

第 1 章 PHP 概述

学习要点:

- 1.PHP 基础知识
- 2.PHP 的环境配置
- 3.安装三款主流程程序
- 4.PHP 开发工具的选择
- 5.一个简单的示例

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 主讲教师李炎恢



特邀试听学员-吴祁

一. PHP 基础知识

PHP

PHP 是一种目前最流行的服务端 Web 程序开发语言之一。PHP 主要的特点是语法简单易于学习、功能强大、灵活易用。在融合了现代编程语言的一些最佳特性后, PHP、Apache 和 MySQL 的组合已成为 Web 服务器的一种配置标准。学习 PHP 的过程是十分轻松、快捷的。可以说 PHP 已经成为 Web 脚本技术的先驱, 并且会继续引领 Web 技术的潮流。

PHP, 一个嵌套的缩写名称, 是英文超级文本预处理语言 (PHP:Hypertext Preprocessor) 的缩写。PHP 是一种 HTML 内嵌式的语言, PHP 与微软的 ASP 颇有几分相似, 都是一种在服务器端执行的嵌入 HTML 文档的脚本语言, 语言的风格有类似于 C 语言, 现在被很多的网站编程人员广泛的运用。PHP 独特的语法混合了 C、Java、Perl 以及 PHP 自创新的语法。它可以比 CGI 或者 Perl 更快速的执行动态网页。用 PHP 做出的动态页面与其他的编程语言相比, PHP 是将程序嵌入到 HTML 文档中去执行, 执行效率比完全生成 HTML 标记的 CGI 要高许多; 与同样是嵌入 HTML 文档的脚本语言 JavaScript 相比, PHP 在服务器端执行, 充分利用了服务器的性能; PHP 执行引擎还会将用户经常访问的 PHP 程序驻留在内存中, 其他用户再一次访问这个程序时就不需要重新编译程序了, 只要直接执行内存中的代码就可以了, 这也是 PHP 高效率的体现之一。PHP 具有非常强大的功能, 所有的 CGI 或者 JavaScript 的功能 PHP 都能实现, 而且支持几乎所有流行的数据库以及操作系统。

PHP 最初是 1994 年 Rasmus Lerdorf 创建的, 刚刚开始只是一个简单的用 Perl 语言编写的程序, 用来统计他自己网站的访问者。后来又用 C 语言重新编写, 包括可以访问数据库。

在 1995 年以 Personal Home Page Tools (PHP Tools) 开始对外发表第一个版本, Lerdorf 写了一些介绍此程序的文档, 并且发布了 PHP1.0。在这早期的版本中, 提供了访客留言本、访客计数器等简单的功能。以后越来越多的网站使用了 PHP, 并且强烈要求增加一些特性, 比如循环语句和数组变量等等, 在新的成员加入开发行列之后, 在 1995 年中, PHP2.0 发布了。第二版定名为 PHP/FI(Form Interpreter)。PHP/FI 加入了对 mSQL 的支持, 从此建立了 PHP 在动态网页开发上的地位。到了 1996 年底, 有 15000 个网站使用 PHP/FI; 时间到了 1997 年中, 使用 PHP/FI 的网站数字超过五万个。而在 1997 年中, 开始了第三版的开发计划, 开发小组加入了 Zeev Suraski 及 Andi Gutmans, 而第三版就定名为 PHP3。2000 年, PHP4.0 又问世了, 其中增加了许多新的特性。

PHP 的特性

1. 开放的源代码: 所有的 PHP 源代码事实上都可以得到。
2. PHP 是免费的。
3. PHP 的便捷性: PHP 十分便捷
4. 基于服务器端: 由于 PHP 是运行在服务器端的脚本, 可以运行在 UNIX、LINUX、WINDOWS 下。
5. 嵌入 HTML: 因为 PHP 可以嵌入 HTML 语言, 所以学习起来并不困难。
6. 简单的语言: PHP 坚持脚本语言为主, 与 Java 和 C++ 不同。
7. 效率高: PHP 消耗相当少的系统资源。
8. 图像处理: 用 PHP 动态创建图像
9. 面向对象: 在 php4, php5 中, 面向对象 (OOP 技术) 方面都有了很大的改进, 现在 php 完全可以用来开发大型商业程序。

PHP4

2000 年 5 月 22 日, 在正式宣布开发新版本之后大约 18 个月, 发布了 PHP 4.0。许多人都认为 PHP 4 的发布是这种语言在企业级开发环境下的正式亮相, 这个观点也由于 PHP 的迅速普及得到了佐证。仅仅在发布后的几个月内, Netcraft (<http://www.netcraft.com/>) 估计就有超过 360 万个站**安装**了 PHP。

PHP 4 添加了以下几项对该语言的企业级改进:

1. **改进了资源处理**。可扩展性是版本 3.x 的主要缺点之一, 这主要是因为设计者低估了这种语言, 没考虑到它会大量用于大规模应用程序。最初并没有打算用这种语言开发企业级网站, 但后来确实有这样的尝试, 这就使得开发人员开始重新考虑这种语言的机制。
2. **面向对象的支持**。版本 4 在一定程度上加入了面向对象的功能, 尽管一般认为实现得 not 怎么样。不过, 对于使用传统面向对象程序设计 (OOP) 语言的用户来说, 这个新特性在吸引这些用户方面起到了非常重要的作用。除了对象重载和运行时类信息, PHP 还支持标准类和对象开发方法。
3. **内置的会话处理支持**。版本 3.x 通过第三方包 PHPLIB (<http://phplib.sourceforge.net>) 来支持 HTTP 会话处理, 在版本 4 中 HTTP 会话处理则是内置的功能。这个特性使得开发人员可以相当高效轻松地跟踪用户活动和偏好。。
4. **加密**。MCrypt (<http://mcrypt.sourceforge.net>) 库引入到默认发行包中, 为用户提供

了完全加密和散列加密，使用的加密算法包括 Blowfish、MD5、SHA1 和三重 DES 等。

5. **ISAPI 支持**。对 ISAPI 的支持使用户能够将 PHP 与微软的 IIS Web 服务器结合使用。

6. **内置 COM/DCOM 支持**。对 Windows 用户来说，另一个好处是 PHP 4 能够访问和实例化 COM 对象。这项功能扩展了与 Windows 应用程序的互操作性。

7. **内置 Java 支持**。这也是 PHP 在互操作性方面的一大进步，版本 4.0 支持 PHP 应用程序绑定 Java 对象。

8. **与 Perl 兼容的正则表达式 (PCRE) 库**。Perl 语言一直在字符串解析领域雄霸天下，占据着统治地位。开发人员知道，如果想让 PHP 得到广泛认可，强大的正则表达式功能会起到重要作用。他们的做法只是集成 Perl 的功能，而不是重新开发，并将 PCRE 库的包集成在 PHP 的默认发行包中。

除了这些特性外，版本 4 还添加了几百项功能，大大提升了这种语言的能力。本书中，我们将讨论其中大部分功能。

在 PHP 语言的发展历程上，PHP 4 代表着一次巨大的飞跃。这个新版本带来的新功能、强大能力和可扩展性对开发新手和老手都产生了不小的震动。但 PHP 开发团队并不满足于长时间只拥有这一项成果，于是不久以后他们就开始着手开展另一个里程碑式的任务，要推出一门领导 Web 脚本领域的语言：PHP 5。

PHP5

版本 5 是 PHP 语言发展历程中的另一座分水岭。虽然前面的主要版本已经增加了许多库，版本 5 则在现有的功能上又进行了许多改进，并且增加了成熟的编程语言架构才有的一些特性。

1. **极大地提高了面向对象能力**。PHP 的面向对象架构得到了改进，这是版本 5 最突出的特点。版本 5 增加了很多功能，如显式构造函数和析构函数、对象克隆、类抽象、变量作用域和接口等。另外，PHP 在对象管理方面也有重大改进。第 6 章和第 7 章将详细介绍这些内容。

2. **try/catch 异常处理**。具有讽刺意味的是，在结构化编程语言中，设计错误处理策略本身却非常容易出错，而且很难保持一致。为了解决这个问题，版本 5 开始支持异常处理。在许多语言中，如 C++、C#、Python 和 Java 等，异常处理长期以来一直都是错误管理方面的中流砥柱，它为建立标准化的错误报告逻辑提供了一种绝佳的方法。

3. **改进的 XML 和 Web 服务支持**。现在的 XML 支持建立在 libxml2 库基础上，还引入一个很新但非常有前途的扩展包来解析和处理 XML，即 SimpleXML。此外，PHP 5 还支持 SOAP 扩展。第 20 章将介绍这两个新扩展，并介绍一些很棒的第三方 Web 服务扩展。

4. **对 SQLite 的内置支持**。开发人员总是乐于提供更多选择，他们为功能强大但很简洁的 SQLite 数据库服务器 (<http://www.sqlite.org/>) 提供了支持。如果开发人员需要一些重量级数据库产品中才有的特性，同时不希望带来相应的管理开销，SQLite 就能为这些开发人员提供一个方便的解决方案。第 22 章将介绍 PHP 为这个强大的数据库引擎提供的支持。

PHP6

1. **Unicode 支持**。增加了本地 Unicode 支持，使得构建和维护多语言应用程序变得容易的多。

2. **安全改进**。已经做了大量有关安全性的改进，基于这些改进，应该能显著遏制安全相关问题的泛滥，说实话，这些问题不能归于语言，而应归咎于只会东拼西凑的没有经验的程

序员。

3. **新语言特性和构造**。增加了许多新的语法特征，其中最突出的就是 64 位整数类型、经过“改造”的用于多维数组的 foreach 循环构造。以及对标签的 break 的支持。

二. 环境配置

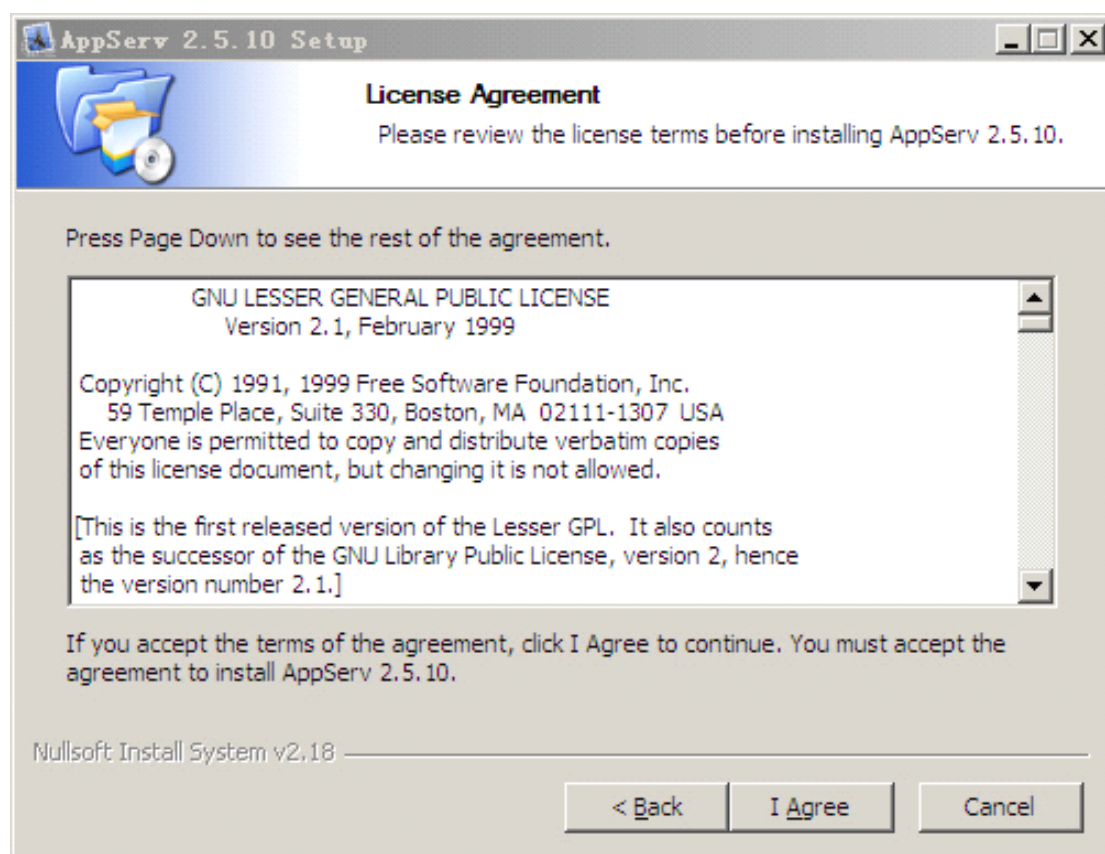
1. Windows/Linux, Apache , php , mysql
2. Linux 下安装方式(Lamp 环境)
 - 原码包安装方式
 - Rpm 包安装方式
 - Linux 下集成软件包 xampp
3. Windows 下安装方式
 - Appserv 官网: <http://www.appservnetwork.com/>
 - phpStudy
 - Wamp
 - Xampp

这里采用 Appserv 来安装

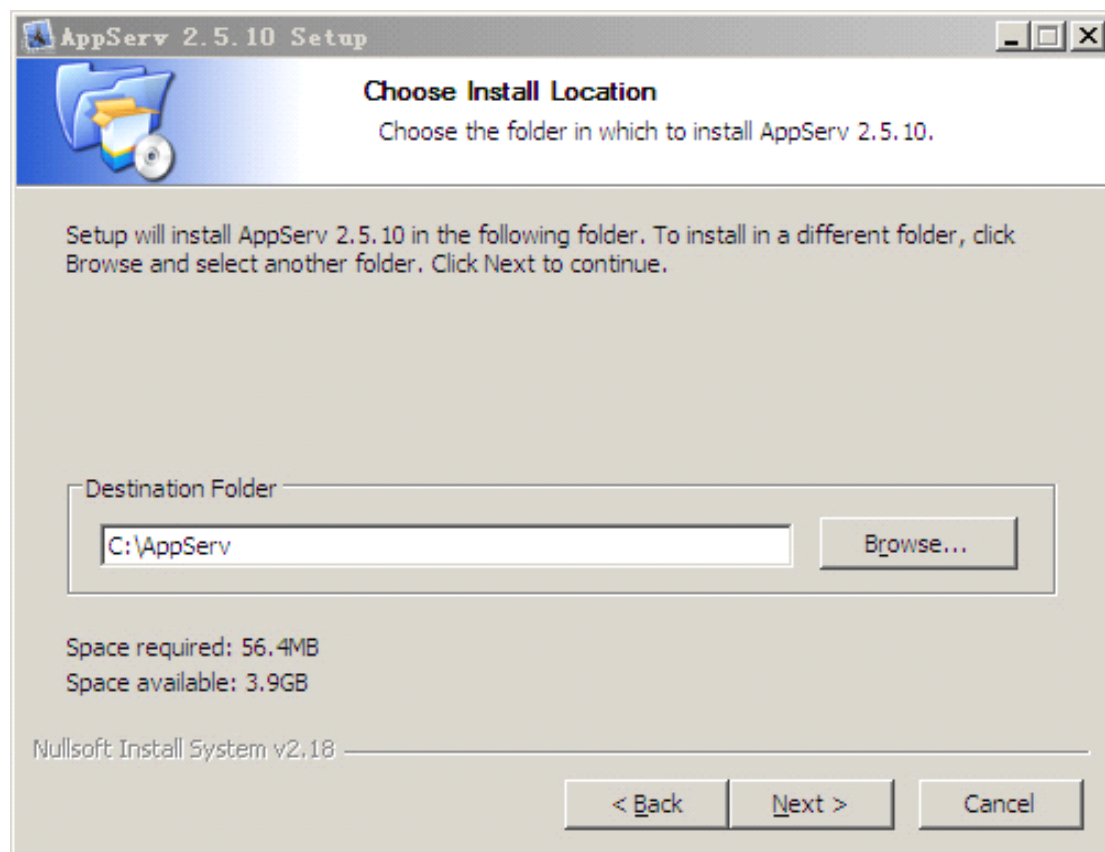
(下一步)



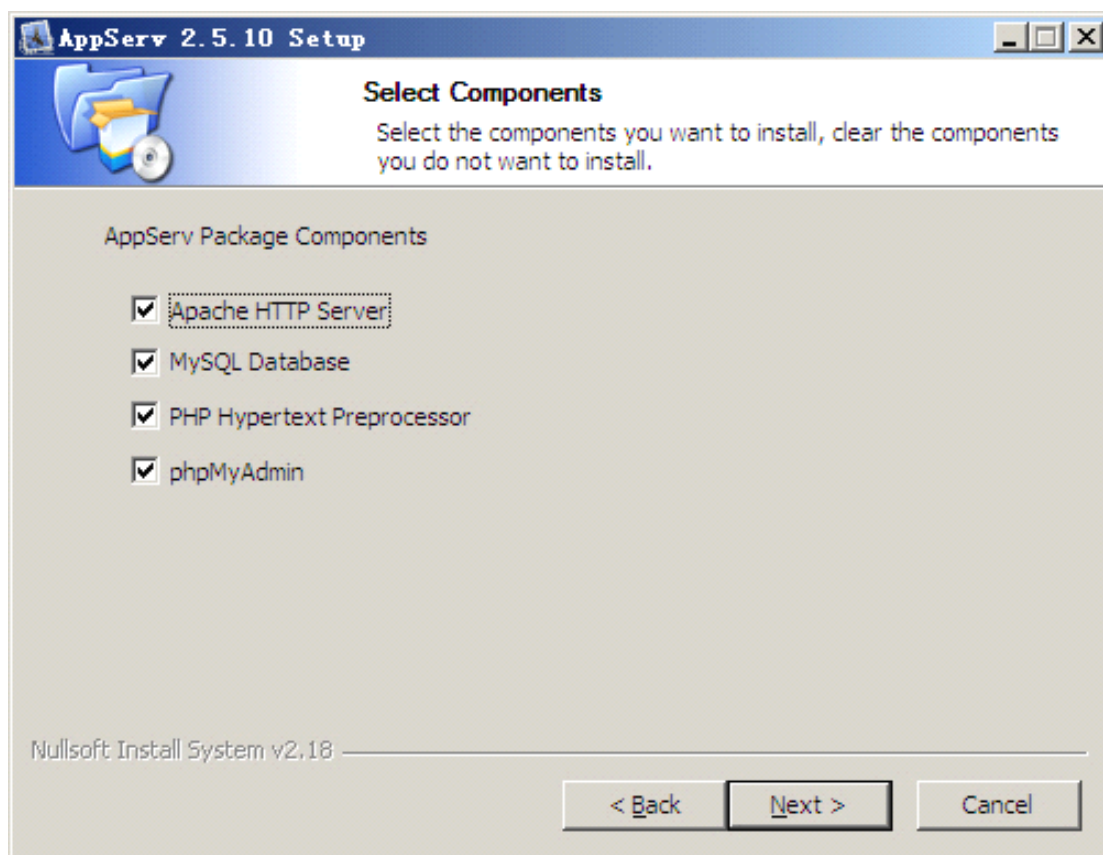
(下一步)



