



开发人员专业技术丛书



Core Web Application Development with PHP and MySQL

PHP 和 MySQL

Web 应用开发核心技术

(美) Marc Wandschneider 著
马朝晖 温继文 等译



附赠

机械工业出版社
China Machine Press

开发人员专业技术丛书

TP312
2033D

Core Web Application Development with PHP and MySQL

PHP 和 MySQL

Web 应用开发核心技术

(美) Marc Wandschneider 著
马朝晖 温继文 等译



机械工业出版社
China Machine Press

本书介绍使用PHP和MySQL构建健壮的Web数据库应用程序的方法，涉及范围从前端设计和计划，直到测试和部署。本书主要内容包括：保护Web应用程序，构建Web服务，验证输入，访问数据，管理会话等技术，处理MySQL以及其他数据库（如Oracle和PostgreSQL等）的技术。书中讲解3个完整的Web应用程序案例，附带光盘提供本书全部代码。

本书适合Web应用程序开发人员参考。

Simplified Chinese edition copyright © 2006 by Pearson Education Asia Limited and China Machine Press.

Original English language title: *Core Web Application Development with PHP and MySQL* (ISBN 0-13-186716-4) by Marc Wandschneider, Copyright © 2006.

All rights reserved.

Published by arrangement with the original publisher, Pearson Education, Inc., publishing as Pearson Education, Inc..

本书封面贴有Pearson Education（培生教育出版集团）激光防伪标签，无标签者不得销售。

版权所有，侵权必究。

本书法律顾问 北京市展达律师事务所

本书版权登记号：图字：01-2006-1945

图书在版编目（CIP）数据

PHP和MySQL Web应用开发核心技术 / (美) 温施耐德 (Wandschneider, M.) 著; 马朝晖等译. -北京: 机械工业出版社, 2006. 9

(开发人员专业技术丛书)

书名原文: Core Web Application Development with PHP and MySQL

ISBN 7-111-19236-2

I. P… II. ①温… ②马… III. ①PHP语言-程序设计 ②关系数据库-数据库管理系统, MySQL IV. TP312②TP311.138

中国版本图书馆CIP数据核字(2006)第059534号

机械工业出版社(北京市西城区百万庄大街22号 邮政编码 100037)

责任编辑: 李东震 王高翔

北京中兴印刷有限公司印刷·新华书店北京发行所发行

2006年9月第1版第1次印刷

186mm × 240mm · 40.25印张

定价: 78.00元(附光盘)

凡购本书, 如有倒页、脱页、缺页, 由本社发行部调换
本社购书热线: (010) 68326294

前 言

几年以前，当我开始决定学习一些关于Web应用程序的知识时，我真希望有本书这样的资料。从编写Windows系统的GUI应用程序到编写动态Web应用程序的过渡过程惊人的复杂。极具开发效率的语言（比如PHP）能够使这一过渡过程简单化，但是学习的过程依然艰难。

带着这种想法，我开始编写本书，解释如何使用编写Web应用程序时所涉及的各种技术。在学习PHP和数据库的时候，我几乎犯了能犯的所有错误，因此，我试图把我学到的所有知识写入到本书中。现在出版了太多书籍仅仅是现有文档或者Unix man页面的翻版，我真正希望的是看到有些不同的出版物。

但是，技术只是成功的一半。不通过策略性和系统化的方式进行设计和实现，应用程序注定是昂贵和杂乱的，这还是最好的情况，最坏的情况就是失败的程序。因此，本书重点介绍你可能在相同领域的其他书籍中看不到的很多内容——包括设计、测试和安全。

面向的读者

本书面向的是需要或者希望编写Web应用程序的读者。你或许是公司的开发人员，希望为公司构建内联网应用程序；或者为客户构建因特网Web应用程序。你或许是顾问，帮助小型企业开发Web商店；或者你只是业余爱好者，希望更多地了解Web应用程序并且编写自己的应用程序。

