

《PHP+MySQL开发实战》

图书基本信息

书名：《PHP+MySQL开发实战》

13位ISBN编号：9787517027913

出版时间：2015-1

作者：刘坤,杨正校,刘静,普星,沈啸

页数：232

版权说明：本站所提供下载的PDF图书仅提供预览和简介以及在线试读，请支持正版图书。

更多资源请访问：www.tushu111.com

《PHP+MySQL开发实战》

内容概要

本书从初学者的角度出发，通过浅显易懂的语言，丰富的项目实战，对PHP、MySQL进行小型系统开发应该掌握的各方面技术进行详细阐述。依据PHP+MySQL技术学习过程，设计了6个具体项目，每个项目又由若干的模块和任务构成，循序渐进、由浅入深，让学生在完成任务的过程中学到知识和技能，很好地体现“任务驱动、项目载体”的理念。6个项目分别是PHP环境搭建、PHP编程技术、MySQL编程技术、使用PHP开发设计同学录系统、系统移植以及Linux基本操作，分别从LAMP、WAMP环境搭建到项目设计实施进行具体介绍，涉及实施的关键部分都有实施代码，可以使读者轻松掌握PHP、MySQL开发的精髓，快速掌握PHP编程、开发技术。

本书配有电子教案，读者可以从中国水利水电出版社网站和万水书院免费下载，网址为：<http://www.waterpub.com.cn/softdown/>和<http://www.wsbookshow.com>。

作者简介

随着开源潮流的蓬勃发展，用来开发源代码的PHP已经与J2EE和.NET商业软件形成三足鼎立的局面，目前越来越多的公司和企业采用LAMP技术开发系统。本书可以使读者掌握Linux操作系统的安装使用、Apache服务器安装配置和MySQL数据库安装配置，以及PHP编程技巧，结合项目实战，掌握PHP开发Web框架的搭建和完整的开发过程。

本书打破了传统的学科体系，根据项目化教学需要，以项目 - 模块 - 任务来组织内容，最后以同学录系统综合项目为实践，给出了完整的PHP+MySQL系统开发设计详细过程，将Apache、PHP编程技术、MySQL数据库技术融为一体。教材设计中不强求理论体系的完整性，以够用为度，以实用为标准。整个项目任务循序递进，先分别讲解了LAMP、WAMP各技术的学习，又通过综合项目将各部分贯通起来，在学习过程中着重培养了学生独立思考问题和主动解决问题的能力及团队合作精神。

本书依据PHP+MySQL学习过程，从Linux操作系统安装使用、PHP分别在Windows系统和Linux系统下的搭建、MySQL数据库技术、PHP编程技术、综合实训项目安排组织内容。对每个项目中的各个具体的任务都做了详细地分析及操作讲解。5个项目学习完成后能够掌握LAMP、WAMP环境搭建及系统开发，学生成就感强。让学生通过完成具体任务学习知识和技能，真正做到了做中学、学中做，体现了学生为主体的思想。教材每个项目后都有相关知识讲解以及拓展任务训练，便于同学们课后的复习与训练。

本书讲解详实，通俗易懂，容易上手，对于初次学习LAMP、WAMP技术的同学非常实用，书中的代码在Zend Studio 9.0环境下全部通过测试。本书可作为高职、大中专院校师生和计算机培训班的教材，也可作LAMP、PHP编程技术的专业技术参考书，同时可作为业余爱好者的自学用书。

本书由刘坤主编，参与部分编写工作的还有杨正校、刘静、普星、沈啸等。参与本书编写的人员全部来自长期从事教学一线和系统设计制作工作岗位的教师和工程师，具有丰富的系统设计与开发经验。本书是2014年江苏省中等职业教育衔接课程体系建设立项课题《基于校企共建专业的中高职衔接的研究与实践》的研究成果。

