chứa các đối tượng khác như ổ đĩa.
- Bằng cách này, các đối tượng thường được xây dựng hoặc kết hợp từ các đối tượng khác:
Đây là sự kết hợp.

Có một mối quan hệ

Trong khi mối quan hệ thừa kế được coi là mối quan hệ Is-a vì những lý do đã được thảo luận, thì mối quan hệ thành phần được gọi là mối quan hệ Has-a.

Có một mối quan hệ

§ Sử dụng ví dụ ở phần trước, tivi Has-a một bộ dò sóng và màn hình hiển thị video Has-a.

§ Tivi rõ ràng không phải là bộ thu sóng nên không có quan hệ thừa kế.

Có một mối quan hệ

Tương tự như vậy, máy tính Có card màn hình, Có bàn phím và Có ổ đĩa.

- Các chủ đề về kế thừa, thành phần và cách chúng liên quan với nhau sẽ được đề cập rất chi tiết ở phần sau của khóa học.