

# Bài 27: Lớp và đối tượng

---

- ✓ Khái niệm
- ✓ Mục đích sử dụng
- ✓ Cú pháp tổng quát
- ✓ Ví dụ minh họa
- ✓ Bài tập thực hành

# Khái niệm

- Trong thế giới thực có nhiều đối tượng có các đặc điểm giống hệt nhau, ví dụ như iphone 12 Pro Max.
- Lý do là vì chúng được tạo ra từ cùng một bản mẫu thiết kế, cùng khuôn đúc do đó có các tính năng và ngoại hình giống nhau
- Để máy tính hiểu được thế giới thực của con người thì ta dùng các lớp(class) để làm công cụ mô tả.
- Lớp là một bản mẫu thiết kế chi tiết mà từ đó các đối tượng riêng lẻ của nó được tạo ra
- Hoặc có thể hiểu lớp là một bản mô tả cụ thể của đối tượng nào đó trong ngôn ngữ lập trình

# Khái niệm

- Một đối tượng thường có các đặc điểm nhận biết và các hành động nào đó
- Đối tượng có thể là đối tượng cụ thể hoặc trừu tượng: xe máy, ô tô, smartphone, đơn hàng, tài khoản, giỏ hàng...
- Bất kì cái gì có thể dùng danh từ để đặt tên cho nó thì đó là đối tượng

# Mục đích sử dụng

- Mô hình hóa một đối tượng từ thế giới thực vào thế giới của ngôn ngữ lập trình để máy tính hiểu được
- Một lớp thường có hai thành phần chính: các thuộc tính và các phương thức. Ngoài ra còn có thể có thêm các thành phần khác: class, interface, enum...
- Các thuộc tính bản chất là các biến của đối tượng dùng để lưu trữ đặc điểm đặc trưng và nhận biết đối tượng
- Các phương thức dùng để mô tả các hành động của đối tượng.

# Cú pháp tổng quát

```
access non-access class name {  
    // properties  
    // constructors  
    // methods  
    // others  
}
```

## ➤ Trong đó:

- Phần access là access modifier, có thể là public, private, protected hoặc để trống. Mỗi lớp chỉ có 1 access modifier và chỉ xuất hiện 1 lần. Mỗi file chương trình Java chỉ có 1 và chỉ 1 public class
- Phần non-access là phần bổ trợ làm rõ nghĩa cho một lớp. Phần này có thể có hoặc không. Các non-access có thể là: final, abstract, static, strictfp

# Cú pháp tổng quát

```
access non-access class name {  
    // properties  
    // constructors  
    // methods  
    // others  
}
```

## ➤ Trong đó:

- Hiện tại ta mặc định phần non-access là bỏ trống và sẽ tìm hiểu sau
- Keyword class là bắt buộc phải có đặt trước tên lớp trong định nghĩa lớp
- Phần name là tên một lớp. Tên lớp thường là danh từ số ít.

# Cú pháp tổng quát

```
access non-access class name {  
    // properties  
    // constructors  
    // methods  
    // others  
}
```

## ➤ Trong đó:

- Tên lớp luôn bắt đầu với chữ cái hoa và nếu tên có nhiều từ thì viết liền các từ và viết hoa chữ cái đầu tất cả các từ
- Tên lớp phải đặt sao cho nó mô tả đúng đối tượng lớp muốn thể hiện

# Cú pháp tổng quát

```
access non-access class name {  
    // properties  
    // constructors  
    // methods  
    // others  
}
```

## ➤ Trong đó:

- Phần thân của lớp nằm trong cặp {} chứa các thành phần mô tả thông tin và hành động cũng như những yếu tố cấu thành khác
- Phần thân của lớp có thể có một hoặc nhiều thành phần như trên
- Phần properties là các thuộc tính mô tả đặc điểm của đối tượng tạo ra từ lớp. Đây là phần dữ liệu của lớp. Để đơn giản bây giờ ta coi các properties là public



# Cú pháp tổng quát

```
access non-access class name {  
    // properties  
    // constructors  
    // methods  
    // others  
}
```

## ➤ Trong đó:

- Tiếp theo là phần constructors là các phương thức khởi tạo. Dùng để tạo đối tượng với các thông tin ban đầu từ lớp
- Phần methods là các phương thức khác constructors. Dùng để mô tả các hành động, nhiệm vụ, hành vi của đối tượng
- Phần others có thể là các lớp, các interface hoặc enum... Phần này sẽ được đề cập sau

# Ví dụ

```
/**
 * Lớp mô tả thông tin về con mèo
 */
public class Cat {
    // thuộc tính
    public String name;
    public int age;
    public float weight;
    public String color;
    public String favoriteFood;

    // constructor:
    public Cat(String name) {
        this.name = name;
    }

    // các phương thức:

    /**
     * phương thức mô tả con mèo nói chuyện với nhau như nào
     */
    public void speak() {
        // mèo nói chuyện...
    }
}
```

# Cú pháp tạo đối tượng

➤ Cú pháp: `type variable = new type(params);`

➤ Trong đó:

- Phần type là tên lớp. Bây giờ tên lớp sẽ được coi như một kiểu người dùng tự định nghĩa và gọi tắt là kiểu
- Variable là tên biến đặt theo quy tắc đặt tên biến
- Vế phải là toàn bộ cú pháp dùng để tạo ra một đối tượng mới, bắt buộc phải có keyword new
- Params là danh sách các tham số truyền vào constructor được gọi

➤ Ví dụ:

```
Cat tom = new Cat("Tom");  
Student linh = new Student("Nguyễn Thùy Linh");  
Student nhan = new Student("B25DCCN111", "Hoàng Trọng Nhân");
```

# Truy cập thành phần của đối tượng

➤ Cú pháp: `object.name`

➤ Trong đó:

- Phần `object` là tên biến đối tượng
- Phần `name` là thành phần muốn truy cập. Thường ta chỉ truy cập được các thành phần `public`.

➤ Ví dụ:

```
Cat tom = new Cat("Tom");
Student linh = new Student("Nguyễn Thùy Linh");
Student nhan = new Student("B25DCCN111", "Hoàng Trọng Nhân");
// hiển thị tên của mèo tom:
System.out.println("Hello " + tom.name);
// lấy mã sinh viên:
System.out.println("Mã sinh viên: " + nhan.studentId);
// cho sinh viên Linh đi thi môn Java:
linh.doExam("Lập trình Java");
```



# Ví dụ minh họa

- Ví dụ 1: Tạo lớp mô tả thông tin sinh viên.
- Ví dụ 2: Tạo lớp mô tả thông tin về mèo kitty.

# Nội dung tiếp theo

**Phương thức khởi tạo**