在撰写本书过程中参考了大量书籍和资料，在此对这些书籍和资料的作者表示最诚挚的谢意。

在编写过程中，我们精益求精、全面周到，由于编者水平有限，难免有疏漏和不妥之处，恳请专家、同仁和广大读者批评指正。

编者

2014年10月

书籍目录

第1章 PHP环境搭建

1

1.1 Linux上PHP开发环境搭建

1

任务1 在虚拟机中安装VMware tools

1

任务2 Linux下Apache服务器安装与配置

2

任务3 Linux下MySQL数据库安装与配置

3

任务4 phpMyAdmin软件安装

4

任务5 Linux下PHP的安装配置与测试

6

1.2 Windows上WAMP配置

7

任务1 Windows上安装配置Apache服务器

7

任务2 Windows上安装配置MySQL数据库

10

任务3 Windows上安装配置PHP运行环境

21

任务4 Windows上安装配置phpMyAdmin

22

任务5 Windows上安装WAMP集成环境

23

第2章 PHP编程技术

31

2.1 PHP编程基础

32

任务1 在浏览器中显示“hello world”语句

32

任务2 在浏览器中输出个人简历

33

任务3 编程实现“电话号码簿”

34

任务4 计算正方形的面积

34

2.2 PHP流程控制语句

42

任务1 比较两个数大小

42

任务2 根据考试成绩判断等级

43

任务3 用户选择自己喜欢的颜色

44

任务4 简单星期转换

45

任务5 计算1~100的累加值

46

任务6 制作乘法口诀表

47

2.3 PHP字符串与正则表达式语句

51

任务1 去除字符串首尾空格

51

任务2 合并与分割字符串

52

任务3 截取指定长度字符串

52

任务4 验证身份证号码是否规范

53

任务5 验证邮箱格式是否正确

54

任务6 验证网址、IP地址是否符合要求

55

任务7 验证电话号码、账号是否符合规范

56

2.4 PHP数组定义与访问

59

任务1 数组合并与拆分

59

任务2 数组元素的添加与删除

60

任务3 查找数组元素

60

2.5 PHP文件上传

66

任务1 实现单个文件上传

66

任务2 上传指定文件类型的文件

67

2.6 PHP访问Web页面

71

任务1 用户注册页面设计与制作

71

任务2 简单留言板设计实现

73

第3章 MySQL编程技术

83	
3.1 MySQL数据库操作命令	
84	
任务1 启动测试MySQL数据库	
84	
任务2 MySQL数据库管理	
85	
任务3 数据库导出与导入	
86	
3.2 MySQL数据库操作	
89	
任务1 创建MySQL数据库及数据表	
89	
任务2 创建sp数据库及数据表	
91	
任务3 创建student数据库及数据表	
92	
任务4 创建sp数据库及其操作	
94	
3.3 phpMyAdmin管理MySQL数据库	
106	
任务1 使用phpMyAdmin操作数据库	
106	
任务2 使用phpMyAdmin操作数据表	
107	
任务3 使用phpMyAdmin操作数据	
109	
任务4 使用phpMyAdmin导入导出数据库	
111	
3.4 PHP操作MySQL数据库	
113	
任务1 利用数据库保存留言	
113	
任务2 显示留言	
115	
第4章 使用PHP开发设计同学录系统	
120	
4.1 同学录系统需求分析	
121	
任务1 同学录系统需求分析	
121	
任务2 同学录系统设计	
121	
任务3 同学录系统数据库表的结构设计	
122	
4.2 同学录系统数据库创建	
126	
4.3 同学录系统框架设计	

131	
任务1 Linux中PHP工具软件Zend Studio 的安装使用	131
任务2 系统公共文件编写	131
4.4 用户注册模块	138
任务1 编写用户注册用户名文件 reguser.php	138
任务2 编写用户注册信息文件 reginfo.php	142
任务3 编写用户注册完成文件 regok.php	147
4.5 用户登录模块	149
4.6 用户首页模块	153
任务1 设计并实现用户首页login.php 功能	153
任务2 设计并实现创建班级页面createclass.php	157
任务3 实现login.php中“加入班级” 功能	161
4.7 班级首页模块	165
4.8 用户信息模块	167
任务1 实现修改用户信息页面 userinfo.php	167
任务2 修改用户密码页面userpwd.php 文件实现	170
任务3 我的名片usercard.php文件实现	171
4.9 班级留言功能模块	176
任务1 完成发表留言表单设计制作	176
任务2 将用户发表的留言写入数据库 guestbook表中	178
任务3 分页显示班级留言	