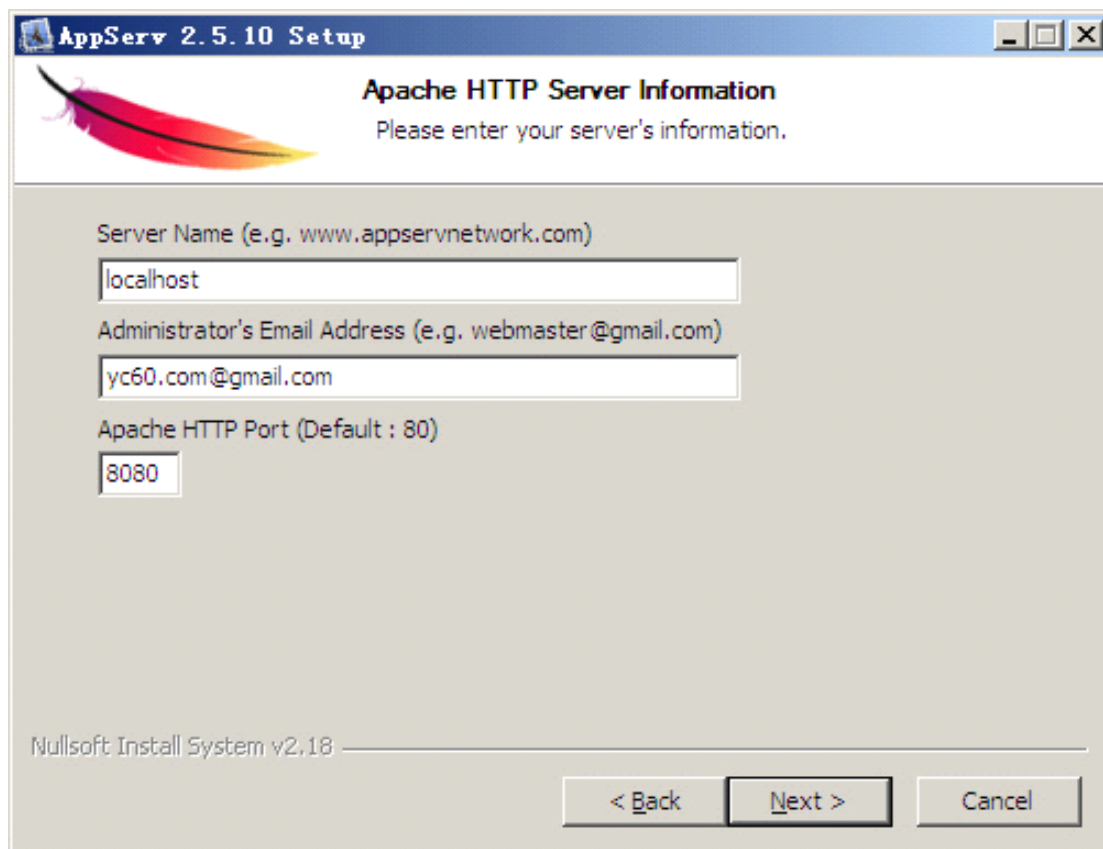
(下一步)



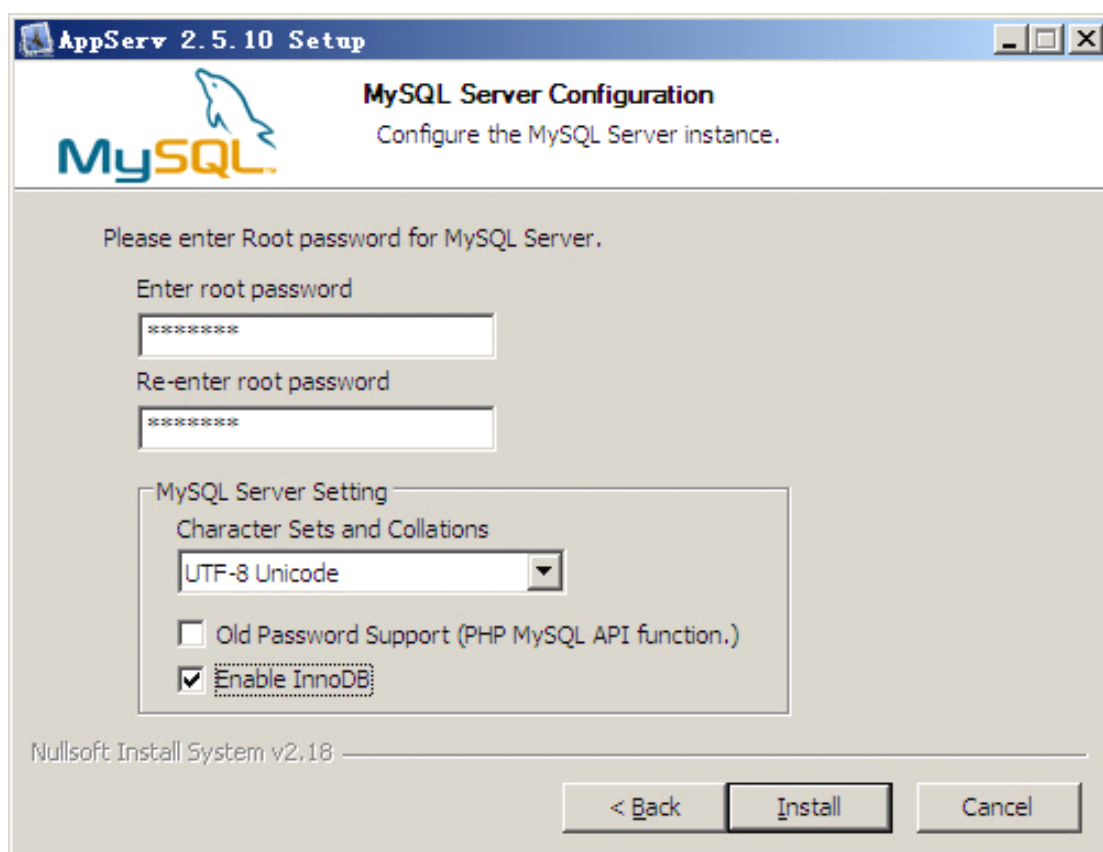
(下一步)



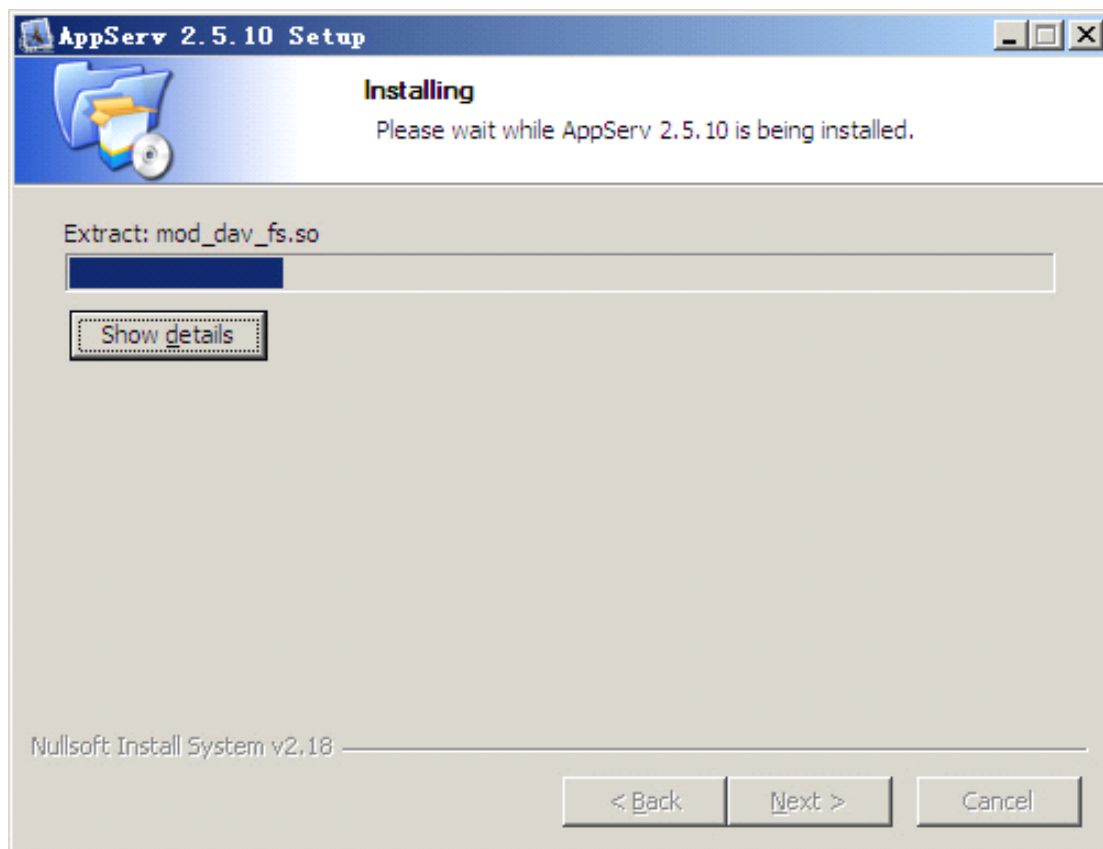
(下一步)



(下一步)



(下一步)



(安全警告--解除)



(完成)



三. 安装第一 Guest 项目、安装 discuz 论坛、安装 ecshop 网店

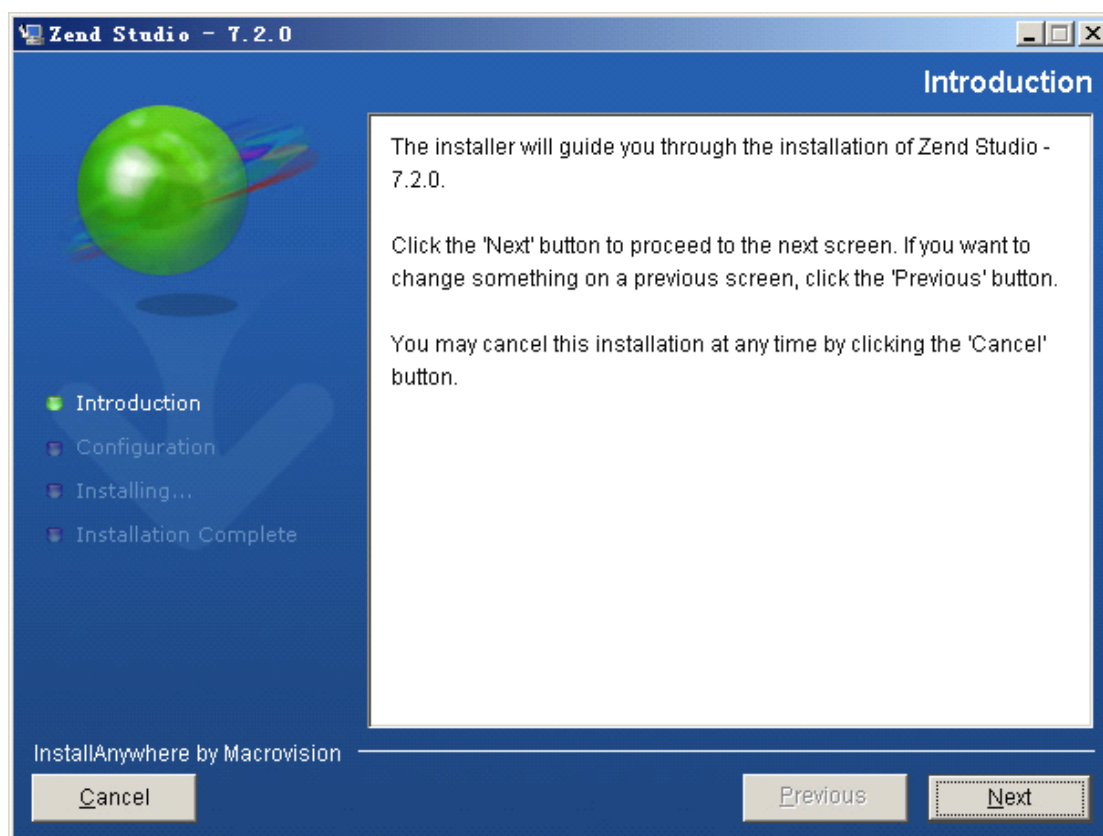
去 <http://www.discuz.com> 下载论坛程序，然后安装

去 <http://www.ecshop.com> 下载网店程序，然后安装

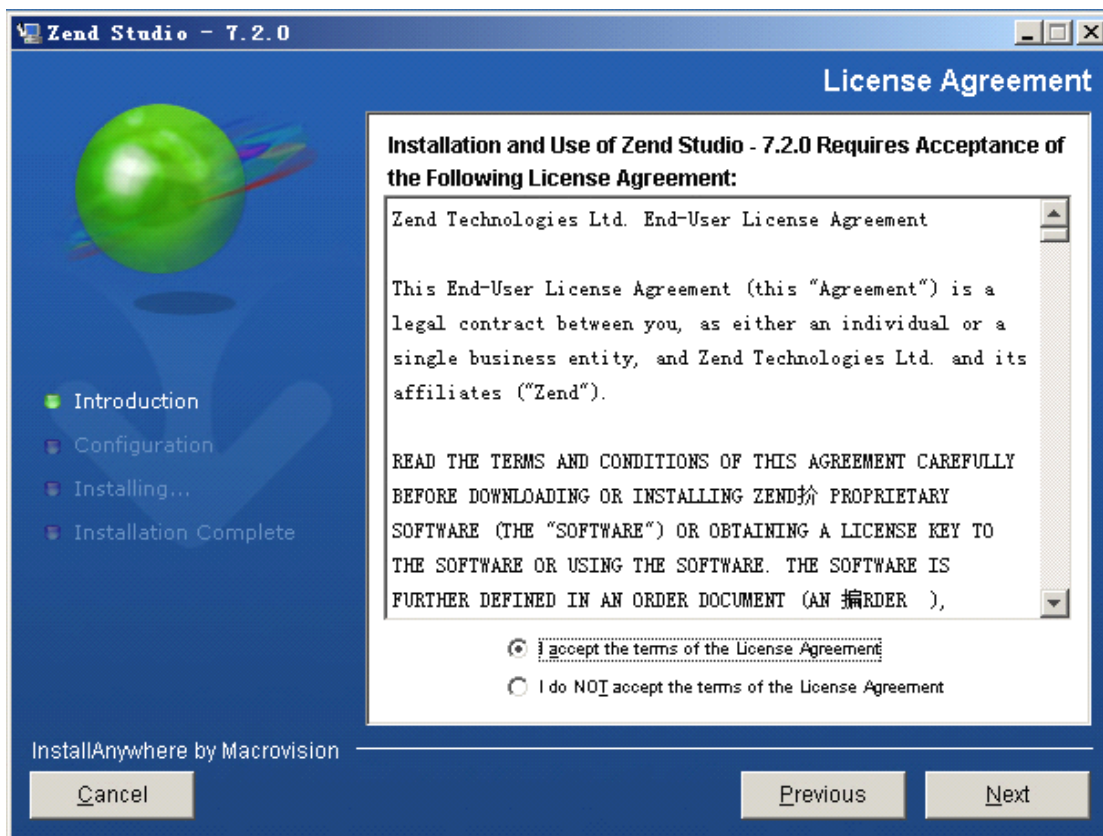
四. 开发工具

Zend Studio 7.2.0

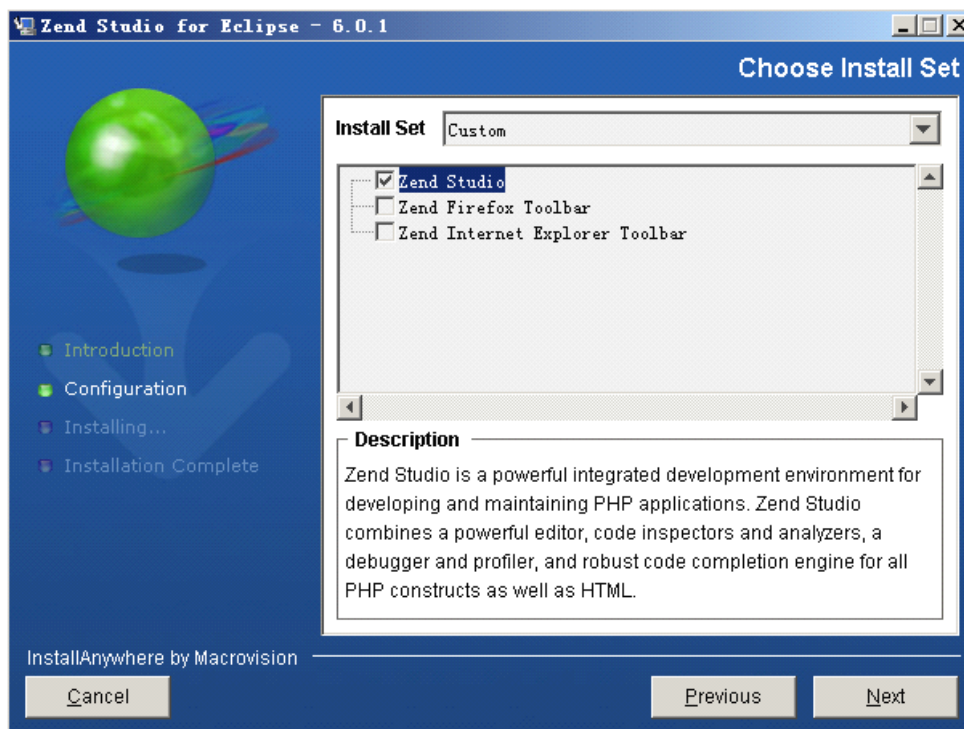
(下一步)



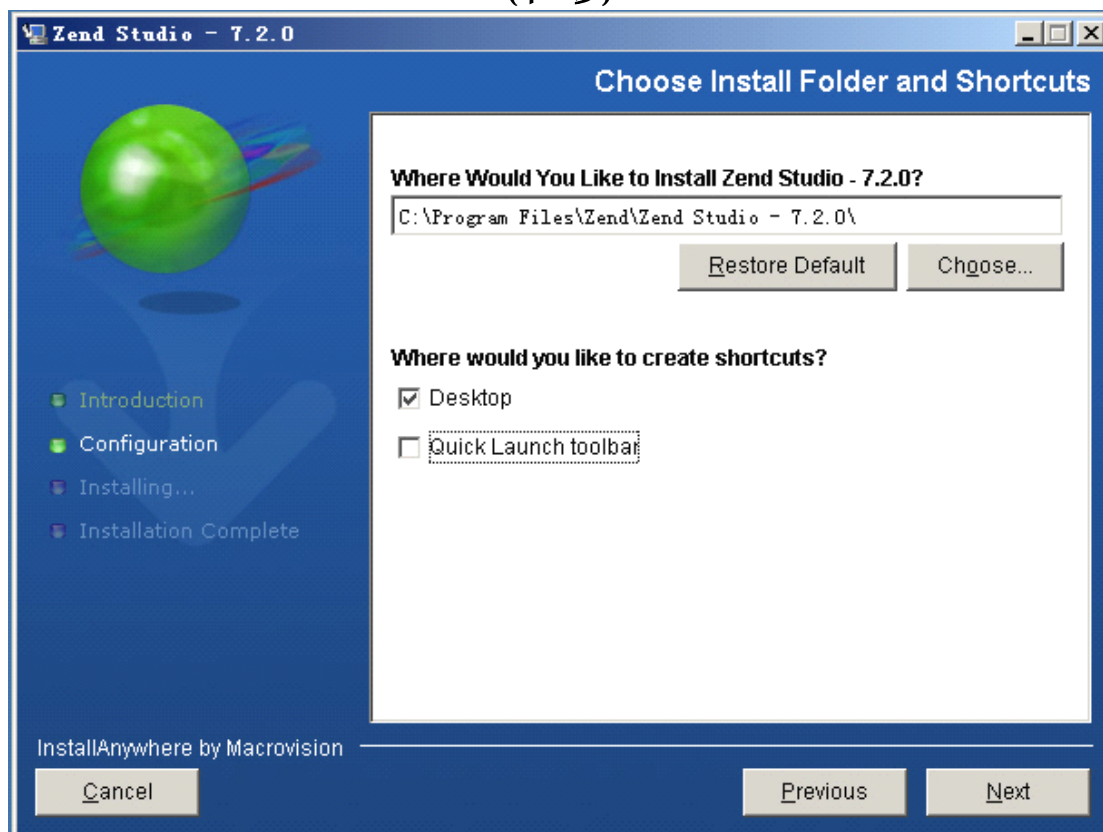
(下一步)



