

# 4 khái niệm bạn phải biết khi lập trình hướng đối tượng

**Lập trình hướng đối tượng** (Object-oriented programming) là một phương pháp quen thuộc đối với các chuyên gia phát triển phần mềm. Sau đây là 4 khái niệm bạn bắt buộc phải biết để hiểu về lập trình hướng đối tượng.

## Mục lục [\[hide\]](#)

- [1. Class \(Lớp\)](#)
- [2. Objects \(Đối tượng\)](#)
- [3. Attributes \(Thuộc tính\)](#)
- [4. Methods \(Phương thức\)](#)

Class là bản thiết kế về thuộc tính và hành vi chung để khái quát hóa các đối tượng cùng đặc điểm.

## 1. Class (Lớp)

Về cơ bản, lớp là các kiểu dữ liệu do người dùng tự định nghĩa. **Lớp là nơi chúng ta tạo ra một bản thiết kế (blueprint) mẫu cho cấu trúc của các đối tượng.** Nội dung của lớp bao gồm phương thức (method) và thuộc tính (attribute). Từ bản thiết kế này, chúng ta có thể khởi tạo những đối tượng riêng lẻ khác nhau. **Nói cách khác, lớp là thứ ta dựa vào để tạo ra các đối tượng khác nhau.**

Dưới đây là một ví dụ về lớp sử dụng ngôn ngữ JavaScript:

Ví dụ về lớp trong lập trình hướng đối tượng.

Trong ví dụ trên, chúng ta có lớp “Dog” (Chó) với các thuộc tính là “Name” (Tên) và “Birthday” (Ngày sinh).

```
1 class Dog {
2   constructor(name, birthday) {
3     this.name = name;
4     this.birthday = birthday;
5   }
6 }
7
8 //instantiate a new object of the Dog class, and individual dog named Rufus
9 const rufus = new Dog("Rufus", "2/1/2017");
```

Lớp chính là khuôn mẫu để tạo mô hình cho một đối tượng. Dựa vào lớp “Dog”, lập trình viên sau đó có thể tạo ra những đối tượng cụ thể, có tên và ngày sinh khác biệt, chẳng hạn như “Dog 1”, “Dog 2”.

## 2. Objects (Đối tượng)

Tất nhiên, trong [lập trình](#) hướng đối tượng thì bắt buộc phải có đối tượng, như tên gọi của nó. Một đối tượng sẽ bao gồm trạng thái và hành vi cụ thể.

Trạng thái của đối tượng được xác định bởi thuộc tính, tức là những thông tin mà bạn muốn lưu trữ trong đối tượng. Hành vi của đối tượng chính là phương thức, tức là những điều mà bạn muốn đối tượng thực hiện.

Quy trở lại với ví dụ trên, khi một lớp “Dog” được đã được khởi tạo. Nếu chúng ta tiếp tục chạy đoạn mã sau:

- `const rufus = new Dog(“Rufus”, “2/1/2017”);`

Khi đó, sẽ có một đối tượng mới được tạo ra tên là “rufus”. Đồng thời, hàm khởi tạo đối tượng cũng sẽ gán giá trị tên và ngày sinh cho đối tượng. Cụ thể, tên của đối tượng này là “rufus” và có ngày sinh là “2/1/2017”.

### 3. Attributes (Thuộc tính)

Có thể hiểu, thuộc tính chính là những thông tin cụ thể được lưu trữ. Thuộc tính sẽ được định nghĩa trong lớp. Khi một đối tượng được khởi tạo, thì sẽ có những dữ liệu khác nhau để lưu trữ thuộc tính của chúng. Lúc này, thuộc tính sẽ giúp chúng ta phân biệt giữa đối tượng này với đối tượng khác.

Trong ví dụ trên, đối tượng “Dog” có hai thuộc tính là “Name” (Tên) và “Birthday” (Ngày sinh). Tất nhiên, sẽ có trường hợp nhiều đối tượng có tên và ngày sinh giống nhau. Lúc đó, ta lại cần thêm một thuộc tính khác để phân biệt chúng.

### 4. Methods (Phương thức)

Trong lập trình hướng đối tượng, phương thức đại diện cho các hành vi. Cụ thể, phương thức có thể yêu cầu đối tượng thực hiện hành động, trả lại thông tin về một đối tượng hoặc cập nhật dữ liệu của một đối tượng.

Phương thức sẽ bao gồm những đoạn mã được xác định trong lớp. Khi một đối tượng được tạo ra, thì đối tượng này có thể gọi đến các phương thức đã được định nghĩa trước từ trong lớp.

Ví dụ, xét trong đoạn mã dưới đây, phương thức “bark()” (sủa) đã được định nghĩa trong lớp “Dog”. Khi đó, đối tượng “rufus” hoàn toàn có thể gọi đến phương thức “bark()”.

Ví dụ về phương thức trong lập trình hướng đối tượng.

**Ý nghĩa của phương thức trong lập trình hướng đối tượng**

Phương thức thường sửa đổi, cập nhật hoặc xóa dữ liệu. Tuy nhiên, có những phương thức được thực hiện mà không thay đổi bất kỳ dữ liệu nào.

Ví dụ: phương thức “bark()” không cập nhật bất kỳ dữ liệu nào bởi vì hành động của phương thức này không sửa đổi bất kỳ thuộc tính nào của lớp “Dog” (tên hoặc ngày sinh).

Thông qua việc dùng phương thức, lập trình viên sẽ tận dụng được khả năng tái sử dụng của mã, đồng thời giữ cho các chức năng gói gọn bên trong một đối tượng. Đây là một điểm hữu ích cho việc gỡ lỗi, vì khi gỡ lỗi, ta chỉ cần tìm đến một nơi duy nhất thay vì nhiều nơi.

Đối tượng, phương thức, lớp và thuộc tính là 4 nội dung cơ bản của một chương trình được lập trình theo hướng đối tượng mà lập trình viên cần nắm rõ. Hy vọng qua bài viết này, bạn đã hiểu rõ hơn về 4 khái niệm bắt buộc phải biết trong **lập trình hướng đối tượng**.