179
4.10 班级读物功能模块
183
任务1 完成classreadings.php页面设计
183
任务2 实现classreadings.php页面功能
184
第5章 系统移植
190
任务1 系统从Linux移植到Windows
190
任务2 系统从Windows移植到Linux
193
第6章 Linux基本操作
196
任务1 在虚拟机中安装Linux操作系统
196
任务2 Linux文件和目录操作命令
197
任务3 Linux下vi编辑器使用
201
任务4 Linux软件包安装
203
参考文献
223

精彩短评

1、PHP就是学自这本书。虽然用的是PHP5.5，但是其他的一些逻辑也是共通的

《PHP+MySQL开发实战》

精彩书评

1、我不明白，明明是今年刚出版的书，为什么在写同学录时会请读者去11年已关闭的网易同学录去参考。另外，现在PHP5.5用的都是mysqli..不是mysql...因此给本书评三星而不是四星。本书的结构很好，让人读的进去，光是这一点，就可以说这是一本好书。本书举了许多实例，以及良好的编码习惯及其他习惯，确确实实的为初学者树立了榜样。

1、《PHP+MySQL开发实战》的笔记-vi编辑器

220页到222页

文本编辑器是所有计算机系统中最常用的一种工具。UNIX下的编辑器有ex、sed和vi等，其中，使用最为广泛的是vi编辑器。vi编辑器有3种工作模式：编辑模式、插入模式和命令模式。

编辑模式：进入vi后首先进入的就是编辑模式，屏幕上会等待用户键入命令，也即输入的字母被解释为编辑命令执行，而不是作为文本写到用户的文件中。如果想从编辑模式切换到命令模式，可按“：”键进入命令模式。

插入模式：在编辑模式下输入命令i、a、o中任意一个都可以进入插入模式。在插入模式下，用户输入的字符被作为文件内容保存，并将在屏幕上显示，要从插入模式切换到编辑模式按Esc键即可。

命令模式：用来编辑、存盘和退出文件的模式。命令执行完后，vi自动回到编辑模式。

(1) 进入vi的命令。

vi filename：打开或新建文件，并将光标置于第一行首。

(2) 移动光标类命令（编辑模式）。

h：光标左移一个字符。

l：光标右移一个字符。

space：光标右移一个字符。

Backspace：光标左移一个字符。

k或Ctrl+p：光标上移一行。

j或Ctrl+n：光标下移一行。

Enter：光标下移一行。

w或W：光标右移一个字至字首。

b或B：光标左移一个字至字首。

e或E：光标右移一个字至自尾。

)：光标移至句尾。

(：光标移至句首。

}：光标移至段落开头。

{：光标移至段落结尾。

nG：光标移至第n行首。

n+：光标下移n行。

n-：光标上移n行。

H：光标移至屏幕顶行。

M：光标移至屏幕中间行。

L：光标移至屏幕最后行。

0：（注意是数字零）光标移至当前行首。

\$：光标移至当前行尾。

(3) 屏幕翻滚类命令。

Ctrl+u：向文件首翻半屏。

Ctrl+d：向文件尾翻半屏。

Ctrl+f：向文件尾翻一屏。

Ctrl+b：向文件首翻一屏。

nz：将第n行滚至屏幕顶部，不指定n时将当前行滚至屏幕顶部。

(4) 插入文本类命令。

i：在光标前。

I：在当前行首。

a：光标后。

A：在当前行尾。

o : 在当前行之下新开一行。
O : 在当前行之上新开一行。
r : 替换当前字符
R : 替换当前字符及其后的字符，直至按Esc键。
s : 从当前光标位置处开始，以输入的文本替代指定数目的字符。
S : 删除指定书目的行，并以所输入文本代替之。
ncw或nCW : 修改指定数目的字。
nCC : 修改指定数目的行。

(5) 删除命令。

ndw或ndW : 删除光标处开始及其后的n-1个字。
do : 删至行首。
d\$: 删至行尾。
nnd : 删除当前行及其后n-1行。
x或X : 删除一个字符，x删除光标后的，而X删除光标前的。
Ctrl+u : 删除输入方式下所输入的文本。