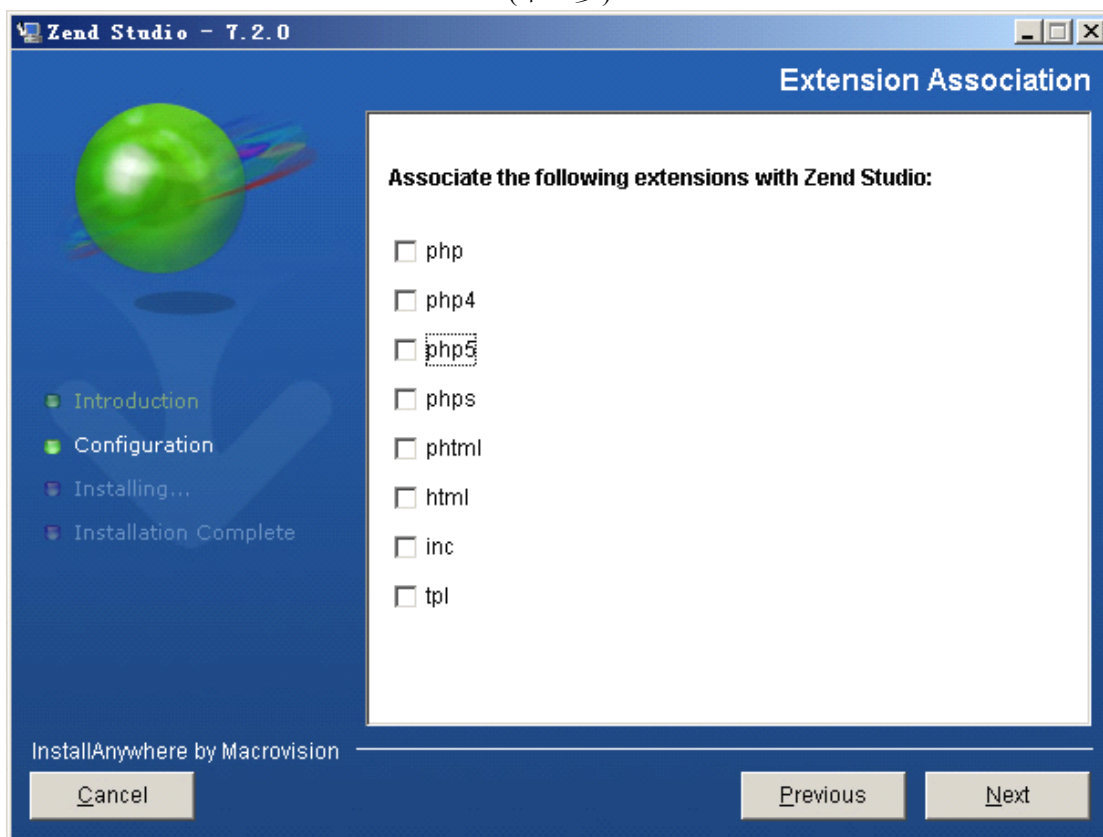
(下一步)



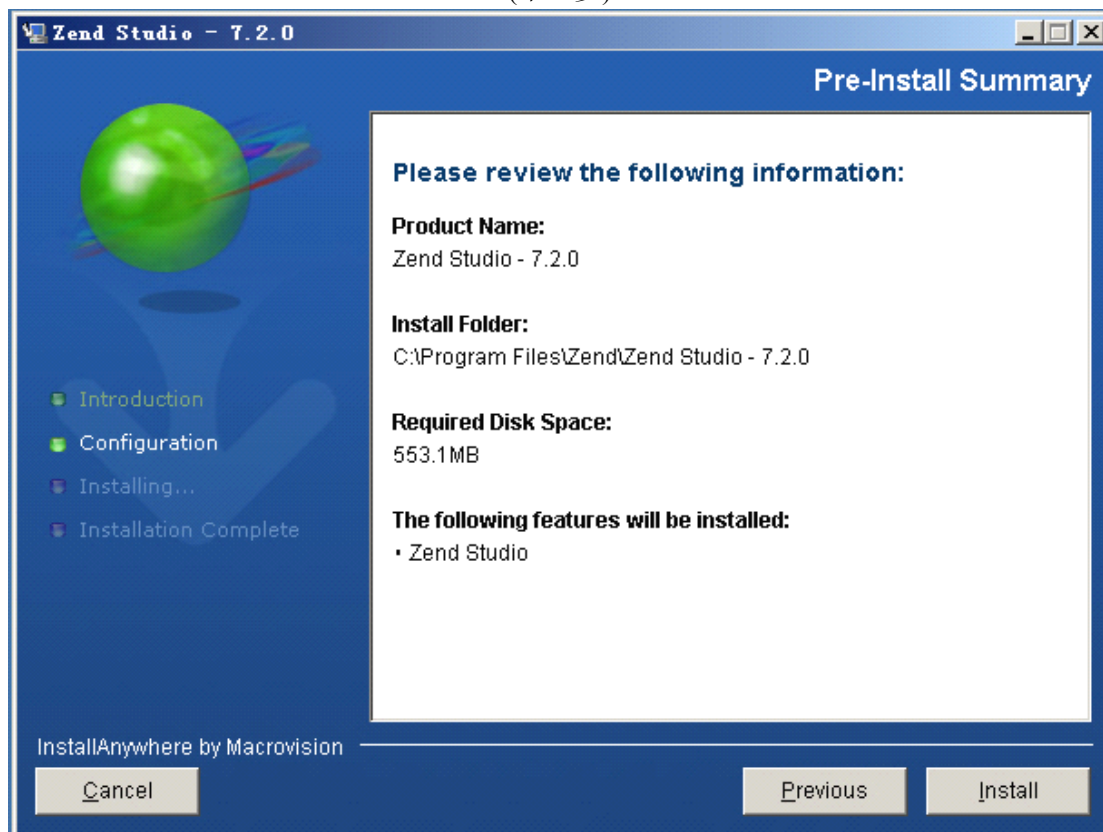
(下一步)



(下一步)



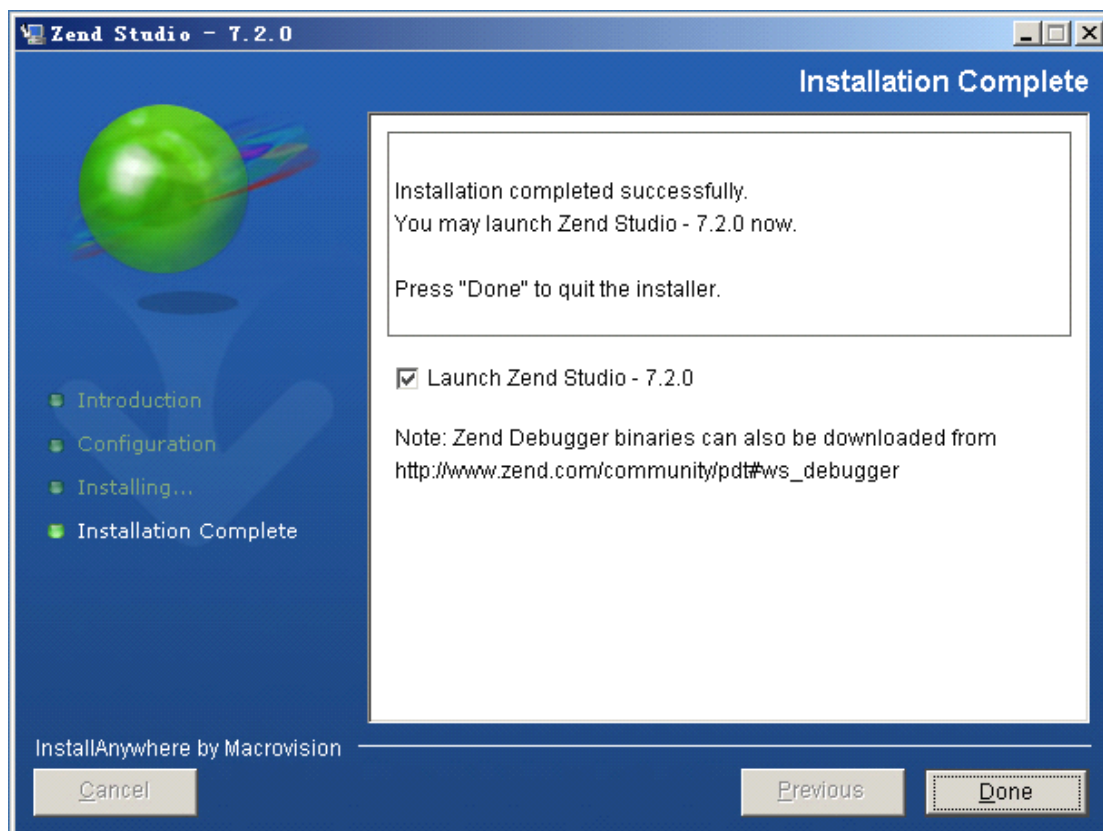
(下一步)



(安装)



(完成)



五. 示例程序

请查看 `orderform.php` 和 `postorder.php` 两个文件的源代码，并且执行它。

学习注意点：

1. 不用刻意的记单词，记不住的查手册
2. 第一阶段不要深入理解很多知识点的原理，而是以完成项目为目标

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

第2章 基本语法

学习要点:

1. Zend Studio 两个小问题
2. 在 Web 页面中嵌入 PHP 代码
3. 标识符与变量、常量
4. 访问表单变量

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>

一. Zend Studio 两个小问题

1. 新建文档的模板设置:

Window -> Preferences -> PHP -> Editor -> Templates -> New simple PHP file
//这个可以修改每次新建文件的起始代码

2. 自动提示代码速度很慢:

Window -> Preferences -> PHP -> Editor -> Content Assist -> Auto Activation
//将代码自动提示的速度 500 设置成 50

二. 在 Web 页面中嵌入 PHP 代码

```
<?php  
echo "<p>我的第一个PHP程序!</p>";  
?>
```

这是一段 PHP 服务器端脚本, 经过服务器解释执行后生成 HTML 文件!

其他几种格式如下:

- 简短风格:

```
<? echo "<p>My PHP!</p>"; ?>
```

- Script 风格:

```
<script language="php">echo "<p>My PHP!</p>"; </script>
```

- ASP 风格:

```
<% echo "<p>My PHP!</p>"; %>
```

//默认情况下是被禁止的, 需修改 asp_tags 选项, 在 PHP.ini 里

注释: 单行// 多行/**/ shell #

向浏览器中输出:

`echo()`、`print()`、`printf()`、`sprintf()`

`echo`、`print`、`printf` 本身是函数,即函数()。但这里的输出函数可以省略括号,用空格+所需显示的字符串或变量。

`echo` 和 `print` 功能几乎相同,而 `echo` 运行速度上比 `print` 稍稍快一点。因为 `print` 有返回值。

//`echo` 不返回任何值(void),`print` 返回的是整型(integer)

`printf()`和 `sprintf()`是 C 语言模式,例如:`printf("我今天买了%d 套视频光盘",5);`

//他们之间的不同点是,`printf` 返回的是整型(integer),而 `sprintf` 返回的字符串(string)

//`printf` 可以在浏览器直接输出,而 `sprintf` 需要 `echo` 将它输出

常用类型指示符

类型	描述
<code>%b</code>	整数,显示为二进制
<code>%c</code>	整数,显示为 ASCII 字符
<code>%d</code>	整数,显示为有符号十进制数
<code>%f</code>	浮点数,显示为浮点数
<code>%o</code>	整数,显示为八进制数
<code>%s</code>	字符串,显示为字符串
<code>%u</code>	整数,显示为无符号十进制数
<code>%x</code>	整数,显示为小写的十六进制数
<code>%X</code>	整数,显示为大写的十六进制数

三. 标识符与变量

标识符是变量的名称,关于标识符,PHP 定义了一些简单的规则:

- 标识符可以是任何长度,而且可以由任何字母、数字、下划线组成。
- 标识符不能以数字开始。
- 在 PHP 中,标识符是区分大小写的。
- 一个变量名称可以与一个函数名称相同。

变量赋值:

```
$sum=0;  
$total=1.22;  
$sum=$total;
```

变量的数据类型

PHP 支持如下所示的基本数据类型:

Integer(整数)、Float(浮点数,也叫 Double,双精度)、String(字符串)、Boolean(布尔)、Array(数组)、Object(对象)。

类型强度

PHP 是一种非常弱的类型语言。在大多数编程语言中，变量只能保存一种类型的数据，而且这个类型必须在使用变量之前声明。而在 PHP 中，变量的类型是由赋值给变量的值确定的。

类型转换

使用类型转换，可以将一个变量或值转换成另一种类型。

```
$sum=0;  
$total=(float)$sum;
```

检测变量

大部分的可变函数都是用来测试一个函数的类型的。PHP 中有两个最常见的函数，分别是 `gettype()` 和 `settype()`。

这两个函数返回的 `string` 类型，也就是变量的类型字符串。

`isset()` 和 `unset()` 用来判断一个变量是否存在，返回的是布尔值 `true` 或 `false`。

`empty()` 用来判断一个变量的值是否为空，如果为空则为 `true` 否则为 `false`。

换句话说，`""`、`0`、`"0"`、`NULL`、`FALSE`、`array()`、`var $var`；以及没有任何属性的对象都将被认为是空的

PHP 还提供了一些特定类型的测试函数。每一个函数都使用一个变量座位其参数，并且返回 `true` 或 `false`。

`is_array()`、`is_double()`、`is_float()`、`is_real()`、`is_long()`、`is_int()`、`is_integer()`、`is_string()`、`is_object()`、`is_resource()`、`is_null()`、`is_numeric()`

可以通过调用一个函数来实现转换变量数据类型的目的。

`intval()`、`floatval()`、`strval()`;

理解变量的作用域

作用域是指在一个脚本中某个变量在哪些地方可以使用或可见。

内置超级全局变量可以在脚本的任何地方使用和可见。

常量，一旦被声明，将可以在全局可见。也就是说，他们可以在函数内外使用。

在一个脚本中声明的全局变量在整个脚本中是可见的，但在函数内部。

函数内部使用的变量声明为全局变量时，其名称要与全局变量名称一致。

在函数内部创建并被声明为静态的变量无法在函数外部可见，但是可以在函数的多次多次执行过程中保持改值

在函数内部创建的变量对函数来说是本地的，而当函数终止时，该变量也就不存在了。

超级全局变量：

<code>\$GLOBALS</code>	所有全局变量数组
<code>\$_SERVER</code>	服务器环境变量数组
<code>\$_GET</code>	通过 GET 方法传递给该脚本的变量数组
<code>\$_POST</code>	通过 POST 方法传递给该脚本的变量数组

<code>\$_COOKIE</code>	cookie 变量数组
<code>\$_FILES</code>	与文件上载相关的变量数组
<code>\$_ENV</code>	环境变量数组
<code>\$_REQUEST</code>	所有用户输入的变量数组
<code>\$_SESSION</code>	会话变量数组

常量

常量一旦被定义之后，就不能再次更改。

```
define("TOTAL",199);
```

除了自定义常量外，PHP 还预定了许多常量。了解这些常量的简单方法就是运行 `phpinfo()` 命令。

变量和常量的另一个差异在于常量只可以保存布尔值、整数、浮点数或字符串数据。这些类型都是标量数据。

四. 访问表单变量

在 PHP 脚本中，可以以 PHP 变量的形式访问每一个表单域，其中 PHP 变量名称必须与表单域的名称一致。你可以很容易识别 PHP 的变量名称，因为他们都是以 \$ 符号开始的。（漏掉这个 \$ 符号是一个常见的编程错误。）

根据 PHP 版本和设置的不同，通过变量，可以有 3 种方法来访问表单数据。这些方法并没有正式的名称，因此我们给他们定义了 3 个昵称，分别是简短风格、中等风格和冗长风格。在任何情况下，一个页面上提交给 PHP 脚本的每一个表单域在 PHP 脚本中都是可以使用的。

```
$userName                //简短风格
$_POST["userName"]        //中等风格  $_Get["QQName"];
$http_post_vars["userName"] //冗长风格  $_
```

- 简短风格：非常方便，但容易引起混淆，不推荐
- 中等风格：4.1.0 版后支持，推荐的
- 冗长风格：最详细，但它已经过时，长远角度看，以后会被剔除

字符串连接：使用点就可以连接".";

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

第3章 操作符与控制结构

学习要点:

1. 字符串插入
2. 操作符
3. 控制结构

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>

一. 字符串插入

为了给开发人员处理字符串值提供最大的灵活性, PHP 为字面插入和内容插入提供了一种方法。

双引号提供了最大的灵活性, 原因是变量和转移序列都会得到相应的解析。

```
<?php
$username = "吴祁";
echo "His name is $username";
echo "<br />";
//中文会出现一些问题
echo "他的名字叫$username, 他19岁了,已经长大成人了! ";
echo "<br />";
//可以采用字符串连接的方式解决
echo "他的名字叫".$username.", 他19岁了,已经长大成人了! "
//转义字符可以得到解析
echo "虽然他的QQ号有很多女生, \n但一个都不属于他";
?>
```

部分常用的转义字符

转义序列	描述
\n	换行符
\r	回车
\t	水平制表符
\\	反斜杠
\\$	美元符
\"	双引号

单引号会按照声明的原样解释，解析字符串时，变量和转义序列都不会进行解析。

```
<?php
    echo '吴祁的变量名为: $userName,转义字符\n在单引号中无效'
?>
```

二. 操作符

操作符是用来对数组和变量进行某种操作运算的符号。

算术操作符

操作符	名称	示例
+	加	$\$a + \b
-	减	$\$a - \b
*	乘	$\$a * \b
/	除	$\$a / \b
%	取余	$\$a \% \b

复合赋值操作符

操作符	使用方法	等价于
+=	$\$a += \b	$\$a = \$a + \$b$
-=	$\$a -= \b	$\$a = \$a - \$b$
*=	$\$a *= \b	$\$a = \$a * \$b$
/=	$\$a /= \b	$\$a = \$a / \$b$
%=	$\$a \% = \b	$\$a = \$a \% \$b$
.=	$\$a .= \b	$\$a = \$a . \$b$

前置递增递减和后置递增递减运算符：

```
$a=++$b;
$a=$b++;
$a=--$b;
$a=$b--;
```

比较运算符

操作符	名称	使用方法
-----	----	------

==	等于	\$a==\$b
===	恒等	\$a=== \$b
!=	不等	\$a!=\$b
!==	不恒等	\$a!== \$b
<>	不等	\$a<>\$b
<	小于	\$a<\$b
>	大于	\$a>\$b
<=	小于等于	\$a<=\$b
>=	大于等于	\$a>=\$b

注：恒等表示只有两边操作数相等并且数据类型也相当才返回 true;

例如：0=="0" 这个返回为 true ， 因为操作数相等

0=== "0" 这个返回为 false， 因为数据类型不同

逻辑运算符

操作符	使用方法	使用方法	说明
!	非	!\$b	如果 \$b 是 false, 则返回 true; 否则相反
&&	与	\$a&&\$b	如果 \$a 和 \$b 都是 true, 则结果为 true; 否则为 false
	或	\$a \$b	如果 \$a 和 \$b 中有一个为 true 或者都为 true 时，其结果为 true; 否则为 false
and	与	\$a and \$b	与 && 相同，但其优先级较低
or	或	\$a or \$b	与 相同，但其优先级较低

操作符 "and" 和 "or" 比 && 和 || 的优先级要低。

三元操作符

Condition ? value if true : value if false

示例：(\$grade>=50 ? "Passed" : "Failed")

错误抑制操作符:

$\$a=@(57/0);$

除数不能为 0，会出错，所以加上@避免出现错误警告。

数组操作符

操作符	使用方法	使用方法	说明
+	联合	! $\$b$	返回一个包含了 $\$a$ 和 $\$b$ 中所有元素的数组
==	等价	$\$a\&\&\b	如果 $\$a$ 和 $\$b$ 具有相同的元素，返回 true
===	恒等	$\$a\ \b	如果 $\$a$ 和 $\$b$ 具有相同的元素以及相同的顺序，返回 true
!=	非等价	$\$a$ and $\$b$	如果 $\$a$ 和 $\$b$ 不是等价的，返回 true
<>	非等价		如果 $\$a$ 和 $\$b$ 不是等价的，返回 true
!==	非恒等	$\$a$ or $\$b$	如果 $\$a$ 和 $\$b$ 不是恒等的，返回 true

操作符的优先级和结合性:

一般地说，操作符具有一组优先级，也就是执行他们的顺序。

操作符还具有结合性，也就是同一优先级的操作符的执行顺序。这种顺序通常有从左到右，从右到左或者不相关。

下面给出操作符优先级的表。最上面的操作符优先级最低，按着表的由上而下的顺序，优先级递增。

操作符优先级

结合性	操作符
左	,
左	Or
左	Xor
左	And
右	Print
左	$= + = - = * = / = \% = \& = = \wedge = \sim = < < = > > =$
左	? :

左	
左	&&
左	
左	^
左	&
不相关	== != = = = != =
不相关	< < = > > =
左	< < > >
左	+ - .
左	* / %
右	! ~ ++ -- (int)(double)(string)(array)(object) @
右	[]
不相关	New
不相关	()