虽然本书并没有期望读者是个专家，但是我假设读者至少熟悉程序设计。了解基本程序设计结构（比如变量和函数），并且基本了解HTML，阅读本书也就绰绰有余了。

因为开始学习Web应用程序设计时，我几乎不了解数据库，所以我对读者也没有更高的要求。从基本术语和设计，到高级查询，本书详细地讲解了数据库。简单明了的指导和引人入胜的范例相结合，可帮助读者进行开发工作。

关于PHP

PHP是PHP/FI（个人主页/表单解释器，Personal Home Page/Forms Interpreter）的一系列脚本，出现于1994年。一个名叫Rasmus Lerdorf的人为了帮助管理他的Web站点而编写了PHP。这些年以来，PHP变得更加正式。在1997年，这个系统的带有附加功能的第二个版本发布了。

在1998年后期发布的PHP 3.0，代码进行了重要的重新编写，两个新的开发者Zeev Suraski和Andi Gutmans参加了开发工作。它的目的是支持Web上日益广泛而且更加复杂的应用程序。在2000年早期发布了PHP 4.0版本。这个版本基于称为Zend Engine的新语言引擎，具有更好的性能和更强的代码模块性。2004年后期发布了进一步发展的PHP 5.0版本。它包含了很多新的特性，比如新的语言结构、更加广泛的Web服务器支持、会话，以及附加的第三方扩展。新的特性之一是显著改进和扩展的面向对象的程序设计模型，本书广泛地使用了这一特性。随着PHP的逐步发展，PHP指的不再是“个人主页（Personal Home Page）”，而是“PHP超文本预处理程序（PHP

Hypertext Preprocessor)”，这是所谓的递归首字母缩写词。(这个首字母缩写词实际上构成了它定义的术语的一部分！)

PHP具有非常高的开发效率。你可以坐下来并且在短期内制成(对，就是这个技术术语)大量代码。几年前就是这种开发效率吸引了我。通过PHP，我能够使用相当少的代码把我在各个国家新奇和动态的旅行见闻令人惊讶地汇合在一起。

一个可能的“美中不足”之处是，开发人员可能会不自觉地把这种语言和其他程序设计语言进行古怪的比较，特别是，如果你(就像我)具有比较类似于C/C++、Java或者Visual BASIC的语言背景。实际上，本书讲解的很多内容是我不知所措并且迷惑不解了几个小时后才想出的结果。但是，当你熟悉了这种语言之后，它能使程序设计成为快乐，并且具有令人惊讶的强大功能。

本书结构

本书包括几大部分，分别介绍编写Web应用程序的逻辑过程。

第一部分“PHP基础”介绍PHP语言本身，开始是对语言最基础的介绍和讨论。然后介绍函数、代码结构和面向对象的程序设计。最后是对数组和字符串的讨论。

第二部分“数据库基础”讲解数据库，开始是设计和数据定义过程中的术语和处理。最后介绍数据库和表的使用，从最基本的查询到事务处理和函数。

第三部分“规划Web应用程序”，这部分不再讨论编写代码，而是回来讨论Web应用程序的设计。很多开发者过于专注于编写他们的代码，以至于忘了应事先静下心来进行实际地考虑。(我希望)这些章节在你进行设计时能给你一些引导。这一部分还特别着重地讲解了如何保证应用程序的安全，因为这是不可小视的事情。

第四部分“实现Web应用程序”讨论从简单的PHP和数据库过渡到编写Web应用程序的过程中所需的更加高级的语言特性和概念，涵盖错误、会话和XML这些主题——包括大量范例代码。

第五部分“范例工程和更多概念”讲解三个完整的Web应用程序范例，开始是简单的预约管理系统，然后介绍博客系统，最后是电子商务商店。这些范例涉及本书讲解的很多内容，并且被设计为健壮的应用程序。这里没有进行过于细致的讲解并且列出范例的所有代码，而是只介绍最重要的代码段，把下载和运行代码的工作留给你去完成。

“附录”部分讨论运行Web应用程序所必须的软件的安装问题，以及如何在不同的数据库引擎上执行各种数据库操作，并且提出一些额外阅读资料的建议。

准备工作

阅读本书之前，请花点儿时间打开本书附带的光盘，把源代码复制到你的计算机中。phpwasrc/目录包含你需要的初始文件。

除了本书第五部分的三个大型范例的完整源代码(和安装指导)之外，还有本书很多章节附带的小型代码段。你可以运行、调试它们，或者改动和尝试它们，以便进一步帮助你了解语言是如何工作的。