(6) 搜索及替换命令 (命令模式)。

/pattern : 从光标开始出向文件尾搜索pattern。
?pattern : 从光标开始处向文件首搜索pattern。
n : 在同一方向重复上一次搜索命令。
N : 在反方向上重复上一次搜索命令。
:n1,n2s/p1/p2/g : 将第n1至n2行中所有p1均用p2替代。
:g/p1/s//p2/g : 将文件中所有p1均用p2替换。

(7) 最后行方式命令。

:n1,n2 co n3 : 将n1行到n2行之间的内容拷贝到第n3行下。
:n1,n2 m n3 : 将n1行到n2行之间的内容移至到第n3行下。
:n1,n2 d : 将n1行到n2行之间的内容删除。
:wq : 保存当前文件并退出。
:q : 退出vi。
:q! : 不保存文件并退出vi。
:!command : 执行shell命令command。
:n1,n2 w!command : 将文件中n1行至n2行的内容作为command的输入并执行之，若不指定n1, n2, 则表示将整个文件内容作为command的输入。

2、《PHP+MySQL开发实战》的笔记-用文本方式将数据装入一个数据库表

如果一条一条地输入，效率比较低，可以用文本文件的方式将所有记录加入数据库表中。创建一个文本文件mytest.txt，每行包含一个记录，用定位符 (tab) 把值分开，并且以在CREATE TABLE语句中列出的列次序给出，例如：
jerry f 1980-07-07 china
mary f 1978-12-12 usa
tom m 1970-09-02 usa

使用下面命令将文本文件mytest.txt装载到mytable表中：
mysql>LOAD DATA LOCAL INFILE
"/root/mytest.txt" INTO TABLE mytable;再使用如下命令看看是否已将数据输入到数据库表中
: mysql>select * from mytable;

3、《PHP+MySQL开发实战》的笔记-PHP输出语句

1.print()输出字符串或变量的值。例如：print \$words或print "你好"。"."运算符用于将字符串进行

合并。

2.echo()输出字符串或变量的值。

3.print_r()输出数组。

4.sprint()将输出指定到字符串标量。

PHP常量定义

```
define(string constant_name,mixed value,case_sensitive=true)
```

该函数有3个参数：

constant_name：必选参数，常量名称，即标识符；

value：必选参数，常量的值；

case_sensitive：可选参数，指定是否大小写敏感，设定为true表示不敏感。

4、《PHP+MySQL开发实战》的笔记-正则表达式符号含义

(1)^：用来匹配字符串的开始。

(2)\$：用来匹配字符串的结束。

(3)*：用来表示一个字符可以出现零次或者多次。

(4)+：用来表示一个字符可以出现一次或者多次。

(5)?：用来表示一个字符可以出现零次或者一次。

(6)[]：方括号表达式，格式为[字母表]，表示任选一个。

(7){ }：可以限定字符出现的个数。

(8)|：表示或者，相当于or。

(9)\：表示转义字符。

(10)\s：表示匹配空白字符。

(11)\S：表示匹配非空白字符。

任务4：验证身份证号码是否规范

```
&lt;代码开始 lang="php"&gt;
$erg="(^[0-9]{15}$)|([0-9]{18}$)|(^[0-9]{17}[x,X]$)";
if(ereg($erg,$_POST["sn"]))
die("恭喜你，已经输入正确的身份证号码！");
else
echo "您输入的身份证号码不正确！";
&lt;/代码结束&gt;
```

任务5：验证邮箱格式是否正确

```
&lt;代码开始 lang="php"&gt;
if(!ereg("^[a-zA-Z0-9_-]+@[a-zA-Z0-9_-]+,$_POST[mail]))
echo "您的邮箱地址不正确！&lt;br&gt;";
else
echo "您的邮箱地址正确！";
&lt;/代码结束&gt;
```

任务6：验证网址、IP地址是否符合要求if(!ereg("^[a-zA-Z]+://[^\s]",\$_POST[url]))

```
$out="您输入的网址不正确！&lt;br&gt;";
if(!ereg("[0-9]{1,3}.[0-9]{1,3}.[0-9]{1,3}.[0-9]{1,3}",$_POST[ip]))
```

```
$out="您输入的IP地址不正确！&lt;br&gt;";  
if($out)  
echo $out;  
else  
echo "您的信息填写完全符合规范！";
```