为了避免优先级混乱，可以使用括号避开优先级。

三. 控制结构

如果我们希望有效地相应用户的输入，代码就需要具有判断能力。能够让程序进行判断的结构称为条件。

if 条件判断语句

```
if (条件判断){
    //.....
}
```

```
if (条件判断){
    //....
}
else {
    //....
}
```

```
if (条件判断) {
    //...
}
```



```
elseif {  
    //...  
}  
elseif {  
    //....  
}  
else {  
    //....  
}
```

switch 语句

```
switch (变量) {  
    case "值 1":  
        //...  
        break;  
    case "值 2":  
        //...  
        break;  
    case "值 N":  
        //...  
        break;  
    default:  
        //....  
        break;  
}
```

while 循环

```
while (条件) {  
    //....  
}
```

for 循环

```
for (初始值;条件;计数器) {  
    //....  
}
```

除了 for 循环外，PHP 还提供了 foreach 循环，它专门用于数组的使用。我们在数组中详细介绍它。

do while 循环

```
do {  
    //...  
} while(条件);
```

如果希望停止一段代码的执行，根据所需要达到的效果不同，可以有 3 中方法实现。

第一种: break; 退出循环; 第二种是 exit ;退出程序 第三种是 continue;退出当前循环

```
if (条件)
{
    break; //continue
}
```

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

第4章 数学运算

学习要点:

1. 数值数据类型
2. 随机数
3. 格式化数据
4. 数学函数

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>

在大多数程序设计语言中, 数值运算都是最基本的元素之一。数值运算允许程序员完成加法到高级计算等各种操作。尽管 PHP 只是设计为用于创建动态 HTML 文档, 但如今它已经发展成为一门通用的编程语言, 包括了一个强大而灵活的数学实现。

一. 数值数据类型

PHP 中, 数字或数值数据以及数学函数的使用很简单。基本来说, 要处理两种数据类型: 浮点数和整数。浮点数和整数值的内部表示分别是 C 数据类型 `double` 和 `int`。类似于 C, PHP 中这些数据类型遵循同样的一组规则。

PHP 是一种松散类型的脚本语言, 变量可以根据计算的需求改变数据类型。这就允许引擎动态地完成类型转换。所以, 如果计算中包含数值和字符串, 字符串会在完成计算之前转换为数值, 而数值则会在与字符串连接之前转换为字符串。

```
<?php
    $a = '5';
    $b = 7 + $a;
    echo "7 + $a = $b";
?>
```

PHP 提供了大量函数来检查变量的数据类型。其中有3个函数可以检查变量是否包含一个数字值, 或更具体地, 可以检查变量是一个浮点数还是一个整数。函数 `is_numeric()` 可以检查作为参数传入的值是否是数值。

函数 `is_int()` 和 `is_float()` 用于检查具体的数据类型。如果传入一个整数或浮点数, 这些函数会返回 `true`, 否则返回 `false`, 即使传入一个有合法数值表示的字符串也会返回 `false`。也可以强制引擎改变数据类型。这称之为类型强制转换, 可以在变量或值前面增加 `(int)`、`(integer)`、`(float)`、`(double)` 或 `(real)` 实现, 也可以通过使用函数 `intval()` 或 `floatval()` 来实现。

二. 随机数

随机数本身就是一门科学。已经有很多不同的随机数生成器实现。PHP实现了其中两种：`rand()`和`mt_rand()`。`rand()`函数是libc(构建PHP所用编译器提供的基本库之一)中定义的随机函数的一个简单包装器。`mt_rand()`是一个很好的替代实现，提供了很多精心设计的特性，而且`mt_rand()`甚至比libc中的版本还要快。

两个函数都提供一些函数来得到MAX_RANDOM的值。`rand()`的是`getrandmax()`，`mt_rand()`的是`mt_getrandmax()`;

三. 格式化数据

除了警告、错误等信息外，PHP的大部分输出都是利用`echo`、`print()`和`printf()`之类的函数生成的。这些函数将参数转换成一个字符串，并发给客户端应用程序。

`number_format()`函数可以把整数和浮点数值转换为一种可读的字符串表示。

```
<?php
    $i = 123456;
    $si = number_format($i,2,".",",");
    echo $si;
?>
```

四. 数学函数

`abs()` 绝对值
`floor()` 舍去法取整
`ceil()` 进一法取整
`round()` 四舍五入
`min()` 求最小值或数组中最小值
`max()` 求最大值数组中最大值

PS：过一遍手册函数库

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

第5章 数组

学习要点:

1. 什么是数组
2. 自定义键数组
3. 数组里的数组
4. 数组的排序
5. 数组的指针操作
6. 统计数组个数
7. 将数组转换成标量变量

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>

一. 什么是数组

传统上把数组(array)定义为有一组有某种共同特性的元素, 包括相似性和类型。每个元素由一个特殊的标识符来区分, 称之为键(key); 而每个键对应一个值(value)。

李彦宏	周鸿祎	马云	俞敏洪	李开复	吴祁
-----	-----	----	-----	-----	----

依照上表, 我们创建一个数组。

第一种:

```
$userNames=array('李彦宏','周鸿祎','马云','俞敏洪','李开复','吴祁');
```

这是索引数组初始化:

数字索引的初始值是从 0 开始计算的:

`userNames[0]`--`userNames[5]`, 代表这6个人的名字。

第二种:

通过 `range()` 函数自动创建一个数组

```
$numbers=range(1,10)    $letters=range('a','z')
```

可以加第三个参数 `$numbers=range(1,10,2)`, 第三个参数表示步长

访问数组的内容:

要访问一个变量的内容, 可以直接使用其名称。如果该变量是一个数组, 可以使用变量名称和关键字或索引的组合来访问其内容。

`$numbers[0]`、`$numbers[1]`、`$numbers[2]`;等。

改变数组的值:

```
$numbers[0]="Lee";
```

使用循环访问数组

由于数组使用有序的数字作为索引，所以使用一个 for 循环就可以很容易地显示数组的内容：

```
for ($i=0;$i<10;$i++) {  
    echo $numbers[$i];  
}
```

也可以使用 foreach 循环来遍历数组：

```
foreach ($numbers as $value) {  
    echo $value;  
}
```

测试是否为数组变量：is_array();

print_r 函数：打印关于变量的易于理解的信息

二. 自定义键数组

初始化相关数组

```
$ages=array('吴祁'=>19,'李炎恢'=>27,'胡心鹏'=>23);
```

访问数组元素

```
$ages["吴祁"];  
$ages["李炎恢"];  
$ages["胡心鹏"];
```

追加数组：首先，创建只有一个元素的数组，然后追加两个元素。

```
$ages=array("吴祁"=>19);  
$ages["李炎恢"]=27;  
$ages["胡心鹏"]=23;
```

直接添加数组：无需创建，直接添加，添加第一个元素自动建立数组。

```
$ages["吴祁"]=19;  
$ages["李炎恢"]=27;  
$ages["胡心鹏"]=23;
```

使用循环语句：因为相关数组的索引不是数字，因此无法使用 for 循环语句中使用一个简单的计数器对数组进行操作。但是可以使用 foreach 循环或 list()和 each()结构。

```
foreach ($ages as $key=>$value) {  
    echo $key."=>".$value."<br />";  
}
```

使用 each()结构打印\$ages 数组的内容：each()函数返回数组的当前元素，并将下一个元素作为当前元素。

```
while (!!$element=each($ages)) {
```

```
echo $element["key"];  
echo "=>";  
echo $element["value"];  
echo "<br />";  
}
```

使用 `list()` 函数，可以用来将一个数组分解为一系列的值。可以按照如下方式将函数 `each()` 返回的两个值分开：

```
list($name,$age)=each($ages);
```

当使用 `each()` 函数时，数组将记录当前元素。如果希望在相同的脚本中两次使用该数组，就必须使用函数 `reset()` 将当前元素重新设置到数组开始处。

```
reset($prices);
```

确定唯一的数组元素：`array_unique()`；它会删除掉里面相同值的元素。

置换数组键和值：`array_flip()`；它会对调数组中的 `key` 和 `value`；

三. 数组里的数组

数组不一定就是一个关键字和值的简单列表---数组中的每个位置用来保存另一个数组。使用这种方法，可以创建一个二维数组。可以把二维数组当成一个矩阵，或者是一个具有宽度和高度或者行和列的网格。

二维表

产品名	数量	价格
苹果	6	28.8
猪肉	2	32.1
饼干	3	45.3

```
$products=array(  
    array("苹果",6,28.8),  
    array("猪肉",2,32.1),  
    array("饼干",3,45.3)  
);  
显示这个二维数组：  
echo "|".$products[0][0]."|".$products[0][1]."|".$products[0][2]."|<br />";  
echo "|".$products[1][0]."|".$products[1][1]."|".$products[1][2]."|<br />";  
echo "|".$products[2][0]."|".$products[2][1]."|".$products[2][2]."|<br />";
```

此外，还可以使用双重 `for` 循环来实现同样的效果：

```
for ($row=0;$row<3;$row++) {  
    for ($column=0;$column<3;$column++) {
```

```
        echo "|". $products[$row][$column];  
    }  
    echo "<br />";  
}
```

使用列明的二维数组:

```
$products=array(  
    array("产品名"=>"苹果","数量"=>6,"价格"=>28.8),  
    array("产品名"=>"猪肉","数量"=>2,"价格"=>32.1),  
    array("产品名"=>"饼干","数量"=>3,"价格"=>45.3)  
);
```

显示这个二维数组:

第一种方式:

```
for ($row=0;$row<3;$row++) {  
    echo "|". $products[$row]["产品名"]. "|".  
        $products[$row]["数量"]. "|". $products[$row]["价格"]. "<br />";  
}
```

第二种方式:

```
for ($row=0;$row<3;$row++) {  
    while (!!list($key, $value)=each($products[$row])) {  
        echo "|". $value;  
    }  
    echo "<br />";  
}
```

四. 数组的排序

对保存在数组中的相关数据进行排序是一件非常有意义的事情。使用并且排序一个一维数组是非常简单的。

使用 sort()函数将数组按字母升序进行排序。

```
$products=array("orange","banner","apple");  
sort($products);
```

使用 sort()函数将数字升序进行排序。

```
$prices=array(100,10,4,23,78);  
sort($prices);
```

sort()函数的第二个参数是可选的。这个可选参数可以传递 SORT_REGULAR (默认值)、SORT_NUMERIC 或 SORT_STRING。指定排序类型的功能是非常有用的。比如, 当要比较可能包含有数字 2 和 12 的字符串时, 从数字角度看, 2 要小于 12, 但是作为字符串, "12" 却要小于 "2"。

使用 `asort()` 函数和 `ksort()` 函数对相关数组排序

创建一个包括 3 个产品及价格的数组，然后将它们按价格的升序进行排序：

```
$prices=array("c"=>苹果,"a"=>猪肉,"b"=>饼干);
```

```
asort($prices);
```

按说明排序使用 `ksort()` 函数

```
ksort($prices)
```

反向排序：`sort()`、`asort()` 和 `ksort()` 都是正向排序，当然也有相对应的反向排序。

实现反向：`rsort()`、`arsort()` 和 `krsort()`。

在一些应用程序中，可能希望按另一种方式对数组排序。函数 `shuffle()` 将数组个元素进行随机排序。函数 `array_reverse()` 给出一个原来数组的反向排序。

```
shuffle($pictures);
```

```
array_reverse($pictures);
```

`array_unshift()` 函数将新元素添加到数组头，`array_push()` 函数将每个新元素添加到数组的末尾。

`array_shift()` 删除数组头第一个元素，与其相反的函数是 `array_pop()`，删除并返回数组末尾的一个元素。

`array_rand()` 返回数组中的一个或多个键。

五. 数组的指针操作

在数组中浏览：`each()`、`current()`、`reset()`、`end()`、`next()`、`pos()`、`prev()`;

调用 `next()` 或 `each()` 将使指针前移一个元素。调用 `each($array_name)` 会在指针前移一个位置之前返回当前元素。`next()` 函数则有些不同----调用 `next($array_name)` 是将指针前移，然后再返回新的当前元素。

要反向遍历一个数组，可以使用 `end()` 和 `prev()` 函数。`prev()` 函数和 `next()` 函数相反。它是将当前指针往回移一个位置然后再返回新的当前元素。

六. 统计数组个数

`count()` 和 `sizeof()` 统计数组下标的个数

`array_count_values()` 统计数组内下标值的个数

七. 将数组转换成标量变量: `extract()`

对于一个非数字索引数组，而该数组又有许多关键字-值对，可以使用函数 `extract()` 将它们转换成一系列的标量变量。`extract()` 函数原型如下：

```
extract(array var_array,[int extract_type],[string prefix]);
```

函数 `extract()` 的作用是通过一个数组创建一系列的标量变量，这些变量的名称必须是数组中关键字的名称，而变量值则是数组中的值。

```
$array=array("key1"=>"value1","key2"=>"value2","key3"=>"value3");
```

```
extract($array);
```

```
echo $key1.$key2.$key3;
```

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

第6章 目录与文件

学习要点:

1. 目录操作
2. 磁盘、目录和文件计算
3. 文件处理

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>

将相关的数据组织为文件和目录等实体,这一直是计算环境的核心概念。出于这个原因,程序员需要有一种方法来获得关于文件和目录的重要细节,如位置、大小、最后修改时间、最后访问时间和其他确定信息。

一. 目录操作

解析目录路径: `basename()`函数返回路径的文件名部分。

```
<?
$path = 'C:\AppServ\www\Basic6\Demo1.php';
echo 'path:'.basename($path);
?>
```

获取路径的目录: `dirname()`函数返回路径目录部分。

```
<?
$path = 'C:\AppServ\www\Basic6\Demo1.php';
echo 'path:'.dirname($path);
?>
```

关于路径的信息: `pathinfo()`函数创建一个关联数组,其中包括:目录名、基本名和扩展名。

```
<?
$path = 'C:\AppServ\www\Basic6\Demo1.php';
$array_path = pathinfo($path);
print_r($array_path);
?>
```

确定绝对路径: `realpath()`函数将 `path` 中的所有符号链接和相对路径引用转换为相应的硬链接和绝对路径。

```
<?
$path = './index.php';
echo realpath($path);
```


二. 磁盘、目录和文件计算

确定文件的大小：filesize()函数返回指定文件字节大小。

<?

```
$file = 'C:\AppServ\www\Basic6\Demo1.php';  
echo round(filesize($file)/1024,2).'KB';
```

?>

计算磁盘的可用空间：disk_free_space()函数返回指定的目录所在磁盘分区的可用空间。

<?

```
$drive = 'C:';  
echo round(disk_free_space($drive)/1024/1024,2).'MB';
```

?>

计算磁盘的总容量：disk_total_space()函数返回指定的目录所在磁盘分区的总容量。

<?

```
$drive = 'C:';  
echo round(disk_total_space($drive)/1024/1024,2).'MB';
```

?>

确定文件的最后访问时间：fileatime()函数返回文件的最后访问时间，采用的 Unix 时间戳格式。

<?

```
$file = 'C:\AppServ\www\Basic6\Demo1.php';  
echo date("Y-m-d,h:i:s",fileatime($file));
```

?>

确定文件的最后改变时间：filectime()函数返回文件的最后改变时间，采用 Unix 时间戳格式。

<?

```
$file = 'C:\AppServ\www\Basic6\Demo1.php';  
echo date("Y-m-d,h:i:s",filectime($file));
```

?>

确定文件的最后修改时间：filemtime()函数返回文件的最后修改时间，采用 Unix 时间戳格式。

<?

```
$file = 'C:\AppServ\www\Basic5\Demo1.php';  
echo date("Y-m-d,h:i:s",filemtime($file));
```

?>

三. 文件处理

资源 (resource) 这个词常常与可以发起输入或输出流的实体联系起来。标准输入或输出、文件和网络套接字都是资源的例子。因此你会经常看到本节所介绍的很多函数都是在资源处理的上下文中讨论的，而不是文件处理，这本身是因为所有这些函数都能够与前面所述的资源结合使用。但是，由于这些函数与文件结合使用是应用中最常见的。

将数据写入一个文件，有 3 个步骤：

1. 打开这个文件。如果文件不存在，需要先创建它。
2. 将数据写入这个文件。
3. 关闭这个文件

同样，从一个文件中读出数据，也有 3 个步骤：

1. 打开这个文件。如果这个文件不能打开，就应该意识到这一点并且正确地退出。
2. 从文件中读出数据。
3. 关闭这个文件。

打开文件

要在 PHP 中打开一个文件，可以使用 `fopen()` 函数。当打开一个文件的时候，还需要指定如何使用它。也就是，文件模式。

选择文件模式：当打开一个文件的时候，有 3 中选择：

1. 打开文件为了只读、只写或者读和写。
2. 如果要写一个文件，你可能希望覆盖所有已有的文件内容，或者仅仅将新数据追加到文件末尾。如果该文件已经存在，也可以终止程序的执行而不是覆盖该文件。
3. 如果希望在一个区分了二进制方式和纯文本方式的系统上写一个文件，还必须制定采用的方式。

函数 `fopen()` 支持以上 3 中方式的组合。

`$fp=fopen(文件路径, 文件模式, [是否在 include_path 中搜索一个文件],[允许文件名以协议名称开始(如 http://)])`

fopen()函数的文件模式总结

模式	模式名称	意义
r	只读	文件指针置于文件开头
r+	读写	文件指针置于文件开头
w	只写	在写入前，删除文件内容，将指针返回到文件开头。如果文件不存在，则尝试创建。
w+	读写	在读取或写入之前，删除文件内容，将指针返回到文件开头。如果文件不存在，则尝试创建。

a	只写	文件指针置于文件末尾。如果文件不存在，则尝试创建。此模式成为追加（append）。
a+	读写	文件指针置于文件末尾。如果文件不存在，则尝试创建。此过程称为追加到文件。
b	二进制	二进制模式---用于与其他模式进行连接。如果文件系统能够区分二进制文件和文本文件，你可能会使用它。Windows 系统可以区分；而 UNIX 则不区分。推荐一直使用这个选项，以便获得最大程度的可移植性。二进制模式是默认的模式。
t	文本	用于与其他模式的结合。这个模式只是 Windows 系统下一个选项。它不是推荐选项，除非你曾经在代码中使用 b 选项。

如果 fopen() 函数调用失败，函数将返回 false。可以用一种对于用户友好的方式来处理这个错误，可以通过抑制 PHP 的错误信息并且根据自己的方式给出错误信息。

```
@$fp=fopen("file.txt",'ab'); //ab 追加并且二进制方式
```

写文件

在 PHP 中写文件相对比较简单。可以使用 fwrite() 或者 fputs() 函数。我们可以使用如下方式调用 fwrite();

```
fwrite($fp, $outputstring, [int length]); //第三个可选参数为最大字符数
```

这个函数告诉 PHP 将保存在 \$outputstring 中的字符串写入到 \$fp 指向的文件中。

fwrite() 函数的一个新的替换函数是 file_put_contents()。

可以通过 PHP 的内置 strlen() 函数获得字符串的长度：

```
fwrite($fp, $outputstring, strlen($outputstring));
```

当然还有一种不需要资源句柄的写入方法：file_put_contents()。

关闭文件

当文件使用完毕后，应该将其关闭。应该调用 fclose() 函数：

```
fclose($fp);
```

读出文件

```
$fp = fopen("file.txt", "r");
```

fgetc(): 读出一个字符，并将指针移到下一个字符。

fgets(): 读出一行字符，可以指定一行显示的长度。

fgetss(): 从文件指针中读取一行并过滤掉 HTML 标记。

fread(): 读取定量的字符。

fpasssthru(): 输出文件指针处的所有剩余数据。

file(): 将整个文件读入数组中，以行分组。

readfile(): 读入一个文件并写入到输出缓冲。

file_get_contents(): 将整个文件读入一个字符串。

```
while (!feof($fp)) {  
    echo fgetc($fp);  
}
```

判断读完文件函数：feof() ， 返回 true ,!feof() 返回 false;

查看文件是否存在：file_exists();

```
if (file_exists("file.txt")) {  
    //..  
}  
else {  
    //...  
}
```

查看一个文件的大小：filesize();

删除一个文件：unlink();

在文件中定位：rewind()、fseek()和 ftell();

rewind()函数可以将文件指针复位到文件的开始。

ftell()函数可以以字节为单位报告文件指针当前在文件中的位置。

fseek()函数可以将文件指针 fp 从 whence 位置移动 offset 字节。

文件锁定

假设两个客户试图同时订购同一件商品。那么他们同时打开了这个文件，同时的更新，就会出现少一个客户订购的数据。

为了避免这样的问题，可以使用文件锁定的方法。

flock()函数。

flock 的操作值

操作值	意义
LOCK_SH(以前为 1)	读写锁定。这意味着文件可以共享，其他人可以读该文件
LOCK_EX(以前为 2)	写操作锁定。这是互斥的。该文件不能被共享
LOCK_UN(以前为 3)	释放已有的锁定
LOCK_NB(以前为 4)	防止在请求加锁时发生阻塞

```
$fp = fopen("file.txt","ab");  
flock($fp,LOCK_EX);    //锁定  
fwrite($fp, $outsum);  
flock($fp, LOCK_UN);   //释放  
fclose($fp);
```