除范例源代码之外，光盘还包含PHP、Apache HTTP服务器和MySQL，你可以使用它们在

本地计算机上安装软件。光盘上包括指导。

致谢

认真来讲，如果没有很多帮助，没有一本书能够完成，本书也不例外。衷心地感谢Prentice Hall和我的编辑Mark Taub，他们的帮助使我这本书的梦想成真。我撰写本书的自由和心情从未受到打扰，他们在回答我发送给他们的问题时显示了无比的热情。

非常感谢Kristine Kuch、Matthew Leingang和Kimberly Jenson组成的技术审阅团队。在过去的六个月中，他们不厌其烦地指出我的愚蠢错误或者不良的书写，他们给了我很多赞许和表扬，也给了我很多教导，并且帮助我成为Web应用程序员和作者。

最后，没有我妻子Samantha的支持，所有这些工作都是不可能完成的，她首先不断地鼓励我完成更多的撰写工作，并且在我发誓再也不看计算机和打字的那些夜晚给了我亲切的鼓励。

结束语

我希望你能够对这本书感到满意。我非常希望在你编写Web应用程序的过程中，本书能够提供帮助，并且至少提供了一些阅读乐趣。我是个非常爱说话的人，并且总是乐于听到其他人的意见；因此，如果你对本书有任何评论，或者建议添加任何内容，或者有任何不同意见，我非常高兴得到你的意见，我的电子邮件地址是marcwan@chipmunkninja.com。

目 录

前言

第一部分 PHP基础

第1章 开始了解PHP	1
1.1 第一个PHP程序	1
1.2 输入PHP脚本	3
1.2.1 标示PHP代码段落	3
1.2.2 混合PHP和HTML	4
1.2.3 语句和注释	5
1.3 如何存储数据	6
1.4 输入PHP的基本类型	7
1.4.1 数字	7
1.4.2 字符串	9
1.4.3 布尔值	12
1.5 一些非常有用的函数	12
1.5.1 nl2br	12
1.5.2 var_dump	13
1.5.3 print_r	13
1.5.4 var_export	14
1.6 小结	14
第2章 PHP语言	15
2.1 输入字符串的更多介绍	15
2.2 数据类型的更多介绍	17
2.2.1 数组	17
2.2.2 对象	18
2.2.3 特殊类型和值	18
2.3 强制类型转换	19
2.3.1 基础	19
2.3.2 特殊强制类型转换	20
2.3.3 有用的强制类型转换函数	23
2.4 变量和常量	24
2.4.1 定义常量	24

2.4.2 按值和按引用的变量	25
2.4.3 变量范围	26
2.4.4 变量生存周期	26
2.4.5 预定义变量	27
2.5 表达式和操作符	27
2.5.1 操作符: 组合表达式	27
2.5.2 组合表达式和操作符的过程	33
2.6 控制结构	34
2.6.1 if语句	34
2.6.2 switch语句	35
2.6.3 while/do...while循环	37
2.6.4 for循环	37
2.6.5 foreach循环	38
2.6.6 中断循环	38
2.7 小结	39
第3章 代码组织和重用	40
3.1 基本代码重用: 函数	40
3.1.1 定义和调用函数	40
3.1.2 把参数传递给函数	42
3.1.3 从函数返回值	47
3.1.4 函数内的变量范围	48
3.1.5 函数范围和可用性	50
3.1.6 把函数作为变量使用	52
3.2 中级代码重用: 使用和包含文件	53
3.2.1 把代码组织到文件中	53
3.2.2 选择文件名和位置	55
3.2.3 在脚本中包含库文件	57
3.2.4 把包含用于页面模板化	61
3.3 小结	62
第4章 面向对象的程序设计	63
4.1 超越库之外	63
4.2 面向对象的程序设计	65
4.2.1 一些基本术语	66