任务7：验证电话号码、账号是否符合规范

验证电话号码要求转换为正则表达式为：

“ [0-9]{3}-[0-9]{8}|[0-9]{4}-[0-9]{7} ”

验证账号要求转换为正则表达式为：

“ ^[a-zA-Z][a-zA-Z0-9_]{4,15}\$ ”

5、《PHP+MySQL开发实战》的笔记-iframe标记的使用格式

```
&lt;iframe src="URL" width="x" height="x" scrolling="[OPTION]"="x"&gt;  
&lt;/iframe&gt;
```

src：文件的路径，既可是HTML文件，也可以使文本、ASP等。

width、height：“画中画”区域的宽与高。

scrolling：当src的指定的HTML文件在指定的区域显示不完全时，滚动选项，如果设置为NO，则不出现滚动条；如为Auto，则自动出现滚动条，为了让“画中画”与邻近的内容相融合，常设置为0。

iframe常用例子

(1) 页面内加入iframe。

```
&lt;iframe width=420 height=330 frameborder=0 scrolling=auto src=URL&gt;&lt;/iframe&gt;
```

scrolling表示是否显示页面滚动条，可选的参数为auto、yes、no，如果省略这个参数，则默认为auto。

(2) 超链接指向这个嵌入的网页，只要给这个iframe命名就可以了。方法是<iframe name=**>，例如命名为aa，写入这句HTML语言<iframe width=420 height=330 name=aa frameborder=0

```
src=http://www.cctv.com&gt;&lt;/iframe&gt;，然后，网页上的超链接语句应该写为：&lt;a href=URL
```

```
target=aa&gt;
```

(3) 如果把frameborder设为1，效果就像文本框一样。透明的iframe的用法必须是IE5.5以上版本才支持，需要在transparentBody.htm文件的<body>标签中，先加入了style=“background-color=transparent”，然后通过以下四种iframe的写法设置iframe背景透明效果。

```
&lt;IFRAME ID="Frame1" SRC="transparentBody.htm" allowTransparency="true"&gt;&lt;/IFRAME&gt;
```

```
&lt;IFRAME ID="Frame2" SRC="transparentBody.htm" allowTransparency="true" STYLE="background-color:
```

```
green"&gt;&lt;/IFRAME&gt;
```

```
&lt;IFRAME ID="Frame3" SRC="transparentBody.htm"&gt;&lt;/IFRAME&gt;
```

```
&lt;IFRAME ID="Frame4" SRC="transparentBody.htm" STYLE="background-color:
```

```
green"&gt;&lt;/IFRAME&gt;
```

6、《PHP+MySQL开发实战》的笔记-在程序中操作MySQL数据库常用函数说明

(1) mysql_connect()建立数据库连接。

格式：

```
resource mysql_connect([string hostname[:port][:path/to/socket],[string username],[string password]])
```

示例：

```
$conn=mysql_connect("localhost","username","password") or die("不能连接到Mysql Server");
```

说明：使用该连接必须显示关闭连接。

(2) mysql_pconnect()建立数据库连接。

格式：

```
resource mysql_pconnect([string hostname[:port][:/path/to/socket][,string username][,string password]])
```

示例：

```
$conn=mysql_pconnect("localhost","username","password") or die("不能连接到Mysql Server");
```

说明：使用该连接函数不需要显示的关闭连接，它相当于使用了连接池。

(3) mysql_close()关闭数据库连接。

示例：

```
$conn=mysql_connect("localhost","username","password") or die("不能连接到Mysql Server");
```

```
mysql_select_db("MyDatabase") or die("不能选择这个数据库，或数据库不存在");
```

```
echo "你已经连接到MyDatabase数据库";
```

```
mysql_close();
```

(4) mysql_select_db()选择数据库。

格式：

```
boolean mysql_select_db(string db_name [,resource link_id])
```

示例：

```
$conn=mysql_connect("localhost","username","password") or die("不能连接到Mysql Server");
```

```
mysql_select_db("MyDatabase") or die("不能选择这个数据库，或数据库不存在");
```

(5) mysql_query()查询MySQL。

格式：

```
resource mysql_query(string query,[resource link_id])
```

示例：

```
$linkId=mysql_connect("localhost","username","password") or die("不能连接到Mysql Server");
```

```
mysql_select_db("MyDatabase") or die("不能选择这个数据库，或数据库不存在");
```

```
$query = "select * from MyTable";
```

```
$result = mysql_query($query);
```

```
mysql_close();
```