目录句柄操作

opendir(): 打开路径指定的目录流。

closedir(): 关闭目录流。

readdir(): 返回目录中的各个元素。

```
$dir = opendir('C:\AppServ\www\Basic5');  
while (!$file = readdir($dir)) {  
    echo $file.'<br />';  
}  
closedir($dir);
```

scandir(): 将目录读入数组。

```
print_r(scandir('C:\AppServ\www\Basic5'));
```

rmdir(): 删除指定的目录。

```
rmdir('C:\AppServ\www\Basic5\123');
```

rename(): 重命名文件。

```
rename('Demo1.php','Demo01.php');
```

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

第7章 自定义函数

学习要点：

1. 标准函数
2. 自定义函数
3. 文件包含
4. 魔法常量

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>

一般来讲，冗余的代码都是不好的。一而再，再而三地重写代码不仅浪费时间，从布局结构角度看也显得粗制滥造。与所有优秀的编程语言一样，PHP 采用了很多方法来缓解冗余代码的问题，其中最常见而且最容易实现的方法就是使用函数（function）。

一. 标准函数

标准的 PHP 发行包中有 1000 多个标准函数，这些标准函数都是系统内置的，不需要用户自己创建而可以直接使用。

```
<?
    echo md5('123456');    //MD5函数对字符串进行加密处理
?>
```

二. 自定义函数

PHP 内置函数允许和文件进行交互、使用数据库、创建图形，还可以连接其他的服务器。但是，在实际工作中，有许多时候所需要的东西是语言的创建者无法预见到的。

声明一个函数可以让我们想内置函数那样使用自己的代码。只要简单地调用这个函数并提供给它必须的参数。这就意味着，在整个脚本中，都可以调用和多次重复使用相同的函数。

创建函数

```
<?
    function functionName() {
        echo '这是一个无参无返回自定义函数';
    }
?>
```

调用函数

```
<?
    functionName();
```

函数命名

1. 函数名不能和已有的函数名重名。
2. 函数名只能包含字母、数字和下划线。
3. 函数名不能以数字开头。

包含参数无返回的函数调用

<?

```
function functionArea($radius) {  
    $area = $radius * $radius * pi();  
    echo $area;  
}
```

```
functionArea(10);
```

?>

包含参数有返回的函数调用：使用 `return()` 语句可以向函数调用者返回任意确定值，将程序控制权返回到调用者的作用域。

<?

```
function functionArea($radius) {  
    return $radius * $radius * pi();  
}
```

```
echo functionArea(10);
```

?>

包含默认参数的函数调用：可以为输入参数指定默认值，在没有提供其他值的情况下，就会把这个默认值自动赋给该参数。

<?

```
function functionArea($radius=10) {  
    return $radius * $radius * pi();  
}
```

```
echo functionArea();
```

?>

返回多个值的函数调用：可以通过返回一个数组然后使用 `list()` 函数构造即可。

<?

```
function functionInfo($name,$age,$job) {  
    $userInfo = array($name,$age,$job); //可以用追加的方式比较常用  
    return $userInfo;  
}
```



```
list($name,$age,$job) = functionInfo('吴祁',19,'学生');  
echo $name.'今年'.$age.'岁了，目前还是个'.$job;  
?>
```

包含引用传参的函数调用：引用传递可以在函数内对参数的修改在函数范围外也能反应。

```
<?  
$prices = 50;  
$tax = 0.5;  
function functionPrices(&$prices,$tax) {  
    $prices = $prices + ($prices * $tax);  
    $tax = $tax * 2;  
}  
  
functionPrices($prices,$tax);  
echo $prices;  
echo '<br />';  
echo $tax;  
?>
```

请注意，函数调用将不区分大小写，所以调用 `functionname()`、`FunctionName()` 或 `FUNCTIOINNAME()` 都是有效的，而且都将返回相同的结果。为了方便，这里都用小写。

注意到函数名称和变量名称是不同的，这一点很重要。变量名是区分大小写的，所以 `$Name` 和 `$name` 是两个不同的变量，但 `Name()` 和 `name()` 则是同一个函数。

理解作用域

变量的作用域可以控制变量在哪里是可见并且可用的。不同的编程语言有不同的变量作用域规则。PHP 具有相当简单的规则：

在函数内部声明的变量作用域是从声明它们的那条语句开始到函数末尾。这叫做函数作用域。这些变量成为局部变量。

在函数外部声明的变量作用域是从声明它们的那条语句开始到文件末尾，而不是函数内部。这叫做全局作用域。这些变量成为全局变量。

特殊的超级全局变量在函数内外部都是可见的。

使用 `require()` 和 `include()` 并不影响作用域。如果这两个语句用于函数内部，函数作用域适用。如果它不在函数内部，全局作用域适用。

关键字 “`global`” 可以用来手动指定一个在函数中定义或使用的变量具有全局作用域。

通过调用 `unset($variable_name)` 可以手动删除变量。如果变量被删除，它就不在参数所指定的作用域中了。

全局变量定义 `global`

```
<?  
$a = 5;  
function fna() {  
    global $a;
```

```
$a = 20;
}
fna();
echo $a;
?>
```

可以通过使用超级全局变量\$GLOBAL，可以访问或改变全局作用域中的任何变量。

```
<?
$GLOBALS['a'] = 5;
function fna() {
    $GLOBALS['a'] = 20;
}
fna();
echo $GLOBALS['a'];
?>
```

创建自己的函数库

通常将函数集文件存放在 library 文件夹里，然后通过文件调用即可。文件名约定促成可以取名为 tool.library.php，tool 可以根据情况来设定，后面两个照抄！

三. 文件包含

为了确保重用性和模块性，最普遍的方式是把功能组建隔离为单独的文件，然后在需要时重新组装。PHP 提供了四种在应用程序中包含文件的语句。

include() 语句将在其被调用的位置处判断并包含一个文件。包含一个文件与在该语句所在位置复制该文件的数据具有相同的结果。

```
<?
include 'include.php';
?>
```

include_once() 函数的作用与 include() 相同，不过它会首先验证是否已经包含了该文件。如果包含了该文件，则不再执行 include_once()。

```
<?
include_once 'include.php';
?>
```

require() 在很大程度上与 include() 相同，都是将一个模板文件包含到 require() 调用所在的位置。

```
<?
require('require.php');
?>
```

require_once() 函数的作用与 require() 相同，不过 require_once() 函数确保文件只包含一次。

在遇到 `require_once()` 后，后面再试图包含相同的文件时都将被忽略。

```
<?
require_once('require.php');
?>
```

`require()` 语句和 `include()` 语句几乎是等价的。二者的差异在于，当这两个语句调用失败后，`require()` 将给出一个致命错误，而 `include()` 只是给出一个警告

四. 魔法常量

PHP 实现了一些所谓的魔法常量，他们并不真的是常量，因为这些魔法常量会根据使用的场合改变值。

名称	描述
<code>__FILE__</code>	当前文件名
<code>__LINE__</code>	当前行号
<code>__FUNCTION__</code>	当前函数名
<code>__CLASS__</code>	当前类名
<code>__METHOD__</code>	当前方法名

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

第 8 章 字符串处理

学习要点:

1. 字符串格式化
2. 操作子字符串
3. 字符串比较
4. 查找替换字符串
5. 处理中文字符

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>

在每天的编程工作中, 处理、调整以至最后控制字符串是很重要的一部分, 一般也认为这是所有编程语言的基础。不同与其他语言, PHP 没有那么麻烦地使用数据类型来处理字符串。这样一来, PHP 中的字符串处理就再容易不过了。

一. 字符串格式化

整理字符串的第一步是清理字符串中多余的空格。虽然这一部操作不是必需的, 但如果要将字符串存入一个文件或数据库中, 或者将它和别的字符串进行比较, 这就是非常有用的。

chop() 函数移除字符串后面多余的空白, 包括新行。

ltrim() 函数移除字符串起始处多余空白。

rtrim() 函数移除字符串后面多余的空白, 包括新行, 此函数是 chop() 的别名。

trim() 函数移除字符串两边多余的空白。

```
<?php
    echo trim('          PHP          ');
?>
```

PHP 具有一系列可供使用的函数来重新格式化字符串, 这些函数的工作方式是各不相同的。

nl2br() 函数将字符串作为输入参数, 用 XHTML 中的
 标记代替字符串中的换行符。

```
<?php
    echo nl2br("This is a Teacher!\nThis is a Student!");
?>
```

将特殊字符转换为 HTML 等价形式, 可以使用 htmlentities() 和 htmlspecialchars 函数。如果想要去掉字符串中的 HTML 标记, 可以使用 strip_tags() 函数

```
<?php
    echo htmlentities('<strong>我是吴祁! </strong>'); //转换所有字符
    echo htmlspecialchars('<strong>我是吴祁! </strong>') //转换特殊字符
    echo strip_tags('<strong>我是吴祁! </strong>') //去掉了<strong>
```

对于字符串来说,某些字符肯定是有效的,但是当将数据插入到数据库中的时候可能会引起一些问题,因为数据库会将这些字符解释成控制符。这些有问题的字符就是引号(单引号和双引)、反斜杠(\)和NULL字符。

PHP 提供了两个专门用于转义字符串的函数。在将任何字符串写到数据库之前,应该使用 addslashes()将它们重新格式化,

在调用了 addslashes()后,所有的引号都加了斜杠,而 stripslashes()函数去掉了这些斜杠。

```
<?php
echo addslashes('This is \' Teacher!');
?>
```

可以重新格式化字符串中的字母大小写。

strtoupper()函数将字符串转换为大写

strtolower()函数将字符串转换成小写

ucfirst()函数将第一个字母转换为大写

ucwords()函数将每个单词第一个字母转换为大写

```
<?php
echo strtoupper('yc60.com@gmail.com');
?>
```

填充字符串函数: str_pad()将字符串用指定个数的字符填充字符串。

```
<?php
echo str_pad('Salad',10).'is good.';
?>
```

二. 操作子字符串

通常,我们想查看字符串的各个部分。例如,查看句子中的单词,或者将一个域名或电子邮件地址分割成一个个的组件部分。PHP 提供了几个字符串函数来实现此功能。

使用函数 explode()、implode()和 join()

为了实现这个功能,我们将使用的第一个函数是 explode()。

使用 implode()或 join()函数来实现与函数 explode()相反的效果,这两个函数的效果是一致的。

```
<?php
$email = 'yc60.com@gmail.com';
$email_array = explode('@',$email);
?>
```

使用 strtok()函数

strtok()函数一次只从字符串取出一些片段(称为令牌)。对于一次从字符串中取出一个单词的处理来说, strtok()函数比 explode()函数的效果更好。

```
<?php
    $str = "I,will.be#back";
    $tok = strtok($str,".#");
    while($tok) {
        echo "$tok<br \>";
        $tok = strtok(".#");
    }
?>
```

使用 substr()函数

函数 substr()允许我们访问一个字符串给定起点和终点的子字符串。这个函数并不适用于我们的例子中，但是，当需要得到某个固定格式字符串中的一部分时，它会非常有用。

```
<?php
    echo substr("abcdef", 1, 3);
?>
```

分解字符串：str_split()返回一个数组，其中各数组元素分别是字符串参数中的一个字符串。

```
<?php
    print_r(str_split("This is a Teacher!"));
?>
```

逆置字符串：strrev()可以将一个字符串逆反过来。

```
<?php
    echo strrev("This is a Teacher!");
?>
```

三. 字符串比较

到目前为止，我们已经用过"=="号来比较两个字符串是否相等。使用 PHP 可以进行一些更复杂的比较。这些比较分为两类：部分匹配和其他情况。

字符串的排序：strcmp()、strcasecmp()和 strnatcmp()

该函数需要两个进行比较的参数字符串。如果这两个字符串相等，该函数返回 0，如果按字典顺序 str1 和 str2 后面（大于 str2）就返回一个正数，如果 str1 小于 str2 就返回一个负数。这个函数是区分大小写的。

函数 strcasecmp()除了不区分大小写之外，其他和 strcmp()一样。

函数 strnatcmp()及与之对应的不区分大小写的 strnatcasecmp()函数是在 PHP4 中新增的。这两个函数按“自然排序”比较字符串，所谓自然排序是按人们习惯的顺序进行排序。

```
<?php
    echo strcmp('a','b');
?>
```

使用 `strspn()` 函数返回一个字符串中包含有另一个字符串中字符的第一部分的长度。也就是求两个字符串之间相同的部分。

```
<?php
    echo strspn('gmail','yc60.com@gmail.com');
?>
```

使用 `strlen()` 函数测试字符串的长度

可以使用函数 `strlen()` 来检查字符串的长度。如果传给它一个字符串，这个函数将返回字符串的长度。例如，`strlen("hello")` 将返回 5。

```
<?php
    echo strlen('This is a Teacher!');
?>
```

确定字符串出现的频率：`substr_count()` 返回一个字符串在另一个字符串中出现的次数。

```
<?php
    echo substr_count('yc60.com@gmail.com','c');
?>
```

四. 查找替换字符串

通常，我们需要检查一个更长的字符串中是否含有一个特定的子字符串。这种部分匹配通常比测试字符串的完全等价更有用处。

在字符串中查找字符串：`strstr()`、`strchr()`、`strrchr()` 和 `stristr()`

函数 `strstr()` 是最常见的，它可以用于在一个较长的字符串中查找匹配的字符串或字符。请注意，函数 `strchr()` 和 `strstr()` 完全一样。

```
<?php
    echo strstr('yc60.com@gmail.com','@');
?>
```

函数 `strstr()` 有两个变体。第一个变体是 `stristr()`，它几乎和 `strstr()` 一样，其区别在于不区分字符大小。对于我们的只能表单应用程序来说，这个函数非常有用，因为用户可以输入 "delivery"、"Delivery" 和 "DELIVERY"。

第二个变体是 `strrchr()`，它也几乎和 `strstr()` 一样，只不过是 `strstr()` 的别名。

查找字符串的位置：`strpos()`、`strrpos()`。

函数 `strpos()` 和 `strrpos()` 的操作和 `strstr()` 类似，但它不是返回一个子字符串，而返回子字符串 `needle` 在字符串 `haystack` 中的位置。更有趣的是，现在的 PHP 手册建议使用 `strpos()` 函数代替 `strstr()` 函数来查看一个子字符串在一个字符串中出现的位置，因为前者的运行速度

更快。

```
<?php
    echo strrpos('yc60.com@gmail.com','c');
?>
```

替换字符串: str_replace()、str_ireplace()、substr_replace()

```
<?php
    echo str_replace('@','#','yc60.com@gmail.com');
    echo substr_replace('yc60.com@gmail.com',###,0,5);
?>
```

五. 处理中文字符

对于以上的字符串函数，有些可以用于中文，但有些却不适用中文。所以，PHP 提供了专门的函数来解决这样的问题。

中文字符可以是 gbk,utf8,gb2312

mb_strlen() 对应的函数为 strlen() 求字符串的长度
mb_strstr() 对应的函数为 strstr() 求某字符串到结尾的字符
mb_strpos() 对应的函数为 strpos() 求出字符最先出现处
mb_substr() 对应的函数为 substr() 取出指定的字符串
mb_substr_count() 对应函数为 substr_str() 返回字符串出现的次数

最后扫一遍手册

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

第9章 正则表达式

学习要点:

1. 正则表达式语法 (Perl 风格)
2. 正则表达式中的元素
3. Perl 风格函数

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>

处理字符串时, 有很多较为复杂的字符串用普通的字符串处理函数无法干净地完成。比如说, 可能需要验证一个 Email 地址是否合法, 为此需要查看许多不容易检查的规则。这正是正则表达式的用武之地。正则表达式是功能强大而简明的字符组, 其中可以包含大量的逻辑, 特别值得一提的是正则表达式相当简短。

一. 正则表达式语法 (Perl 风格)

Perl 一直被认为是最伟大的解析语言之一, 它提供了一种全面的正则表达式, 即使是最复杂的字符串模式, 也可以用这种正则表达式语言搜索和替换。PHP 开发人员认识到, 与其重新发明正则表达式, 不如让 PHP 用户直接使用声名赫赫的 Perl 正则表达式语言, 即 Perl 风格的函数。

模式规则: `/php/` 在字符串前后加上两条斜杠即可。

匹配函数: `preg_match()` 函数在字符串中搜索模式, 如果存在则返回 `true`, 否则返回 `false`。

```
<?php  
    preg_match('/php/', 'php');  
?>
```

二. 正则表达式中的元素

正则表达式中包含三种元素分别为: 量词、元字符、修饰符

量词

语法	描述
<code>+</code>	匹配任何至少包含一个前导字符串
<code>*</code>	匹配任何包含零个或多个前导字符串
<code>?</code>	匹配任何包含零个或一个前导字符串
<code>.</code>	匹配任意字符串
<code>{x}</code>	匹配任何包含 x 个前导字符串
<code>{x,y}</code>	匹配任何包含 x 到 y 个前导字符串

{x,}	匹配任何包含至少 x 个前导字符串
\$	匹配字符串的行尾
^	匹配字符串的行首
	匹配字符串的左边或者右边
()	包围一个字符分组或定义个反引用，可以使用\1\2 提取

元字符

语法	描述
[a-z]	匹配任何包含小写字母 a-z 的字符串
[A-Z]	匹配任何包含大写字母 A-Z 的字符串
[0-9]	匹配任何包含数字 0-9 的字符串
[abc]	匹配任何包含小写字母 a、b、c 的字符串
[^abc]	匹配任何不包含小写字母 a、b、c 的字符串
[a-zA-Z0-9_]	匹配任何包含 a-zA-Z0-9 和下划线的字符串
\w	匹配任何包含 a-zA-Z0-9 和下划线的字符串（同上）
\W	匹配任何没有下划线和字母数字的字符串
\d	匹配任何数字字符，和[0-9]相同
\D	匹配任何非数字字符，和[^0-9]相同
\s	匹配任何空白字符
\S	匹配任何非空白字符
\b	匹配是否到达了单词边界
\B	匹配是否没有达到单词边界
\	匹配正则中的特殊字符

修饰符

语法	描述
i	完成不区分大小写的搜索
m	在匹配首内容或者尾内容时候采用多行识别匹配
x	忽略正则中的空白
A	强制从头开始匹配
U	禁止贪婪匹配 只跟踪到最近的一个匹配符并结束

三. Perl 风格函数

PHP 为使用 Perl 兼容的正则表达式搜索字符串提供了 7 个函数，包括：preg_grep()、preg_match()、preg_match_all()、preg_quote()、preg_replace()、preg_replace_callback() 和 preg_split()。

搜索字符串：preg_grep() 函数搜索数组中的所有元素，返回由与某个模式匹配的所有元素组成的数组。

```
<?php
    $language = array('php','asp','jsp','python','ruby');
    print_r(preg_grep('/p$/', $language));
?>
```

搜索模式：preg_match() 函数在字符串中搜索模式，如果存在则返回 true，否则返回 false。

```
<?php
    echo preg_match('/php[1-6]/', 'php5');
?>
```

电子邮件验证小案例（分组应用）

```
<?php
    $mode = '/([\w\.\_]{2,10})@(\w{1,}).([a-z]{2,4})/';
    $string = 'yc60.com@gmail.com';
    echo preg_match($mode, $string);
?>
```

匹配模式的所有出现：preg_match_all() 函数在字符串中匹配模式的所有出现，然后将所有匹配到的全部放入数组。

```
<?php
    preg_match_all('/php[1-6]/', 'php5sdfphp4sdflljkphp3sdlfjphp2', $out);
    print_r($out);
?>
```

定界特殊的正则表达式：preg_quote() 在每个对于正则表达式语法而言有特殊含义的字符前插入一个反斜线。这些特殊字符包含：\$ ^ * () + = { } [] | \ : < >。

```
<?php
    echo preg_quote('PHP的价格是：$150');
?>
```