4.2.2 PHP中对象的基础	66	6.2.3 远东地区字符集	110
4.2.3 初始化和清除对象	70	6.2.4 Unicode	111
4.2.4 可见性: 控制看这些内容的人	72	6.2.5 Unicode编码	111
4.2.5 把静态数据添加到类	74	6.3 了解PHP中字符集的处理	111
4.3 扩展对象	78	6.3.1 处理其他字符集	112
4.3.1 扩展已有的类	78	6.3.2 问题的出现	113
4.3.2 再次讲解可见性	78	6.3.3 我们应该如何处理字符	113
4.3.3 基类方法的重新实现	79	6.4 配置PHP处理Unicode	114
4.3.4 使类的行为相同: 多态性	80	6.4.1 安装以及配置mbstring和mbregex	114
4.4 其他特性	87	6.4.2 函数重载	115
4.4.1 对象的比较	87	6.5 对字符串的操作	115
4.4.2 对象的克隆	88	6.5.1 获得信息	115
4.4.3 用户界面友好的输出	89	6.5.2 整理字符串	117
4.4.4 类型提示	90	6.5.3 搜索和比较	118
4.4.5 自动加载	90	6.5.4 提取	121
4.5 小结	91	6.5.5 大小写操作	122
第5章 处理数组	92	6.5.6 字符编码转换	122
5.1 再次介绍数组	92	6.6 小结	123
5.1.1 创建数组和添加数据	92	第7章 与服务器进行交互: 表单	124
5.1.2 访问数组中的元素	95	7.1 一个表单的范例	124
5.1.3 删除数组元素和整个数组	97	7.2 处理HTML表单	126
5.1.4 计算数组中元素的数量	97	7.2.1 把表单添加到页面中	126
5.2 数组元素的迭代	98	7.2.2 如何提交数据	128
5.2.1 foreach循环	98	7.2.3 在脚本中访问表单数据	130
5.2.2 常规循环	99	7.2.4 字符集和表单	132
5.2.3 内部循环计数器以及each、next、 prev、pos和reset	100	7.3 服务器的操作	132
5.2.4 array_walk方法	101	7.3.1 服务器的注意事项	133
5.3 多维数组	102	7.3.2 服务器变量	133
5.4 对数组的操作	103	7.3.3 环境变量	136
5.4.1 数组排序	104	7.4 重新定向用户	137
5.4.2 其他数组操作	106	7.5 小结	139
5.5 小结	108	第二部分 数据库基础	
第6章 世界上的字符串和字符	109	第8章 数据库介绍	141
6.1 字符串和PHP	109	8.1 我们讨论的是什么	141
6.2 字符集和Unicode	109	8.1.1 基础	141
6.2.1 ASCII	109	8.1.2 关系型数据库	142
6.2.2 ISO 8859字符集	110	8.2 使用DBMS的动机	143

