

# PHP面向对象(第1天)

- 1.概念
- 2.继承
- 3.封装
- 4.实例:分页类
- 5.实例:数据库操作类
- 6.抽象类
- 7.接口
- 8.多态
- 9.魔术方法
- 10.关键字

## 1.面向对象的概念（OOP）

OOP(Object-Oriented Programming, 面向对象的编程)技术为编程人员敞开了一扇大门，使其编程的代码更简洁、更易于维护，并且具有更强的可重用性。

## 2.OOP达到了软件工程的三个目标：

重用性、灵活性、扩展性

## 3.OOP面向对象编程的特点：

封装、继承、多态

1.类与对象的关系就如模具和铸件的关系，类的实例化结果就是对象，而对对象的抽象就是类.类描述了一组有相同特性（属性）和相同行为（方法）的对象。

2.什么是类：

在面向对象的编程语言中，类是一个独立的程序单位，是具有相同属性和服务的一组对象集合。它为属于该类的所有对象提供了统一抽象描述，其内部包括成员属性和服务的方法两个主要部分。

3.什么是对象：

在客观世界里，所有的事务都是由对象和对象之间的联系组成的。对象是系统中用来描述客观事物的一个实体，他是构成系统的一个基本单位。

- 1.成员属性
- 2.成员方法

1. 实例化对象
2. 对象中成员的访问
3. 特殊的对象引用\$`this`
4. 构造方法和析构方法

- \$obj->a()->b();

- public



- 1) 属性的作用域
- 2) 局部变量的作用域

1.父类

2.子类

- 子类可以访问父类的成员，继承关键字 **extends**，可以代码的重用性，子类中写了一个与父类同名的方法，这个方法会覆盖父类中的方法。

- ```
function say(){  
    parent::say();  
    echo "我的名字: $this->name,我的年龄:  
    $this->age";  
    echo "我的学校: $this->sch";  
}
```

1.public

2.protected

3.private

注:子类属性或方法的权限一定要比父类的权限宽松

- 主要为了给类成员设置访问权限用的，关键字有三个，  
`public`, `protected`, `private`, `public`外界和子类可以使用, `protected`子类可以使用, `private`本类可以使用

|           | 本类  | 子类  | 外面  |
|-----------|-----|-----|-----|
| public    | yes | yes | yes |
| protected | yes | yes | no  |
| private   | yes | no  | no  |

- 实例1:分页类
- 实例2:数据库操作类



1. 官方网址: [www.yzmedu.com](http://www.yzmedu.com)

2. 咨询电话: 400-105-8680

3. 官方微信: lampym