替换模式的所有出现：preg_replace() 函数搜索到所有匹配，然后替换成想要的字符串返回出来。

```
<?php
    echo preg_replace('/php[1-6]/', 'python', 'This is a php5, This is a php4');
?>
```

ubb 小案例：贪婪问题+分组使用()

```
<?php
    $mode = '/[b\](.*)[/b\]/U';
    $replace = '<strong>\1</strong>';
    $string = 'This is a [b]php5[/b],This is a [b]php4[/b]';
    echo preg_replace($mode,$replace,$string);
?>
```

以不区分大小写的方式将字符串划分为不同的元素：preg_split()用来分割不同的元素。

```
<?php
    print_r(preg_split('/\./','yc60.com@gmail.com' ));
?>
```

注：目前为 PHP 使用 POSIX 风格的正则表达式搜索字符串提供了 7 个函数，包括：ereg()、ereg_replace()、eregi()、eregi_replace()、split()、spliti() 和 sql_regcase()。

PS：和 Perl 风格基本相同，可以参考手册自行学习。

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

第 10 章 日期与时间

学习要点:

1.PHP 日期和时间库

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>

使用 PHP 编程时, 与你遇到的大多数其他类型的数据相比, 日期和时间有很大不同。因为日期和时间没有明确的结构, 并且日期的计算和表示也很麻烦。在 PHP 中, 日期和时间函数库是 PHP 语言的一个核心部分。

时间戳是自 1970 年 1 月 1 日 (00:00:00 GMT) 以来的秒数。它也被称为 Unix 时间戳 (Unix Timestamp)。Unix 时间戳 (Unix timestamp), 或称 Unix 时间 (Unix time)、POSIX 时间 (POSIX time), 是一种时间表示方式, 定义为从格林威治时间 1970 年 01 月 01 日 00 时 00 分 00 秒起至现在的总秒数。Unix 时间戳不仅被使用在 Unix 系统、类 Unix 系统中, 也在许多其他操作系统中被广泛采用。例如 (1184557366 表示 2007-07-16 03:42:46)

一. PHP 日期和时间库

验证日期: `checkdate()` 函数能够很好地验证日期, 提供的日期如果有效, 则返回 `true`, 否则返回 `false`。

```
<?php
    if (checkdate(2,29,2007)) {
        echo '日期合法';
    } else {
        echo '日期不合法';
    }
?>
```

格式化时间和日期: `date()` 函数返回根据预定义指令格式化时间和日期的字符串形式。所有格式参数, 可以参考手册。

```
<?php
    echo date('Y-m-d H:i:sa'); //直接输入日期和时间
    echo date('今天的日期和时间为: Y/m/d H:i:sa'); //可以插入无关的字符串
?>
```

查看更多时间信息: `gettimeofday()` 函数返回与当前时间有关的元素所组成的一个关联数组。

```
<?php
    print_r(gettimeofday()); //可以传入一个真(1)
?>
```


将时间戳转换成友好的值：getdate()函数接受一个时间戳，并返回一个由其各部分组成的关联数组。如果不给参数，那么返回当前的时间和日期。

```
<?php
    print_r(getdate(1184557366));
?>
```

获取当前的时间戳：time()函数可以获取当前的时间戳，并且可以通过设置时间戳的值。

```
<?php
    echo date('Y-m-d H:i:s',time()+(7*24*60*60));
?>
```

获取特定的时间戳：mktime()函数可以生成给定日期时间的时间戳。

```
<?php
    echo mktime(14,14,14,11,11,2007);
    echo date('Y-m-d H:i:s',mktime(14,14,14,11,11,2007));
?>
```

计算时间差

```
<?php
    $now = time();
    $taxday = mktime(0,0,0,7,17,2010);
    echo round(($taxday - $now)/60/60);
?>
```

将日期转换成时间戳：strtotime()将人可读的日期转换为 Unix 时间戳。

```
<?php
    echo strtotime('2007-10-31 14:31:33');
?>
```

计算时间差

```
<?php
    echo (strtotime('2007-10-31 14:31:33') - strtotime('2007-10-31 11:31:33'))/60/60;
?>
```

获取当前文件最后修改时间：getlastmod()可以得到当前文件最后修改时间的时间戳。

```
<?php
    echo date('Y-m-d H:i:s',getlastmod());
?>
```

设置时区和 GMT/UTC:

修改 php.ini 文件中的设置，找到[date]下的;date.timezone = 选项，将该项修改为 date.timezone=Asia/Shanghai，然后重新启动 apache 服务器。

putenv()函数可以设置当前的默认时区。

```
<?php
    putenv('TZ=Asia/Shanghai');
    echo date('Y-m-d H:i:s');
?>
```

date_default_timezone_set()可以设置当前的默认时区。
date_default_timezone_get()可以获取当前的默认时区。

```
<?php
    date_default_timezone_set('Asia/Shanghai');
    echo date('Y-m-d H:i:s');
?>
```

取得本地时间 localtime()函数可以取得本地时间数据，然后返回一个数组。

```
<?php
    date_default_timezone_set('Asia/Shanghai');
    print_r(localtime());
    print_r(localtime(time(), true));
?>
```

计算页面脚本运行时间：microtime()函数，该函数返回当前 UNIX 时间戳和微秒数。返回格式为 msec sec 的字符串，其中 sec 是当前的 UNIX 时间戳，msec 为微秒数。

```
<?php
function fntime() {
    list($msec, $sec) = explode(' ', microtime());
    return $msec+$sec;
}
$start_time = fntime();
for($i=0;$i<1000000;$i++) {
    //
}
$end_time = fntime();
echo round($end_time - $start_time,4);
?>
```

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

第 11 章 表单与验证

学习要点:

1. Header() 函数
2. 接收及验证数据

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>

我们对 Web 感兴趣, 认为它有用的原因是其主要通过基于 HTML 的表单发布和收集信息的能力。这些表单用来鼓励网站的反馈、进行论坛会话、收集在线订购的邮件地址, 等等。但是对 HTML 表单进行编码只是有效接受用户输入的必须操作的一部分, 必须由服务器端组件来处理输入。

一. Header() 函数

标头 (header) 是服务器以 HTTP 协议传 HTML 资料到浏览器前所送出的字符串, 在标头与 HTML 文件之间尚需空一行分隔。

1. 用于重新导向指定的 URL

```
<?php
    header('Location:http://www.baidu.com');
?>
```

2. 用于设置页面字符编码

```
<?php
    header('Content-Type: text/html; charset=gbk');
    echo '嘿嘿, 我是中文! 页面编码是GBK, 文件也是GBK';
?>
```

注意: 除非启用了输出缓冲, 否则这些命令必须在返回任何输出之前执行。

启用输出缓冲: ob_start()

```
<?php
    ob_start();
?>
```

二. 接受及验证数据

HTML 表单元素

表单元素	描述
text input	文本框

password input	密码框
hidden input	隐藏框
select	下拉列表框
checkbox	复选框
radio	单选按钮
textarea	区域框
file	上传
submit	提交按钮
reset	重置按钮

GET 与 POST

处理表单时，必须指定输入到表单的信息以何种方式传输到其目的地（`method=""`）。对此，Web 开发人员可以采用 GET 和 POST。使用 GET 方法发送数据时，所有域都追加到浏览器的 URL 后面，并且为数据随 URL 地址发送。采用 POST 方法时，值会作为标准值发送。

PHP 分别使用 `$_GET` 和 `$_POST` 超全局变量来处理 GET 和 POST 变量。通过使用这两个超全局变量，可以准确地指定信息应当来自哪里，并以你希望的方式处理数据。

使用 `$_GET` 或 `$_POST` 来接收数据

1. `$_GET['username']`，发送的表单 method 必须是 get;
2. `$_POST['username']`，发送的表单 method 必须是 post;
3. 采用 `isset()` 来验证 `$_GET['username']` 超级全局变量是否定义;
4. 使用 `htmlspecialchars()` 函数将 HTML 特殊字符进行过滤。

对数据有效性进行验证

1. 使用函数 `trim()` 去除数据的前后空格;
2. 使用函数 `strlen()` 判断数据的长度;
3. 使用函数 `is_numeric()` 判断数据是纯数字;
4. 使用正则表达式验证邮箱是否合法。

```
<?php
    if (!isset($_POST['send']) || $_POST['send']!='提交') {
        header("Location:Demo1.php");
        exit;
    }

    if (preg_match('/([\\w.]{2,255})@([\\w-]{1,255}).([a-z]{2,4})/',$_POST['email'])) {
        echo '电子邮件合法';
    } else {
        echo '电子邮件不合法';
    }
?>
```

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

第 12 章 会话控制

学习要点:

- 1.Cookie 的应用
- 2.Session 会话处理

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>

HTTP (超文本传输协议) 定义了通过万维网 (WWW) 传输文本、图形、视频和所有其他数据所有的规则。HTTP 是一种无状态的协议, 说明每次请求的处理都与之前或之后的请求无关。虽然这种简化实现对于 HTTP 的普及做出了卓越的贡献, 但对于希望创建复杂的 Web 应用程序的开发人员来说, 这点有点困扰。为了解决这个问题, 出现了一种在客户端机器上存储少量信息 (cookie)。

由于 cookie 大小限制、数量及其他原因, 开发人员又提出了一种解决方案: session 会话处理。

一. Cookie 的应用

设置 cookie: setcookie() 函数可以在客户端生成一个 cookie 文件, 这个文件可以保存到期时间、名称、值等。

创建 cookie

```
<?php
    setcookie('name','Lee',time()+(7*24*60*60));
?>
```

参数 1: cookie 名称

参数 2: cookie 值

参数 3: cookie 过期时间

读取 cookie

```
<?php
    echo $_COOKIE['name'];
?>
```

删除 cookie

```
<?php
    setcookie('name','');
    setcookie('name','Lee',time()-1);
?>
```

使用 Cookie 的限制

- 1、必须在 HTML 文件的内容输出之前设置;

- 2、不同的浏览器对 Cookie 的处理不一致，且有时会出现错误的结果。
- 3、限制是在客户端的。一个浏览器能创建的 Cookie 数量最多为 30 个，并且每个不能超过 4KB，每个 WEB 站点能设置的 Cookie 总数不能超过 20 个。

二. Session 会话处理

在使用 session 会话处理，必须开始 session，使用 session_start() 开始会话。

创建 session 并读取 session

```
<?php
    session_start();
    $_SESSION['name'] = 'Lee';
    echo $_SESSION['name'];
?>
```

判断 session 是否存在

```
<?php
    session_start();
    $_SESSION['name'] = 'Lee';
    if (isset($_SESSION['name'])) {
        echo $_SESSION['name'];
    }
?>
```

删除 session

```
<?php
    session_start();
    $_SESSION['name'] = 'Lee';
    unset($_SESSION['name']);
    echo $_SESSION['name'];
?>
```

销毁所有 session

```
<?php
    session_start();
    $_SESSION['name'] = 'Lee';
    $_SESSION['name2'] = 'Lee';
    session_destroy();
    echo $_SESSION['name'];
    echo $_SESSION['name2'];
?>
```

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

第 13 章 上传文件

学习要点：

- 1.PHP 上传配置
- 2.\$_FILES 数组
- 3.PHP 上传函数

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>

虽然大多数人认为 Web 只包含网页，但 HTTP 协议实际上可以传输任何文件，如 office 文档、PDF、可执行文件、AVI、压缩文件及各种其他文件类型。虽然 FTP 在历史上一直是向服务器上传文件的标准方式，但通过网页上传文件也逐渐流行起来。

一. PHP 上传配置

有一些配置指令可用于精细地调节 PHP 的文件上传功能。这些指令用来确定是否启用 PHP 的文件上传、可允许的最大上传文件大小、可允许的最大脚本内存分配和其他各种重要的资源。

- 1.file_uploads=on|off：确定服务器上的 PHP 脚本是否可以接受文件上传。
- 2.max_execution_time=integer：PHP 脚本在注册一个致命错误之前可以执行的最长时间，以秒为单位。
- 3.memory_limit=integer：设置脚本可以分配到的最大内存，以 MB 为单位。这可以防止失控的脚本独占服务器内存。
- 4.upload_max_filesize=integer：设置上传文件最大大小，以 MB 为单位。此指令必须小于 post_max_size。
- 5.upload_tmp_dir=string：设置上传文件在处理之前必须存放在服务器的临时一个位置，直到文件移动到最终目的地为止。
- 6.post_max_size=integer：确定通过 POST 方法可以接受的信息的最大大小，以 MB 为单位。

二. \$_FILES 数组

上传表单的 HTML

```
<form enctype="multipart/form-data" action="upload.php" method="post">
  <input type="hidden" name="MAX_FILE_SIZE" value="1000000" />
  上传文件: <input type="file" name="userfile" />
  <input type="submit" value="上传" />
</form>
```

ENCTYPE="multipart/form-data"，这里是固定写法，否则文件上传失败

ACTION="upload.php"，定义要处理上传的程序文件路径

METHOD="post"，定义传输方式为POST，一般情况下Form提交数据都设置为POST

`<input type="hidden" name="MAX_FILE_SIZE" value="1000000" >`，这是一个隐藏域，定义了上传文件的大小上限，超过这个值时，上传失败。它必须定义在文件上传域的前面。而且这里定义的值不能超过在 `php.ini` 文件中 `upload_max_filesize` 设置的值，否则没有意义了。（注意：`MAX_FILE_SIZE` 的值只是对浏览器的一个建议，实际上它可以被简单的绕过。因此不要把对浏览器的限制寄希望于该值。实际上，`PHP.ini`设置中的上传文件最大值，是不会失效的。但是最好还是在表单中加上 `MAX_FILE_SIZE`，因为它可以避免用户在花时间等待上传大文件之后才发现该文件太大了的麻烦。）

`<input type="file" name="userfile" />`，这是文件上传域，`Type`属性必须设置为 `file`，但 `Name`属性可以自定义，这个值会在代码文件中使用。

`$_FILES` 超级全局变量，它储存各种与上传有关的信息，这些信息对于通过 `PHP` 脚本上传到服务器的文件至关重要。

1. 存储在 `$_FILES["userfile"]["tmp_name"]` 变量中的值就是文件在 Web 服务器中临时存储的位置。

2. 存储在 `$_FILES["userfile"]["name"]` 变量中的值就是用户系统中的文件名称。

3. 存储在 `$_FILES["userfile"]["size"]` 变量中的值就是文件的字节大小。

4. 存储在 `$_FILES["userfile"]["type"]` 变量中的值就是文件的 MIME 类型，例如：`text/plain` 或 `image/gif`。

5. 存储在 `$_FILES["userfile"]["error"]` 变量中的值将是任何与文件上载相关的错误代码。这是在 `PHP4.2.0` 中增加的新特性。`error` 分别提供了一些数组常量：`0`:表示没有发生错误，`1`:表示上载文件的大小超出了约定值。文件大小的最大值是 `PHP` 配置文件中指定的，该指令是 `upload_max_filesize`。`2`:表示上载文件大小超出了 `HTML` 表单的 `MAX_FILE_SIZE` 元素所指定的最大值。`3`:表示文件只被部分上载。`4`:表示没有上载任何文件。

```
<?php
    print_r($_FILES);
?>
```

三. PHP 上传函数

`PHP` 的文件系统库中提供了大量文件处理函数，除此之外，`PHP` 还提供了两个专门用于文件上传过程的函数：`is_uploaded_file()` 和 `move_uploaded_file()`。

1. 确定是否上传文件：`is_uploaded_file()`

```
if (is_uploaded_file($_FILES["userfile"]["tmp_name"])) {
    echo '已经上传到临时文件夹';
} else {
    echo '失败';
}
```

2. 移动上传文件：`move_uploaded_file()`

```
if (!move_uploaded_file($_FILES["userfile"]["tmp_name"], $_FILES["userfile"]["name"])) {
    echo '移动失败';
    exit;
}
```

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](http://www.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

第 14 章 处理图像

学习要点:

1. 创建图像
2. 简单小案例

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>

在 PHP5 中, 动态图像的处理要比以前容易得多。PHP5 在 php.ini 文件中包含了 GD 扩展包, 只需去掉 GD 扩展包的相应注释就可以正常使用了。PHP5 包含的 GD 库正是升级的 GD2 库, 其中包含支持真彩图像处理的一些有用的 JPG 功能。

一般生成的图形, 通过 PHP 的文档格式存放, 但可以通过 HTML 的图片插入方式 SRC 来直接获取动态图形。比如, 验证码、水印、微缩图等。

一. 创建图像

创建图像的一般流程:

- 1). 设定标头, 告诉浏览器你要生成的 MIME 类型。
- 2). 创建一个图像区域, 以后的操作都将基于此图像区域。
- 3). 在空白图像区域绘制填充背景。
- 4). 在背景上绘制图形轮廓输入文本。
- 5). 输出最终图形。
- 6). 清除所有资源。
- 7). 其他页面调用图像。

设定标头指定 MIME 输出类型

```
<?php
    header('Content-Type: image/png');
?>
```

创建一个空白的图像区域

```
<?php
    $im = imagecreatetruecolor(200,200);
?>
```

在空白图像区域绘制填充背景

```
<?php
    $blue = imagecolorallocate($im,0,102,255);
    imagefill($im,0,0,$blue);
?>
```

在背景上绘制图形轮廓输入文本

```
<?php
    $white = imagecolorallocate($im,255,255,255);
    imageline($im,0,0,200,200,$white);
    imageline($im,200,0,0,200,$white);
    imagestring($im, 5, 80, 20, "Mr.Lee", $white);
?>
```

输出最终图形

```
<?php
    imagepng($im);
?>
```

清除所有资源

```
<?php
    imagedestroy($im);
?>
```

其他页面调用创建的图形

```

```

二. 简单小案例

简单验证码

```
<?php
    header('Content-type: image/png');
    for($Tmpa=0;$Tmpa<4;$Tmpa++){
        $nmsg.=dechex(rand(0,15));
    }
    $im = imagecreatetruecolor(75,25);
    $blue = imagecolorallocate($im,0,102,255);
    $white = imagecolorallocate($im,255,255,255);
    imagefill($im,0,0,$blue);
    imagestring($im,5,20,4,$nmsg,$white);
    imagepng($im);
    imagedestroy($im);
?>
```

加载已有的图像

```
<?php
    header('Content-Type:image/png');
    define('__DIR__',dirname(__FILE__).'\\');
    $im = imagecreatefrompng(__DIR__.'222.png');
    $white = imagecolorallocate($im,255,255,255);
```

```
imagestring($im,3,5,5,'http://www.yc60.com',$white);  
imagepng($im);  
imagedestroy($im);
```

?>

加载已有的系统字体

<?php

```
$text = iconv("gbk","utf-8","李炎恢");  
$font = 'C:\WINDOWS\Fonts\SIMHEI.TTF';  
imaggottext($im,20,0,30,30,$white,$font,$text);
```

?>

图像微缩

<?php

```
header('Content-type: image/png');  
define('__DIR__',dirname(__FILE__).'\\');  
list($width, $height) = getimagesize(__DIR__.'222.png');  
$new_width = $width * 0.7;  
$new_height = $height * 0.7;  
$im2 = imagecreatetruecolor($new_width, $new_height);  
$im = imagecreatefrompng(__DIR__.'222.png');  
imagecopyresampled($im2, $im, 0, 0, 0, 0,  
                    $new_width, $new_height, $width, $height);  
  
imagepng($im2);  
imagedestroy($im);  
imagedestroy($im2);
```

?>

PS: 扫一遍图像函数手册

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

第 15 章 MySQL 数据库

学习要点:

- 1.Web 数据库概述
- 2.MySQL 的操作
- 3.MySQL 常用函数
- 4.SQL 语句详解
- 5.phpMyadmin

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>

一. Web 数据库概述

现在, 我们已经熟悉了 PHP 的基础知识, 这是我们想暂时离开 PHP 一章, 来重点介绍一下关系型数据库, 让大家了解数据库比文件储存的有点。这些优点包括:

- 1.关系型数据库比普通文件的数据访问速度更快。
- 2.关系型数据库更容易查阅并提取满足特定条件的数据。
- 3.关系型数据库更具有专门的内置机制处理并发访问, 作为程序员, 不需要为此担心。
- 4.关系型数据库可以提供对数据的随即访问。
- 5.关系型数据库具有内置的权限系统。

关系数据库的概念

至今为止, 关系数据库是最常用的数据库类型。在关系代数方面, 它们具有很好的理论基础。当使用关系数据库的时候, 并不需要了解关系理论 (这是一件好事), 但是还是需要理解一些关于数据库的基本概念。