8.2.1 与简单文件或者电子表格的比较	143	9.6.6 外键和级联删除	167
8.2.2 数据库服务器	143	9.7 删除表和数据库	169
8.3 主要的数据库服务器	144	9.8 小结	169
8.3.1 MySQL	144	第10章 使用数据库: 存储和检索数据	170
8.3.2 PostgreSQL	145	10.1 准备工作	170
8.3.3 Oracle数据库	145	10.2 把数据插入表	170
8.3.4 Microsoft SQL Server	145	10.2.1 INSERT INTO语句	170
8.3.5 其他服务器	145	10.2.2 批量数据插入	171
8.4 如何选择数据库服务器	145	10.2.3 日期类型条目	173
8.4.1 考虑你的数据	146	10.3 检索表中的数据	174
8.4.2 能力	146	10.3.1 基本语法	175
8.4.3 性能	146	10.3.2 在检索表数据时进行组合	176
8.4.4 是否可以从PHP访问	146	10.3.3 对检索出的数据进行排序	179
8.4.5 费用	146	10.3.4 一次获得少量行	180
8.5 我们的选择	147	10.4 修改表中的数据	180
8.6 通用接口	147	10.5 删除表中的数据	181
8.7 小结	148	10.6 小结	181
第9章 设计和创建数据库	149	第11章 使用数据库: 高级数据访问	183
9.1 把什么存放到数据库中	149	11.1 事务处理	183
9.2 组织数据	150	11.1.1 问题	183
9.2.1 主键	150	11.1.2 解决方案	184
9.2.2 选择数据类型	151	11.1.3 编写事务处理	185
9.2.3 把数据组织到表中	153	11.1.4 一个比较复杂的问题	185
9.2.4 便于更快搜索的索引	155	11.2 更高级的查询	188
9.3 SQL简介	156	11.2.1 组合表达式	188
9.4 创建数据库	157	11.2.2 指定值的集合和范围	188
9.4.1 与数据库服务器对话	157	11.2.3 获得唯一的值	189
9.4.2 连接和身份验证	157	11.2.4 查询中的SQL函数	189
9.4.3 创建数据库	158	11.2.5 对聚合函数数据进行分组	193
9.5 设置用户权限	159	11.3 修改表模式	194
9.5.1 准备创建用户	159	11.4 小结	195
9.5.2 创建用户	161	第12章 PHP和数据访问	196
9.5.3 删除用户或者删除权限	161	12.1 准备工作	196
9.6 创建表	162	12.2 连接和身份验证	197
9.6.1 SQL数据类型	162	12.2.1 事件序列	197
9.6.2 数据库服务器客户的生存周期	164	12.2.2 建立连接	198
9.6.3 创建表	165	12.2.3 设置连接字符集	199
9.6.4 表存储引擎	166	12.3 执行查询	199
9.6.5 创建索引	167		

12.3.1 检索数据	200	14.2.1 级联样式单 (CSS)	235
12.3.2 验证用户输入	202	14.2.2 包含文件	237
12.3.3 插入、删除和更新数据	204	14.2.3 生成用户界面的代码库	238
12.3.4 事务处理	205	14.2.4 实现用户界面驱动	242
12.3.5 来自mysql的错误	206	14.3 小结	243
12.4 预备语句	208	第15章 用户管理	244
12.4.1 绑定参数	208	15.1 用户如何连接到我们的应用程序	244
12.4.2 绑定结果	209	15.2 访问者和已知用户	246
12.5 旧式接口	210	15.3 用户身份验证	247
12.5.1 过程式数据库接口的基础知识	210	15.3.1 部分和完全登录	247
12.5.2 持久性连接	212	15.3.2 在什么位置存储用户信息	247
12.6 小结	213	15.3.3 存储什么和不存储什么	248
第三部分 规划Web应用程序		15.4 小结	249
第13章 Web应用程序和因特网	215	第16章 保护Web应用程序：规划和代码 的安全性	250
13.1 详细讲解World Wide Web	215	16.1 处理安全性问题的策略	250
13.1.1 因特网：并没有你想像的那么复杂	215	16.1.1 最重要的事情	250
13.1.2 计算机和计算机的通信	216	16.1.2 安全性和可用性之间的平衡	251
13.1.3 超文本传输协议	217	16.1.3 开发工作完成之后	251
13.1.4 MIME类型	219	16.1.4 我们的基本方法	251
13.1.5 安全套接字层 (SSL)	220	16.2 标识威胁	251
13.1.6 其他重要协议	222	16.2.1 威胁	252
13.2 设计Web应用程序	222	16.2.2 造成威胁的人	253
13.2.1 术语	222	16.3 保护代码	254
13.2.2 基本布局	222	16.3.1 黄金法则	254
13.2.3 用户界面	224	16.3.2 过滤输入	255
13.2.4 业务逻辑	224	16.3.3 代码的组织	259
13.2.5 后端/服务器	227	16.3.4 代码中的内容	260
13.2.6 n层体系结构	227	16.3.5 文件系统注意事项	260
13.2.7 性能和稳定性	228	16.3.6 代码稳定性和bug	261
13.3 小结	229	16.3.7 执行引用和exec	261
第14章 实现用户界面	230	16.4 小结	262
14.1 用户界面的注意事项	230	第17章 保护Web应用程序：软件和硬件 安全性	263
14.1.1 了解你的用户界面	230	17.1 保护Web服务器和PHP	263
14.1.2 计划用户界面	231	17.1.1 保持软件为最新版本	263
14.1.3 帮助出现问题的用户	233	17.1.2 php.ini	264
14.1.4 设计技巧和策略	234	17.1.3 Web服务器配置	264
14.2 实现用户界面	235		

