

面向对象基本概念

1. 简介

- (1) 传统面向过程，将要完成的工作分为若干个步骤，或再细分为子步骤，然后从前往后一步步完成，最终达至目标，工作完成，是解决简单问题的自然的思维过程。
- (2) 面向对象，将要完成的工作，拆分为一个一个对象的任务(功能)，各对象各司其职，独立完成任务，功能完成(方法调用)整体的工作。
是设计思想的升华，是解决大规模的复杂问题的良好思想。

2. 面向对象基本概念

(1) 类与方法

类是用来描述” 某一些具有共同特征的对象的概念，单个物体即一对象”，通常对象离不开类，先有类才有对象，类决定对象所拥有的特性。

(2) 类相关成员

```
class People{
    const PI=3.14;//常量
    var $name;//普通变量
    static $instance;//静态变量
    static function sleep(){//静态方法
    }
    function eat(){//普通方法
    }
}

$person = new People();

$pi = People::PI;//常量获取
$name = $person->name;//普通变量获取
$instance = People::$instance;//静态变量获取
People::sleep();//静态方法调用
$person->eat();//普通方法调用
```

(3) 语法对比

面向过程语法：

- 1) 定义变量
- 2) 定义函数
- 3) 使用变量
- 4) 调用函数
- 5) 流程控制

面向对象语法

1) 定义类

定义属性(类、实例变量)

定义方法(类、实例方法)

定义常量(属于类常量)

2) 创建类对象

3) 使用类对象

(4) 对象创建的形式

1) \$对象1=new 类名(); //已定义类

2) \$name = "c1"

\$对象1 = new \$name(); //可变类

3) \$对象1 = new self(); //只能类内部调用

4) 通过对象创建同类对象

\$对象1 = new \$person;

(5) 对象的传值

\$p1 = new Person();

\$p2 = \$p1; 为值传递而非引用传递，只不过复制了一份“对象标识符”，内含有的数据都是引用同一块内存。unset p1后，p2不受影响。

\$p2=&\$p1; 才是引用传递