



Chương 05

Lập trình hướng đối tượng



Phần 01: Class & Object

Lập trình hướng thủ tục & hướng đối tượng

Lập trình hướng thủ tục

- Giải quyết vấn đề từng bước đến khi đạt yêu cầu
- Lập trình từ trên xuống,
- Lập trình theo hàm → chỉ tạo ra hàm xử lý khi gặp vấn đề nào đó

Lập trình hướng đối tượng

- Dựa trên nền tảng các lớp đã xây dựng sẵn
- Xác định trước các chức năng cần phải thực hiện

Lập trình hướng đối tượng

- OOP – Object oriented programming là kiểu lập trình lấy đối tượng làm nền tảng
- Đơn giản hóa việc phát triển chương trình
- Tạo ra các chương trình có tính mềm dẻo và linh động cao
- Dễ dàng phát triển, bảo trì và nâng cấp.

Phân biệt Class & Object

- Class chỉ một cái gì đó chung chung, object là một cái cụ thể
- Ví dụ:
 - Công thức làm bánh quy là một Class → bánh quy là một Object
 - Con gái là một Class → Hồng là một Object
 - Con mèo là một Class → Con mèo Mimi nhà tôi là một Object
 - Bản vẽ là một Class → Ngôi nhà của tôi là một Object
- Hướng dẫn khai báo và sử dụng class ?



Phần 02: Property & Method

Phương thức và thuộc tính của đối tượng

Thuộc tính (Property)

- Là các đặc điểm, đặc tính của một lớp

Phương thức (Method)

- Là các hành động có thể được thực hiện từ lớp
- Phương thức cũng giống như hàm, nhưng là hàm riêng của lớp

Ví dụ về phương thức và thuộc tính của đối tượng

Xét Object “Bạn gái Hồng”

- Đặc điểm: tóc dài, má lúng đồng tiền, cao 1.7 m, ... → thuộc tính
- Hành động: ăn, ngủ, dạy anh văn, đánh đàn, ... → phương thức

Xét Object “Con mèo Mimi nhà tôi”

- Đặc điểm: lông xoăn, màu xám, đuôi ngắn, ... Hành động: ăn, ngủ, bắt chuột,

Xét Object “Con mèo Kitty nhà bạn”

- Đặc điểm: màu trắng, đuôi dài, ... Hành động: ăn, ngủ, làm nũng, ...

Xây dựng Class với các phương thức và thuộc tính

- Chúng ta đã học qua các kiểu dữ liệu như INT, STRING, FLOAT để mô tả số nguyên, chuỗi và số thực → có kiểu dữ liệu mới → mô tả kiểu dữ liệu mới bằng cách nào → định nghĩa kiểu dữ liệu mới này thông qua Class
- Trước khi xây dựng một Class, chúng ta cần xác định 2 vấn đề sau:
 - Thuộc tính
 - Phương thức
- **Xây dựng Class ConMeo với các phương thức và thuộc tính của nó !**



Phần 03: Constructor & Destructor

Phương thức `__construct()`

- `__construct()` được gọi tự động và được gọi đầu tiên khi một object được khởi tạo.
- Thường dùng để khởi tạo các giá trị ban đầu, các trường hợp gọi `__construct()`
 - `__construct()`
 - `__construct()` với tham số
 - `__construct()` với tham số mặc định
 - `__construct()` với cách đặt tên trùng với tên class
 - `__construct()` với tham số là mảng

Phương thức `__destruct()`

- `__destruct()` là phương thức tự động chạy khi đối tượng được khởi tạo. Nó chỉ được thực thi ở cuối trang mà đối tượng được tạo ra
- Phương thức này thường dùng để tạo hoặc hủy một session, giải phóng bộ nhớ, đóng kết nối của ứng dụng đến database, đóng kết nối đến tập tin, ...



Phần 04: Tính kế thừa & overwrite

Tìm hiểu từ khóa FINAL

- PHP5 cho phép định nghĩa class và method với từ khóa FINAL
- Đối với method: Lớp con không thể override các phương thức ở lớp cha nếu các phương thức ở lớp cha có khóa FINAL
- Đối với class: khi chúng ta dùng từ khóa FINAL thì chúng ta không thể extends từ class đó



Phần 05: Phạm vi và sự ảnh hưởng

Public – Private – Protected

- Public: có thể truy cập từ mọi nơi
- Protected: chỉ sử dụng cho class đó và các class được kế thừa từ class đó
- Private: chỉ sử dụng ở chính class đó

Static

- Được dùng với phương thức và biến
- Truy cập nhanh phương thức mà không cần khởi tạo đối tượng

Self & Parent

self

- Là đại diện cho cách khởi tạo lớp hiện thời và thường được sử dụng gọi đến biến số có khóa static hay hàm nào đó trong lớp hiện tại

parent

- Là đại diện của lớp cha và thường được sử dụng gọi đến biến số có khóa static hay phương thức nào đó trong lớp cha của lớp hiện tại

const (Constant)

- Định nghĩa các biến có giá trị không thay đổi bằng từ khóa const
- Để truy cập vào lấy giá trị chúng ta sử dụng toán tử ::



Phần 06: Một số phương thức khác

clone()

- Sao chép một đối tượng từ đối tượng khác

`__autoload()`

- Tự động nạp các lớp từ một đường dẫn chứa các lớp được truyền vào

__sleep & __wakeup()

- Phương thức __sleep() được thực hiện khi chúng ta đưa một đối tượng vào hàm serialize. Lúc này ở phương thức __sleep() chúng ta sẽ khai báo những thành phần nào của đối tượng được chuyển thành một chuỗi đặc biệt
- Phương thức __wakeup() dùng để làm sạch đối tượng như khi mới khởi tạo ban đầu với các giá trị mặc định được gán vào. Điều này giúp chúng ta không phải khởi tạo đối tượng một lần nữa

__toString()

- Phương thức __toString() giúp chuyển đổi tượng thành chuỗi

__set & __get()

- Phương thức __set() thiết lập các thuộc tính cho đối tượng cho dù thuộc tính đó chưa được khai báo trong class
- Phương thức __get() lấy giá trị của một thuộc tính nào đó trong class cho dù thuộc tính đó được gán khóa private hoặc protected