17.1.4	虚拟服务器	265
17.1.5	商业托管的Web应用程序	266
17.2	安全套接字层 (SSL)	267
17.3	数据库的安全性	268
17.3.1	用户和权限系统	268
17.3.2	把数据发送到服务器	269
17.3.3	连接到服务器	269
17.3.4	运行服务器	269
17.4	保护网络	270
17.4.1	防火墙	270
17.4.2	DMZ	270
17.4.3	了解DoS和DDoS攻击	271
17.5	计算机和操作系统的的天性	271
17.5.1	保持操作系统为最新版本	271
17.5.2	只运行必须的程序	271
17.5.3	服务器的物理安全	271
17.6	灾难计划	272
17.7	小结	272

第四部分 实现Web应用程序

第18章	错误处理和调试	273
18.1	错误是如何出现的	273
18.1.1	来自PHP的错误	273
18.1.2	我们代码中的缺陷	274
18.1.3	外部错误	275
18.2	PHP如何处理错误	276
18.2.1	PHP如何显示错误	276
18.2.2	PHP生成哪些错误	276
18.2.3	处理错误	277
18.2.4	配置PHP的错误处理	282
18.3	异常	282
18.3.1	异常的基础知识	282
18.3.2	处理异常	284
18.3.3	未处理的异常	287
18.3.4	扩展异常	288
18.4	调试	290
18.4.1	测试代码	290
18.4.2	源代码调试器	292

18.5	小结	292
第19章	cookie和会话	293
19.1	cookie: 诱人并且有用	293
19.1.1	基本操作	293
19.1.2	cookie如何工作	296
19.1.3	控制cookie的有效性	297
19.1.4	删除cookie	298
19.1.5	cookie数组	298
19.1.6	把什么放到cookie中	298
19.1.7	能力监视器	298
19.2	会话	299
19.2.1	基本用法	300
19.2.2	配置PHP的会话	301
19.2.3	如何传输会话ID	302
19.2.4	使用会话存储数据	303
19.2.5	页面缓存	306
19.2.6	破坏会话	307
19.2.7	会话存储如何工作	308
19.3	会话的安全性	312
19.3.1	获得会话ID	312
19.3.2	限制泄密的会话ID造成的损害	313
19.4	小结	314
第20章	用户身份验证	315
20.1	成员计划	315
20.2	Web服务器提供的身份验证	317
20.2.1	基本的HTTP身份验证	317
20.2.2	Microsoft Windows身份验证方案	321
20.3	实现自己的身份验证	322
20.3.1	配置数据库来处理登录	322
20.3.2	添加新的用户	323
20.3.3	登录用户	331
20.3.4	更新需要用户登录的页面	337
20.3.5	注销用户	340
20.3.6	删除用户	342
20.4	小结	343
第21章	高级输出和输出缓冲	344
21.1	全球化和地区	344
21.1.1	地区及其属性	344