1) 表格

关系数据库由关系组成, 这些关系通常称为表格。顾名思义, 一个关系就是一个数据的表格。电子数据表就是一种表格。

编号	姓名	地址	电话
1	周杰伦	台湾高雄	0323839233
2	陈道明	大陆上海	0212324534
3	李炎恢	大陆盐城	0152343434

2) 列

表中的每一列都有惟一的名称, 包含不同的数据。此外, 每一列都有一个相关的数据类型。

3) 行

表中的每一行代表一个客户。每一行具有相同的格式, 因而也具有相同的属性。行也成为记录。

4) 值

每一行由对应每一列的单个值组成。每个值必须与该列定义的数据类型相同。

5) 键

每一条数据所对应的唯一的标识。

6) 模式

数据库整套表格的完整设计成为数据库的模式。

7) 关系

外键标识两个表格数据的关系。

如何设计 Web 数据库

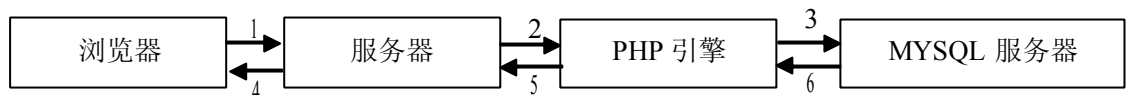
- 1) 考虑要建模的实际对象。
- 2) 避免保存冗余数据。
- 3) 使用原子列值（对每一行的每个属性只存储一个数据。）
- 4) 选择有意义的键。
- 5) 考虑需要询问数据库的问题。
- 6) 避免多个空属性的设计

Web 数据库架构

浏览器和 Web 服务器之间的通信：



浏览器和 PHP&MySQL 服务器之间的通信



- 1) 用户的 Web 浏览器发出 HTTP 请求，请求特定 Web 页面。
- 2) Web 服务器收到.php 的请求获取该文件，并将它传到 PHP 引擎，要求它处理。
- 3) PHP 引擎开始解析脚本。脚本中有一条连接数据库的命令，还有执行一个查询的命令。PHP 打开通向 MySQL 数据库的连接，发送适当的查询。
- 4) MySQL 服务器接收数据库查询并处理。将结果返回到 PHP 引擎。
- 5) PHP 以你干完完成脚本运行，通常，这包括将查询结果格式化 HTML 格式。然后再输出 HTML 返回到 Web 服务器。
- 6) Web 服务器将 HTML 发送到浏览器。

二. MySQL 操作

登录到 MySQL

- 1) 打开 MySQL Command Line Client
- 2) 输入 root 的设置密码

MySQL 常规命令

1)显示当前数据库的版本号和日期。

```
SELECT VERSION(),CURRENT_DATE();
```

2)通过 AS 关键字设置字段名。

```
SELECT VERSION() AS version; //可设置中文，通过单引号
```

3)通过 SELECT 执行返回计算结果

```
SELECT (20+5)*4;
```

4)通过多行实现数据库的使用者和日期

```
>SELECT
```

```
>USER()
```

```
> ,
```

```
>NOW()
```

```
>;
```

5)通过一行显示数据库使用者和日期

```
>SELECT USER();SELECT NOW();
```

6)命令的取消

```
>\c
```

7)MySQL 窗口的退出

```
>exit;
```

MySQL 常用数据类型

整数型: TINYINT, SMALLINT, INT, BIGINT

浮点型: FLOAT, DOUBLE, DECIMAL(M,D)

字符型: CHAR, VARCHAR

日期型: DATETIME, DATE, TIMESTAMP

备注型: TINYTEXT, TEXT, LONGTEXT

日期型

列类型	“零”值
DATETIME	'0000-00-00 00:00:00'
DATE	'0000-00-00'
TIMESTAMP	0000000000000000
TIME	'00:00:00'
YEAR	0000

字符串型

值	CHAR(4)	存储需求	VARCHAR(4)	存储需求
"	' '	4个字节	"	1个字节
'ab'	'ab '	4个字节	'ab '	3个字节
'abcd'	'abcd'	4个字节	'abcd'	5个字节
'abcdefgh'	'abcd'	4个字节	'abcd'	5个字节

整数型

类型	字节	最小值 (带符号的/无符号的)	最大值 (带符号的/无符号的)
TINYINT	1	-128	127
		0	255
SMALLINT	2	-32768	32767
		0	65535
MEDIUMINT	3	-8388608	8388607
		0	16777215
INT	4	-2147483648	2147483647
		0	4294967295
BIGINT	8	-9223372036854775808	9223372036854775807
		0	18446744073709551615

整数型

类型	字节	最小值	最大值
FLOAT	4	+ -1.175494351E-38	+ -3.402823466E+38
DOUBLE	8	+ - 2.2250738585072014E-308	+ - 1.7976931348623157E+308
DECIMAL	可变	它的取值范围可变。	

备注型

类型	描述
TINYTEXT	字符串，最大长度255个字符
TEXT	字符串，最大长度65535个字符
MEDIUMTEXT	字符串，最大长度16777215个字符
LONGTEXT	字符串，最大长度4294967295个字符

MySQL 数据库操作

- 1)显示当前存在的数据库
>SHOW DATABASES;
- 2)选择你所需要的数据库
>USE guest;
- 3)查看当前所选择的数据库
>SELECT DATABASE();
- 4)查看一张表的所有内容
>SELECT * FROM guest; //可以先通过 SHOW TABLES;来查看有多少张表
- 5)根据数据库设置中文编码
>SET NAMES gbk; //set names utf8;
- 6)创建一个数据库

>CREATE DATABASE book;

7)在数据库里创建一张表

>CREATE TABLE users (

>username VARCHAR(20), //NOT NULL 设置不允许为空

>sex CHAR(1),

>birth DATETIME);

8)显示表的结构

>DESCRIBE users;

9)给表插入一条数据

>INSERT INTO users (username,sex,birth) VALUES ('Lee','x',NOW());

10)筛选指定的数据

> SELECT * FROM users WHERE username = 'Lee';

11)修改指定的数据

>UPDATE users SET sex = '男' WHERE username='Lee';

12)删除指定的数据

> DELETE FROM users WHERE username='Lee';

13)按指定的数据排序

> SELECT * FROM users ORDER BY birth DESC; //正序

14)删除指定的表

>DROP TABLE users;

15)删除指定的数据库

>DROP DATABASE book;

三. MySQL 常用函数

文本函数

函数	用法	描述
CONCAT()	CONCAT(x,y,...)	创建形如 xy 的新字符串
LENGTH()	LENGTH(column)	返回列中储存的值的长度
LEFT()	LEFT(column,x)	从列的值中返回最左边的 x 个字符
RIGHT()	RIGHT(column,x)	从列的值中返回最右边的 x 个字符
TRIM()	TRIM(column)	从存储的值删除开头和结尾的空格
UPPER()	UPPER(column)	把存储的字符串全部大写
LOWER()	LOWER(column)	把存储的字符串全部小写
SUBSTRING()	SUBSTRING(column, start, length)	从 column 中返回开始 start 的 length 个字符（索引从 0 开始）
MD5()	MD5(column)	把存储的字符串用 MD5 加密
SHA()	SHA(column)	把存储的字符串用 SHA 加密

数字函数

函数	用法	描述
ABS()	ABS(x)	返回 x 的绝对值
CEILING()	CEILING(x)	返回 x 的值的最大整数
FLOOR()	FLOOR(x)	返回 x 的整数
ROUND()	ROUND(x)	返回 x 的四舍五入整数
MOD()	MOD(x)	返回 x 的余数
RNAD()	RNAD()	返回 0-1.0 之间随机数
FORMAT()	FORMAT(x,y)	返回一个格式化后的小数
SIGN()	SIGN(x)	返回一个值，正数(+1)，0，负数(-1)
SQRT()	SQRT(x)	返回 x 的平方根

日期和时间函数

函数	用法	描述
HOUR()	HOUR(column)	只返回储存日期的小时值
MINUTE()	MINUTE(column)	只返回储存日期的分钟值
SECOND()	SECOND(column)	只返回储存日期的秒值
DAYNAME()	DAYNAME(column)	返回日期值中天的名称
DAYOFMONTH()	DAYOFMONTH(column)	返回日期值中当月第几天
MONTHNAME()	MONTHNAME(column)	返回日期值中月份的名称
MONTH()	MONTH(column)	返回日期值中月份的数字值
YEAR()	YEAR(column)	返回日期值中年份的数字值
CURDATE()	CURDATE()	返回当前日期
CURTIME()	CURTIME()	返回当前时间
NOW()	NOW()	返回当前时间和日期

格式化日期和时间(DATE_FORMAT()和 TIME_FORMAT())

名词	用法	示例
%e	一月中的某天	1~31
%d	一月中的某天，两位	01~31
%D	带后缀的天	1st~31st
%W	周日名称	Sunday~Saturday

%a	简写的周日名称	Sun-Sat
%c	月份编号	1~12
%m	月份编号, 两位	01~12
%M	月份名称	January~December
%b	简写的月份名称	Jan~Dec
%Y	年份	2002
%y	年份, 两位	02
%l	小时	1~12
%h	小时,两位	01~12
%k	小时, 24 小时制	0~23
%H	小时, 24 小制度, 两位	00~23
%i	分钟	00~59
%S	秒	00~59
%r	时间	8:17:02 PM
%T	时间, 24 小时制	20:17:02 PM
%p	上午或下午	AM 或 PM

四. SQL 语句详解

1. 创建一个班级数据库 school, 里面包含一张班级表 grade, 包含编号(id)、姓名(name)、邮件(email)、评分(point)、注册日期(regdate)。

```
mysql> CREATE DATABASE school;    //创建一个数据库
mysql> CREATE TABLE grade (
    //UNSIGNED 表示无符号, TINYINT(2) 无符号整数 0-99, NOT NULL 表示不能为空,
    //AUTO_INCREMENT 表示从 1 开始没增加一个字段, 累计一位
    -> id TINYINT(2) UNSIGNED NOT NULL AUTO_INCREMENT,
    -> name VARCHAR(20) NOT NULL,
    -> email VARCHAR(40),
    -> point TINYINT(3) UNSIGNED NOT NULL,
    -> regdate DATETIME NOT NULL,
    -> PRIMARY KEY (id)    //表示 id 为主键, 让 id 值唯一, 不得重复。
    -> );
```

2. 给这个班级表 grade 新增 5-10 条学员记录

```
mysql> INSERT INTO grade (name,email,point,regdate) VALUES
    ('Lee','yc60.com@gmail.com',95,NOW());
```

3. 查看班级所有字段的记录, 查看班级 id,name,email 的记录

```
mysql> SELECT * FROM grade;
```



```
mysql> SELECT id,name,email FROM grade;
```

WHERE 表达式的常用运算符

MYSQL 运算符	含义
=	等于
<	小于
>	大于
<=	小于或等于
>=	大于或等于
!=	不等于
IS NOT NULL	具有一个值
IS NULL	没有值
BETWEEN	在范围内
NOT BETWEEN	不在范围内
IN	指定的范围
OR	两个条件语句之一为真
AND	两个条件语句都为真
NOT	条件语句不为真

4. 姓名等于'Lee'的学员，成绩大于 90 分的学员，邮件不为空的成员，70-90 之间的成员

```
mysql> SELECT * FROM grade WHERE name='Lee';
mysql> SELECT * FROM grade WHERE point>90;
mysql> SELECT * FROM grade WHERE email IS NOT NULL;
mysql> SELECT * FROM grade WHERE point BETWEEN 70 AND 90;
mysql> SELECT * FROM grade WHERE point IN (95,82,78);
```

5. 查找邮件使用 163 的学员，不包含 yc60.com 字符串的学员

```
mysql> SELECT * FROM grade WHERE email LIKE '%163.com';
mysql> SELECT * FROM grade WHERE email NOT LIKE '%yc60.com%';
```

6. 按照学员注册日期的倒序排序，按照分数的正序排序

```
mysql> SELECT * FROM grade ORDER BY regdate DESC;
mysql> SELECT * FROM grade ORDER BY point ASC;
```

7. 只显示前三条学员的数据，从第 3 条数据开始显示 2 条

```
mysql> SELECT * FROM grade LIMIT 3;
mysql> SELECT * FROM grade LIMIT 2,2;
```

8. 修改姓名为'Lee'的电子邮件

```
mysql> UPDATE grade SET email='yc60.com@163.com' WHERE name='Lee';
```

9. 删除编号为 4 的学员数据

```
mysql> DELETE FROM grade WHERE id=4;
```

MYSQL 分组函数

函数	用法	描述
AVG()	AVG(column)	返回列的平均值
COUNT()	COUNT(column)	统计行数
MAX()	MAX(column)	求列中的最大值
MIN()	MIN(column)	求列中的最小值
SUM()	SUM(column)	求列中的和

10. 过一遍以上的分组函数

略。

11. 检查这个表的信息

```
mysql> SHOW TABLE STATUS \G;
```

12. 优化一张表

```
mysql> OPTIMIZE TABLE grade;
```

五. PhpMyAdmin

phpMyAdmin(简称 PMA)是一个用 PHP 编写的, 可以通过互联网在线控制和操作 MySQL。他是众多 MySQL 管理员和网站管理员的首选数据库维护工具, 通过 phpMyAdmin 可以完全对 MySQL 数据库进行操作。

创建数据库 scholl

创建一个数据库->选择 utf8 字符集

导出另一个数据库 SQL

1. 选择另一个数据库->导出

2. 选择需要导出的表->全选

3. 选择 Add DROP TABLE / DROP VIEW (基本表一旦删除, 表中的数据以及相应建立的索引和视图都将自动被删除)

4. 选择另存为文件

5. 选择执行, 保存 sql 文件


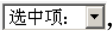
导入数据库

1. 选择被导入的数据库

2. 选择 Import(导入), 选择 sql 文件

3.执行即可

删除表

- 1.可以直接选择操作中的 , 然后确认即可删除数据表.
- 2.也可以选择复选按钮 ☐, 然后选择选中项: , 选择删除, 执行即可

重建表

- 1.找到 sql 文件中的刚才输出的建表语句.
- 2.复制建表语句
- 3.然后选择 sql, 选择粘贴, 执行即可

修复数据表

- 1.选择要修复的表
- 2.在选中项中, 选择修复表,即可

优化数据表

- 1.选择要优化的表
- 2.在选中项中, 选择优化表, 即可

修改, 删除, 插入表记录

执行 SQL 语句

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](http://www.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

第 16 章 PHP 操作 MySQL

学习要点:

1. PHP 连接到 MySQL
2. 增删改查
3. 其他常用函数

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>

如果你已经具有了使用 PHP、SQL 和 MySQL 的丰富经验, 现在就可以把所有这些技术组合在一起。PHP 与 MySQL 之间稳固的集成只是众多程序员采纳它的一个原因, 还有一个原因就是, 它如此的简单方便。

一. PHP 连接到 MySQL

这里, 我们全面采用 UTF-8 编码。

设置 Zend Studio 的编码: Window -> Preferences -> Workspace

标头设置, 让火狐和 IE 保持编码统一:

```
<?php
    header('Content-Type:text/html; charset=utf-8');
?>
```

连接 MySQL

```
<?php
    $conn = @mysql_connect(DB_HOST,DB_USER,DB_PASSWORD) or
        die("数据库连接失败! 错误信息: ".mysql_error());
?>
```

数据库连接参数, 可以用常量存储, 这样就不能修改, 更加安全。

```
<?php
    define('DB_USER','root');
    define('DB_PASSWORD','yangfan');
    define('DB_HOST','localhost');
    define('DB_NAME','school');
?>
```

选择你所需要的数据库

```
<?php
    @mysql_select_db(DB_NAME) or die("数据库找不到! 错误信息: ".mysql_error());
?>
```

设置字符集，如果是 GBK，直接设置 SET NAMES GBK 即可

```
<?php
    @mysql_query('SET NAMES UTF8') or die('字符集设置错误');
?>
```

获取记录集

```
<?php
    $query = "SELECT * FROM grade";
    $result = @mysql_query($query) or die('SQL 语句有误！ 错误信息：'.mysql_error());
?>
```

输出一条记录

```
<?php
    print_r(mysql_fetch_array($result,MYSQL_ASSOC));
?>
```

释放结果集资源

```
<?php
    mysql_free_result($result);
?>
```

关闭数据库

```
<?php
    mysql_close($conn);
?>
```

二. 增删改查

新增数据

```
<?php
    $query = "INSERT INTO grade (name,email,point,regdate) VALUE
              ('李炎恢','yc60.com@gmail.com',,NOW())";
    @mysql_query($query) or die('添加数据出错：'.mysql_error());
?>
```

修改数据

```
<?php
    $query = "UPDATE grade SET name='小可爱' WHERE id=6";
    @mysql_query($query) or die('修改出错：'.mysql_error());
?>
```

删除数据

```
<?php
    $query = "DELETE FROM grade WHERE id=6";
    @mysql_query($query) or die("删除错误: ".mysql_error());
?>
```

显示数据

```
<?php
    $query = "SELECT id,name,email,point FROM grade";
    $result = @mysql_query($query) or die("查询语句出错: ".mysql_error());
    while (!$row = mysql_fetch_array($result)) {
        echo $row['id']. '----'. $row['name']. '----'. $row['email']. '----'. $row['point'];
        echo '<br />';
    }
?>
```

三. 其他常用函数

mysql_fetch_row(): 从结果集中取得一行作为枚举数组

mysql_fetch_assoc(): 从结果集中取得一行作为关联数组

mysql_fetch_array(): 从结果集中取得一行作为关联数组，或数字数组，或二者兼有

mysql_fetch_lengths(): 取得结果集中每个输出的长度

mysql_field_name(): 取得结果中指定字段的字段名

mysql_num_rows(): 取得结果集中行的数目

mysql_num_fields(): 取得结果集中字段的数目

mysql_get_client_info(): 取得 MySQL 客户端信息

mysql_get_host_info(): 取得 MySQL 主机信息

mysql_get_proto_info(): 取得 MySQL 协议信息

mysql_get_server_info(): 取得 MySQL 服务器信息

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

第 17 章 多用户留言系统 1.7-JS 验证(17)

学习要点:

- 1.为什么要 JS 验证
- 2.创建 JS 验证

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>

一. 为什么要 JS 验证

优点:

- 1.减少服务器的压力
- 2.不会刷新
- 3.速度快

缺点:

- 1.占用客户端资源
- 2.被禁用就失效了

二. 创建 JS 验证

```
var fm = document.getElementsByTagName('form')[0];
fm.onsubmit = function () {
    //用户名验证
    if (fm.username.value.length < 2 || fm.username.value.length > 20) {
        alert('用户名不得小于2位或者大于20位');
        fm.username.value = '';
        fm.username.focus();
        return false;
    }
};
```


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

第 17 章 多用户留言系统 2.2-博友界面(22)

学习要点:

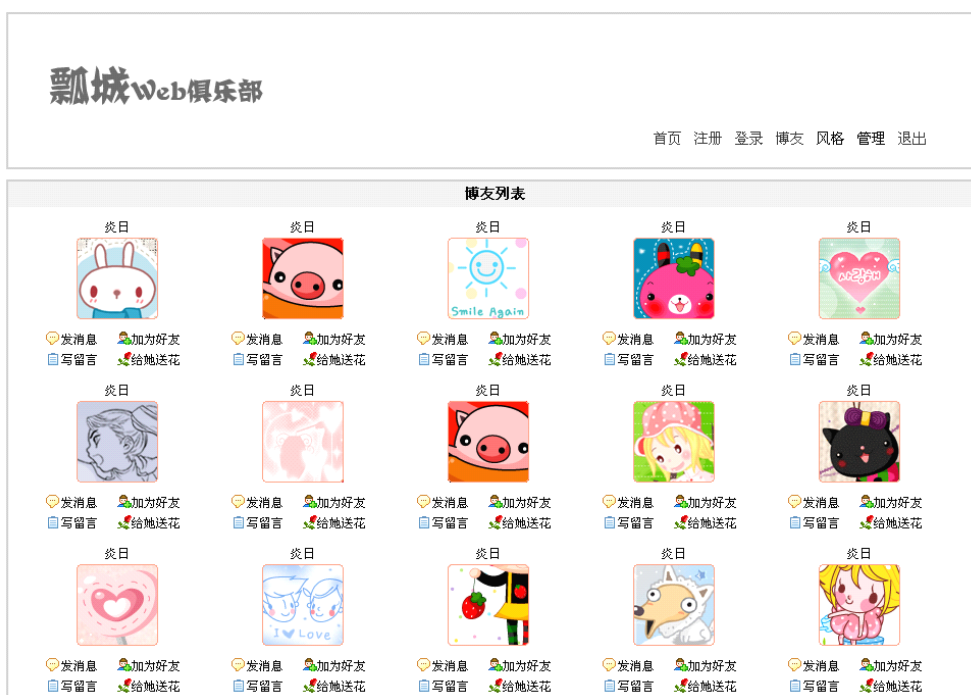
- 1.创建界面
- 2.循环博友

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>

一. 创建界面



二. 循环博友

按照上图的方法循环博友

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

第 17 章 多用户留言系统 1.6-操作包装(16)