说明：若SQL查询执行成功，则返回资源标识符，失败时返回false。若执行更新成功，则返回true否则返回false。

(6) mysql_result()获取和显示数据。

格式：

```
mixed mysql_result(resource result_set, int row [, mixed field])
```

示例：

```
$query = "select id,name from MyTable order by name";
```

```
$result = mysql_query($query);
```

```
for($count=0;$count<&lt;=mysql_numrows($result);$count++)
```

```
{
```

```
    $c_id = mysql_result($result, 0, "id");
```

```
    $c_name = mysql_result($result, 0, "name");
```

```
    echo $c_id, $c_name;
```

```
}
```

说明：最简单、也是效率最低的数据获取函数。

(7) mysql_fetch_row()获取和显示数据。

格式：

```
array mysql_fetch_row(resource result_set)
```

示例：

```
$query = "select id,name from MyTable order by name";
```

```
$result = mysql_query($query);
```

```
while(list($id,$name) = mysql_fetch_row($result))
```

```
{
```

```
    echo("Name: $name($id)&lt;br/&gt;");
```

```
}
```

说明：函数从result_set中获取整个数据行，将值放在一个索引数组中。通常会结使list()函数使用。

(8) mysql_fetch_array()获取和显示数据。

格式：

```
array mysql_fetch_array(resource result_set [, int result_type])
```

示例1：

```
$query = "select id,name from MyTable order by name";
$result = mysql_query($query);
while($row = mysql_fetch_array($result, MYSQL_ASSOC)
{
    $id = $row["id"];
    $name = $row["name"];
    echo "Name: $name($id)<br/>";
}
```

```
}
```

示例2：

```
$query = "select id,name from MyTable order by name";
$result = mysql_query($query);
while($row = mysql_fetch_array($result, MYSQL_NUM)
{
    $id = $row[0];
    $name = $row[1];
    echo "Name: $name($id)<br/>";
}
```

```
}
```

result_type的值有：

MYSQL_ASSOC：字段名表示键，字段内容为值。

MYSQL_NUM：数值索引数组，操作与mysql_fetch_row()函数一样。

MYSQL_BOTH：既作为关联数组又作为数值索引数组返回。result_type的默认值。

(9) mysql_num_rows()所选择的记录的个数。

格式：

```
int mysql_num_rows(resource result_set)
```

示例：

```
$query = "select id,name from MyTable where id > 65";
$result = mysql_query($query);
echo "有".mysql_num_rows($result)."条记录的ID大于65";
说明：只在确定select查询所获取的记录数时才有用。
```

7、《PHP+MySQL开发实战》的笔记-PHP函数

函数定义：函数是一个被命名的、独立的代码段，函数执行特定任务，并可以给调用它的程序返回一个值。

函数的优点：提高程序的重用性，提高程序的可维护性，可以提高开发效率，提高软件的可靠性，控制程序的复杂性。

函数的声明：function 函数名()

```
{
```

```
}
```

```
function 函数名(参数1,参数2,参数...)
```

```
{
```

```
}
```

```
function 函数名()  
{  
    函数体 ;  
    返回值 ;  
}  
function 函数名(参数列表...)  
{  
    函数体 ;  
    返回值 ;  
}
```

注意：

(1)函数必须调用才能执行，可以在声明之前调用，也可以在声明之后调用。

(2)函数名命名和变量一样，aaa bbb ccc aaaBbbCcc（第一单词小写，以后每个单词首字母大写）。

(3)函数在声明时不能重名。

(4)可以通过向函数传递参数，改变函数的行为。形参：在声明函数时声明的参数，参数就是变量，多个参数用“，”分开，实参：调用函数时传给形参数值（数据，也可以是变量）。

(5)如果没有返回值则成为过程。

(6)通过使用return语句返回数据。

(7)函数执行到return语句就结束，不要在这个语句后写代码，也可以用return结束函数的执行。

函数名的作用：调用函数，开始执行函数，可以向函数中传递数据，函数名就是返回的值。

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:www.tushu111.com