

第 20 章 面向对象的工具

学习要点:

- 1.OOP 的魔术方法
- 2.类函数和对象函数
- 3.OOP 的反射 API

主讲教师: 李炎恢

合作网站: <http://www.ibeifeng.com>

讲师博客: <http://hi.baidu.com/李炎恢>

PHP通过类和方法等语言结构支持面向对象编程。同时也通过对象相关的函数和内置类为面向对象提供广泛支持。

一. OOP的魔术方法

PHP引入了__autoload()内置方法来自动包含类文件。__autoload()应该被写成单个参数的方法。当PHP引擎遇到试图实例化未知类的操作时,会调用__autoload()方法,并将类名当作字符串参数传递给它。

```
function __autoload($_className) {  
    require $_className.'.class.php';  
}
```

```
$demo = new Computer();
```

PHP采用了__call()内置方法来屏蔽对象调用方法时产生的错误。当对象调用一个不存在的方法时,会自动调用__call()方法。

```
private function __call($_methodName,$args) {  
    echo $_methodName.'方法不存在';  
    print_r($args);  
}
```

```
$computer->go('我',1,'知道');
```

PHP使用__toString()内置方法来打印对象的引用。没有使用__toString()的对象是产生一个错误,当打印对象的时候会自动调用__toString()方法。

```
class Computer {  
    private function __toString() {  
        return '打印对象';  
    }  
}
```

```
echo new Computer();
```

PHP可以在类中定义一个__clone()内置方法来调整对象的克隆行为。当一个对象被克隆的时候自动执行__clone()方法，而复制的对象可以在其方法体内进行调整。

```
class Computer {
    public $_name ;
    public function __clone() {
        $this->_name = 'ibm';
    }
}

$computer1 = new Computer();
$computer1->_name = 'dell';
$computer2 = clone $computer1;
echo $computer2->_name;
```

二. 类函数和对象函数

PHP提供了一系列强大的函数来检测类和对象。以便在第三方系统，运行时知道正在使用的是哪个。

1.class_exists()函数接受表示类的字符串，检查并返回布尔值。如果类存在，返回true，否则返回false。

```
echo class_exists('Computer');
```

2.get_class()函数获取对象的类名，如果不是对象，则返回false。

```
echo get_class($computer);
```

3.get_class_methods()函数获取类中的方法(公共的)，以数组的形式返回出来。

```
print_r(get_class_methods($computer));
```

4.get_class_vars()函数获取类中的字段(公共的)，以数组的形式返回出来

```
print_r(get_class_vars('Computer'));
```

5.get_parent_class()函数获取子类的父类，如果没有返回false。

```
echo get_parent_class('NoteComputer');
```

6.interface_exists()函数确定接口是否存在，如果存在返回true，否则返回false。

```
echo interface_exists('Computer');
```

7.is_a()函数确定对象是否是类或者是否是这类的父类时，返回ture，否则返回false。

```
echo is_a($computer,'Computer');
```

8.is_subclass_of()函数确定对象是否是类的子类，是返回ture，否则返回false。

```
echo is_subclass_of($notecomputer,'Computer');
```

9.method_exists()函数确定对象的方法是否存在，是返回ture， 否则返回false。

```
echo method_exists($computer,'_run');
```

三. OOP的反射API

PHP5的类和对象函数并没有告诉我们类内部的所有一切，而只是报告了它们的公共成员。要充分了解一个类，需要知道其私有成员和保护成员，还要知道其方法所期望的参数。对此，使用反射API。

1.获得反射API的转储信息

```
$rc = new ReflectionClass('Computer');  
Reflection::export($rc);
```

2.获得PHP内置的类库的信息

```
Reflection::export(new ReflectionClass('Reflection'));
```

3.获取类里的某个元素

```
$_rc = new ReflectionClass('Computer');  
echo $_rc->getFileName();  
echo $_rc->getName();
```

感谢收看本次教程！

本课程是由北风网(ibeifeng.com)

瓢城 **Web** 俱乐部(yc60.com)联合提供:

本次主讲老师: 李炎恢

我的博客: hi.baidu.com/李炎恢/

我的邮件: yc60.com@gmail.com