学习要点:

1. 新增 mysql.func.php
2. 将功能性的代码写入函数库

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>

一. 新增 mysql.func.php

创建 mysql.func.php 文件, 将数据库操作代码包装到函数中去。

二. 将功能性的代码写入函数库

把常用的功能代码写入函数库, 以后直接调用即可。例如:

```
function _connect() {  
    global $conn;  
    if (!$conn = @mysql_connect(DB_HOST,DB_USER,DB_PWD)) {  
        exit("数据库连接错误!");  
    }  
}
```

上面代码有一个全局变量, 用途是函数里创建, 函数外也能访问。保证了资源句柄的可访问性。

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

第 17 章 多用户留言系统 1.4-创建用户表(14)

学习要点:

- 1.创建数据库
- 2.创建一个用户表

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>

一. 创建数据库

使用 phpmyadmin 创建一个数据库 TestGuest。

二. 创建一个用户表

创建一个表: tg_user;

表所需的字段如下:

dg_id, //自动编号
dg_uniq_id, //验证登录唯一标识符
dg_username, //用户名
dg_password, //密码
dg_question, //问题
dg_answer, //答案
dg_email, //邮件
dg_qq, //QQ
dg_url, //网址
dg_active, //激活账户的唯一标识符
dg_sex, //性别
dg_face, //头像
dg_reg_time, //注册时间
dg_last_time, //最后登录时间
dg_last_ip //最后登录的 IP

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

第 17 章 多用户留言系统 1.9-登录界面(19)

学习要点:

- 1.设计登录界面
- 2.创建必要的 JS

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>

一. 设计登录界面

瓢城Web俱乐部

首页 注册 登录 个人中心 风格 管理 退出

登录

用户名:

密码:

保 留:

☒ 不保留 ☐ 一天 ☐ 一周 ☐ 一月

验证码:

9 b g a

登录

注册

本程序执行耗时: 0.0025秒

版权所有 翻版必究

本程序由瓢城Web俱乐部提供 源代码可以任意修改或发布 (c) yc60.com

二. 创建必要的 JS 验证

可以从注册页面的 JS 复制过来即可。

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

第 17 章 多用户留言系统 0.2-分离调用(2)

学习要点：

- 1.设置注释模板
- 2.分离头尾
- 3.防止恶意调用
- 4.拒绝 PHP 低版本

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>

一. 设置注释模板

```
/**
 * TestGuest Version1.0
 * =====
 * Copy 2010-2012 yc60
 * Web: http://www.yc60.com
 * =====
 * Author: Lee
 * Date: ${date}
 */
```

将以上的注释信息添加到 Templates 模板指定的新建文件里即可。

```
<link rel="shortcut icon" href="favicon.ico"/>
```

调用 ico 文件

二. 分离头尾

将头部和尾部的 HTML 分离出来，分别定义为 header.inc.php 和 footer.inc.php。这样做的好处是，降低了代码量，降低了重复工作量。

三. 防止恶意调用

为了防止外部网站的恶意调用 inc.php 模块文件，可以添加如下语句。

```
//防止恶意调用
```

之后，再调用的页面添加常量。例如：

```
//定义常量
```

```
define('IN_TG',true);
```

四. 拒绝 PHP 低版本

```
//拒绝低版本
```

```
if (PHP_VERSION < '4.1.0') {  
    exit('Version is to Low!');  
}
```

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

第 17 章 多用户留言系统 2.7-分页函数(27)

学习要点:

1. 包装分页
2. 包装参数

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>

一. 包装分页

将下面的分页包装起来

二. 包装参数

将上面的分页初始参数包装起来

感谢收看本次教程!

本课程是由北风网([ibeifeng.com](http://www.ibeifeng.com))

瓢城 Web 俱乐部([yc60.com](http://www.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

第 17 章 多用户留言系统 1.0-过滤用户名(10)

学习要点:

1. 创建验证函数
2. 各种限制与过滤

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>

一. 创建验证函数

在 includes 创建 register.func.php 文件, 用来过滤和返回表单中的各个数据。

二. 各种限制与过滤

首先, 必须去掉两边的空格

```
$_string = trim($_string);
```

其次长度限制

```
if (mb_strlen($_string,'utf-8') < $_min_num || mb_strlen($_string,'utf-8') > $_max_num)
```

敏感字符限制

```
$_char_pattern = '/[<>\'\" \ ]/';
```

```
if (preg_match($_char_pattern,$_string)) {}
```

敏感用户名限制

```
$_mg[0] = '李炎恢';
```

```
$_mg[1] = '胡心鹏';
```

```
$_mg[2] = '吴祁';
```

```
foreach ($_mg as $value) {
```

```
    $_mg_string .= '['.$value.'].'\n';
```

```
}
```

```
if (in_array($_string,$_mg)) {
```

```
    _alert_back($_mg_string.'以上敏感用户名不得注册! ');
```

```
}
```

转义输入, 有效防止 SQL 注入问题

```
//mysql_escape_string();
```

```
//addslashes()
```

```
return mysql_real_escape_string($_string); //这个 mysql_是需要连接数据库的
```

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

第 17 章 多用户留言系统 1.8-激活账户(18)

学习要点:

1. 创建 active 页面
2. 模拟邮件点击超链接

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>

一. 创建 active 页面

设计 active 页面, 这个页面是模式电子邮件的信件页面。里面包含网站发送的激活超级链接。

二. 模拟邮件点击超链接

通过点击发送的超级链接, 将指定的账户的 active 设置为空, 那么这个账户就被激活了。

感谢收看本次教程!

本课程是由北风网([ibeifeng.com](http://www.ibeifeng.com))

瓢城 Web 俱乐部([yc60.com](http://www.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

第 17 章 多用户留言系统 0.1-界面布局(1)

学习要点：

- 1.源项目预览
- 2.设计设计图
- 3.配置新建 XHTML 文件

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>

从今天开始，我们就开始走进项目课堂。

一. 源项目预览



二. 设计图稿



三. 配置新建 XHTML 文件

通过 Window -> Preferences -> PHP -> Editor -> Templates
修改指定的 XHTML 配置新增文件项

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

第 17 章 多用户留言系统 2.5-容错处理(25)

学习要点:

1. Page 传值容错
2. 数据库清零容错

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>

一. Page 传值容错

```
if (isset($_GET['page'])) //判断是否存在 page, 不存在就=1
```

```
empty($_page) || $_page < 0 || !is_numeric($_page) //page 是否为空, 是否小于 0,  
是否不是数字。//如果其中有一个是, 那么就=1
```

```
如果是数字, 但是小数, 那么就$_page = intval($_page);转换成整数
```

```
if ($_page > $_pageabsolute) 页码大于总页码, 就等于总页码
```

二. 数据库清零

如果数据库一条数据都没有, 会报错。

那么 `if($_num == 0)` 页码至少=1

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

第 17 章 多用户留言系统 2.1-生成 cookie(21)

学习要点:

- 1.生成 cookie
- 2.退出

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>

一. 生成 cookie

数据验证成功后, 要生成 cookie 完成登录状态。

```
setcookie('username',$_username,time()+86400);
```

```
setcookie('uniqid',$_uniqid,time()+86400);
```

```
break;
```

并且在 header.inc.php 文件读入 cookie 值

二. 退出

退出登录, 可以直接删除 cookie 和必要的 session

```
setcookie('username','',time()-1);
```

```
setcookie('uniqid','',time()-1);
```

```
_session_destroy();
```

```
_location(null,'index.php');
```

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

第 17 章 多用户留言系统 2.4-数字分页(24)

学习要点:

- 1.LIMIT 用法
- 2.各种参数
- 3.超链接调用

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>

一. LIMIT 用法

LIMIT 0,10 //表示从第 0+1 条开始显示 10 条数据

二. 各种参数

`$_pagesize = 10;` //每页多少条

`$_pagenum = ($_page-1)*$_pagesize;` //从第几条开始

`$_pageabsolute=ceil($_num/$_pagesize);` //页码

三. 超链接调用

```
for ($i=0;$i<$_pageabsolute;$i++) {  
    if ($_page == $i+1) {  
        echo '<li><a  
            href="blog.php?page='.$(i+1)."  
            class="selected">'.(i+1).'/>';  
    } else {  
        echo '<li><a href="blog.php?page='.$(i+1)."'>'.(i+1).'/>';  
    }  
}
```

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

第 17 章 多用户留言系统 0.9-提交数据(9)

学习要点:

- 1.提交数据
- 2.校对验证码
- 3.接受数据

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>

一. 提交数据

将数据提交到本页, 必须做一个名值对, 用来判断是否提交了数据。

?action=register

也可以设计一个隐藏字段来做名值对

```
<input type="hidden" name="action" value="register" />
```

二. 核对验证码

验证码的用途是为了防止恶意注册和一些表单伪造跨站攻击等。

验证码里保存在 session 里面, 可以判断是否是当前表单提交, 而刷新一次后验证码随即变化, 又可以防止多次恶意注册。

```
if (!($_POST['yzm'] == $_SESSION['code'])) {  
    _alert_back('验证码有误, 请重新输入! ');  
}
```

三. 接受数据

设计变量, 将数据提交出来赋值给变量

```
$_username = $_POST['username'];
```

也可以通过一个数组来存放提交过来的值

```
$_clean = array();
```

```
$_clean['username'] = $_POST['username']
```


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

第 17 章 多用户留言系统 0.6-头像取值(6)

学习要点:

- 1.普通方法取值
- 2.分离取值

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>

上一节一个小问题: 屏蔽非 HTML 页面加载 title.inc.php

一. 普通方法取值

通过 onclick 事件处理, 将当前图片的值传输给父页面。

```
function _opener(img) {  
    var faceimg = window.opener.document.getElementById('faceimg');  
    faceimg.src = img;  
}
```

二. 分离方法值

为了保证 HTML 页面的整齐干净, 通过加载节点取值的方式进行操作。

```
window.onload = function () {  
    var img = document.getElementsByTagName('img');  
    for (var i=0;i<img.length;i++) {  
        img[i].onclick = function () {  
            _opener(this.src);  
        };  
    }  
};
```

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

第 17 章 多用户留言系统 2.6-文本分页(26)

学习要点:

1. 创建界面
2. 设计分页

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>

一. Page 传值容错

1/4页 | 共有7篇文章 | [首页](#) | [上一页](#) | [下一页](#) | [尾页](#)

二. 设计分页

```
echo '<li><a href="'.SCRIPT.'.php">首页</a> | </li>';  
echo '<li><a href="'.SCRIPT.'.php?page='.'($_page-1)'. ">上一页</a> | </li>';
```

感谢收看本次教程!

本课程是由北风网(ibelfeng.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

第 17 章 多用户留言系统 2.3-显示博友(23)

学习要点:

1.显示数据

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>

一. 显示数据

这里必须采用 while 循环来数据, 注意, 不能使用之前的封装代码了。会造成死循环, 解决的方法是改装之前的代码。或者再写一个。

```
function _fetch_array_list($_result) {  
    return mysql_fetch_array($_result,MYSQL_ASSOC);  
}
```

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

第 17 章 多用户留言系统 1.5-新增用户(15)

学习要点:

- 1.连接数据库
- 2.判断是否重复
- 3.新增数据

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>

一. 连接数据库

```
<?php
    define('DB_USER','root');
    define('DB_PASSWORD','yangfan');
    define('DB_HOST','localhost');
    define('DB_NAME','testguest');

    $conn = @mysql_connect(DB_HOST,DB_USER,DB_PASSWORD) or
        die('数据库连接失败! 错误信息: '.mysql_error());

    mysql_select_db(DB_NAME) or die('数据库找不到! 错误信息: '.mysql_error());

    mysql_query('SET NAMES UTF8') or die('字符集设置错误');

?>
```

二. 判断用户名是否重复

```
SELECT tg_username FROM tg_user WHERE tg_username='{$_POST['username']}'
```

三. 新增用户

```
"INSERT INTO dg_user (tg_username) VALUES ('{$_POST['username']}');
```

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

第 17 章 多用户留言系统 0.5-选择头像(5)

学习要点:

- 1.将 CSS 做成包含文件
- 2.创建 face.php 页面
- 3.安装 JSEclipse

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>

一. 将 CSS 做成包含文件

每个页面都要导入 CSS, 着实麻烦, 我们定一个文件 title.inc.php。
但是每个页面却有一个不相同的 CSS, 那么可以定义一个常量来证明本页。
`define('SCRIPT','index');`

二. 创建 face.php 页面

这个页面存放着 64 张头像页面, 可以通过数组循环的方式罗列出来

```
<?php foreach (range(1,9) as $number) {?>
<dd></dd>
<?php }?>
```

三. 安装 JSEclipse

Zend Studio 自带的 JS 编辑器不够强大, 可以通过安装 JSEclipse 插件来编写开发。
将所有的 JS 代码分离出来。

特别注意的是: 必须将 JSEclipse 设置为首选, 才能新建生成编辑器。

安装方法搜索一下: 安装 JSEclipse

Help -> Install new software

在线安装地址 <http://download.macromedia.com/pub/labs/jseclipse/autoinstall>

安装好后选择 JSEclipse 首选即可

Window -> Preferences -> Editors -> File Associations

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

第 17 章 多用户留言系统 2.0-验证登录(20)

学习要点:

1.服务器端验证

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>

一. 服务器端验证

JS 验证过后, 再写一遍 PHP 服务器端验证, 加强安全性。

可以从注册那里复制代码过来。

考虑未激活的不能登录。

感谢收看本次教程!

本课程是由北风网([ibeifeng.com](http://www.ibeifeng.com))

瓢城 Web 俱乐部([yc60.com](http://www.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

第 17 章 多用户留言系统 0.7-验证码上(7)

学习要点:

1. 创建验证码函数
2. 注册页面调用

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>

一. 创建验证码函数

验证码函数输入通用函数, 将函数放入 global.func.php 里。

```
//创建一个随机码
for ($i=0;$i<4;$i++) {
    $_nmsg .= dechex(mt_rand(0,15));
}

//将验证码保存到session里
$_SESSION['code'] = $_nmsg;

//设定验证码的图片的长度和高度
$_width = 75;
$_height = 25;

//创建图片
$_img = imagecreatetruecolor($_width,$_height);

//创建一个白色
$_white = imagecolorallocate($_img,255,255,255);

//填充背景
imagefill($_img,0,0,$_white);

//创建一个黑色的边框
$_black = imagecolorallocate($_img,100,100,100);
imagerectangle($_img,0,0,$_width-1,$_height-1,$_black);
```

//随机划线条

```
for ($i=0;$i<6;$i++) {  
    $_rnd_color= imagecolorallocate($_img,mt_rand(0,255),mt_rand(0,255)  
    ,mt_rand(0,255));  
    imageline($_img,mt_rand(0,75),mt_rand(0,75),mt_rand(0,75),mt_rand(0,75)  
    ,$_rnd_color);  
}
```

//随机打雪花

```
for ($i=1;$i<100;$i++) {  
    imagestring($_img,1,mt_rand(1,$_width),mt_rand(1,$_height),"*",  
    imagecolorallocate($_img,mt_rand(200,255),mt_rand(200,255),mt_rand(200,255)));  
}
```

//输出验证码

```
for ($i=0;$i<strlen($_SESSION['code']);$i++){  
    imagestring($_img,mt_rand(3,5),$i*$_width/4+mt_rand(1,10),  
    mt_rand(1,$_height/2),$_SESSION['code'][$i],  
    imagecolorallocate($_img,mt_rand(0,100),mt_rand(0,150),mt_rand(0,200)));  
}
```

//输出销毁

```
header("Content-Type: image/png");  
imagepng($_img);  
imagedestroy($_img);
```

二. 注册页面调用

code.php 已经生成一张图片，所以，直接用图片插入的方式即可

```

```

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

第 17 章 多用户留言系统 0.8-验证码下(8)

学习要点：

1. JS 点击局部刷新
2. 将验证码包装成函数
3. 编写函数注释

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>

一. JS 点击局部刷新

这段代码放入图片区域，可以自动刷新局部

`onclick="javascript:this.src='code.php?tm='+Math.random()"`

然后采用分离 JS 的原则分离出去

二. 将验证码包装成函数

在核心函数库里创建一个 `_code` 函数，将验证码代码放入。

然后设置参数，最大的提供灵活性。

注意：如果没有任何设置，必须有默认值。

三. 编写函数注释

将 `_code()` 函数编写注释，让代码更清晰。

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

第 17 章 多用户留言系统 0.3-执行耗时(3)

学习要点：

- 1.获取运行时间
- 2.创建核心函数库
- 3.底部调用

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>

一. 获取运行时间

通过 `microtime()` 可以获取时间戳和微秒数

```
$_mtime = explode(' ', microtime());
```

```
$_mtime[1] + $_mtime[0];
```

通过睡眠函数来进行测试

```
usleep(2000000) 表示睡眠两秒钟
```

二. 创建核心函数库

创建一个文件，里面存放着各种函数：`global.func.php`。

并且对创建的函数进行必要的注释

三. 底部调用

通过运算好的执行耗时，在底部进行取值显示。

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

第 17 章 多用户留言系统 0.4-注册界面(4)

学习要点:

1.设计注册界面

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>

一. 设计注册界面

瓢城Web俱乐部

首页 注册 登录 个人中心 风格 管理 退出

会员注册

请认真填写以下信息

用户名:

(*必填, 至少两位)

密码:

(*必填, 至少六位)

确认密码:

(*必填, 同上)

密码提示:

(*必填, 至少两位)

密码回答:

(*必填, 至少两位)

性别:

☒ 男 ☐ 女

电子邮件:

QQ:

主页地址:

http://

验证码:

注册

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

第 17 章 多用户留言系统 1.1-密码处理(11)

学习要点:

- 1.验证密码
- 2.对密码进行加密
- 3.验证密码提示
- 4.对密码回答加密

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>

一. 验证并加密密码

编写相关函数, 直接调用传递过来的密码和确认密码进行验证。并且返回使用 md5 或者 sha1 加密的值。

二. 验证密码提示回答并加密回答

编写相关函数, 直接调用传递过来的密码提示和回答。并且返回使用 md5 或者 sha1 加密的值。

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

第 17 章 多用户留言系统 1.2-可选验证(12)

学习要点:

1. 邮件验证
2. QQ 验证
3. 网址验证

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>

一. 邮件验证

```
preg_match('/^[\\w\\-\\.]+@[\\w\\-\\.]+(\\.\\w+)+$/',$_string)
```

当然, 对于什么都没有填写的, 就直接通过。

二. QQ 验证

```
preg_match('/^[1-9]{1}[0-9]{4,9}$/',$_string)
```

三. 网址验证

```
preg_match('/^https?:\\/\\w(\\w+\\.)?[\\w\\-\\.]+(\\.\\w+)+$/',$_string)
```

感谢收看本次教程!

本课程是由北风网([ibeifeng.com](http://www.ibeifeng.com))

瓢城 Web 俱乐部([yc60.com](http://www.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

第 17 章 多用户留言系统 1.3-唯一标识符(13)

学习要点:

1. 表单自动转义
2. 唯一标识符简介
3. 创建两款唯一标识符

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>

一. 表单自动转义

在测试的时候, 我们发现 PHP5 会将表单自动转义。所以我们需要处理一下。可以通过 PHP.INI 来关闭此功能。或者通过一个函数来判断是否开启了自动转义。然后再进行过滤处理。

```
get_magic_quotes_gpc = 0|1
```

二. 唯一标识符简介

uniqid();

依据当时的毫秒以及指定的前置字符串产生一个独一无二的字符串。参数 prefix 为前置的字符串, 最多可达 114 字符

产生 40 个字符的独一无二字符串

```
sha1(uniqid(rand(),true)
```

三. 创建两款唯一标识符

第一款为了防止恶意注册提交, 存入数据库后, 还是可以验证登录身份。

第二款为了激活注册用户。

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