21.1.2 了解用户所在地区	345	23.1.3 基本术语	373
21.1.3 设置当前页面的地区 (Unix)	346	23.1.4 XML文档的结构	374
21.1.4 设置当前页面的地区 (Windows)	347	23.1.5 命名空间	378
21.1.5 了解当前地区	349	23.1.6 验证XML	380
21.2 格式化输出	349	23.1.7 相关技术	381
21.2.1 数字的格式化	350	23.2 在PHP中处理XML	382
21.2.2 货币值	350	23.2.1 使用SAX还是DOM	383
21.2.3 其他格式化函数	353	23.2.2 使用DOM	383
21.3 输出缓冲	355	23.3 XHTML	392
21.3.1 输出缓冲如何工作	355	23.3.1 为什么使用XHTML	393
21.3.2 使用输出缓冲	356	23.3.2 如何处理XHTML	393
21.3.3 编写自己的处理程序	357	23.3.3 转换为XHTML	395
21.4 小结	358	23.4 小结	395
第22章 使用正则表达式进行数据验证	359	第24章 文件和目录	396
22.1 使用正则表达式	359	24.1 访问文件	396
22.1.1 正则表达式是什么	359	24.1.1 打开文件	396
22.1.2 设置	360	24.1.2 关闭文件	398
22.1.3 测试表达式	360	24.1.3 读取文件	398
22.1.4 基本搜索	361	24.1.4 写入文件	400
22.1.5 字符类	362	24.1.5 文件权限和其他信息	402
22.1.6 边界	363	24.1.6 删除和重命名文件	404
22.1.7 点号	364	24.2 访问目录	404
22.1.8 重复模式	364	24.2.1 路径操作	404
22.1.9 分组和选择	365	24.2.2 使用类浏览目录内容	405
22.1.10 技巧和陷阱	365	24.2.3 使用函数浏览目录内容	406
22.2 使用正则表达式进行数据验证	366	24.2.4 改变当前目录	406
22.2.1 验证用户名	366	24.2.5 创建和删除目录	406
22.2.2 匹配电话号码	366	24.3 安全性注意事项	407
22.2.3 匹配邮政编码	367	24.4 小结	408
22.2.4 匹配电子邮件地址	367	第25章 上传文件	409
22.3 其他正则表达式函数	368	25.1 上传用户文件	409
22.3.1 ereg_replace	368	25.1.1 文件上传如何工作	409
22.3.2 split	370	25.1.2 为上传配置PHP	409
22.4 小结	371	25.1.3 客户表单	410
第23章 XML和XHTML	372	25.1.4 服务器代码	411
23.1 XML	372	25.1.5 限制上传文件的大小	414
23.1.1 XML是什么	372	25.1.6 处理多个文件	415
23.1.2 为什么使用XML	373	25.2 上传文件范例	416

25.2.1 设置	416	27.2.2 WSDL	444
25.2.2 新账号表单	417	27.2.3 HTTP	449
25.2.3 创建新的账户	417	27.2.4 XML-RPC	450
25.2.4 查看用户数据	420	27.3 在PHP中使用Web服务	450
25.2.5 从数据库获得图像	422	27.3.1 选择Web服务	450
25.3 安全性注意事项	423	27.3.2 配置PHP	452
25.3.1 只允许可信用户	423	27.3.3 处理服务	452
25.3.2 拒绝服务	424	27.4 范例: 处理Google API	456
25.3.3 文件验证	424	27.4.1 为使用Google API进行设置	456
25.3.4 恶意文件名	424	27.4.2 更多地了解服务	456
25.4 小结	424	27.4.3 搜索如何工作	457
第26章 处理日期和时间	425	27.4.4 搜索关键字	459
26.1 日期和时间的来源	425	27.5 小结	463
26.1.1 PHP	425	第28章 使用PEAR	464
26.1.2 操作系统	425	28.1 PEAR介绍	464
26.1.3 数据库服务器	426	28.1.1 代码库	464
26.1.4 Web页面和用户	426	28.1.2 PEAR基础类	464
26.2 PHP中的日期和时间	427	28.1.3 支持社区	465
26.2.1 PHP中的时间戳	427	28.1.4 PECL	465
26.2.2 获得日期和时间	428	28.2 安装和配置	465
26.2.3 验证日期和时间	431	28.2.1 Unix用户	465
26.2.4 比较日期和时间	432	28.2.2 Windows用户	465
26.2.5 输出格式化的日期和时间	435	28.3 基本命令	467
26.2.6 时间戳的问题	437	28.3.1 获得帮助	467
26.3 数据库服务器中更多的日期和时间	438	28.3.2 列出包	467
26.3.1 常用数据库服务器中的日期和时间 范围	438	28.3.3 下载和安装包	468
26.3.2 增加或者减去时间间隔	438	28.3.4 获得信息	470
26.3.3 解析数据库中的日期	439	28.3.5 更新现有包	470
26.3.4 MySQL和时间戳	439	28.3.6 卸载包	471
26.4 小结	440	28.3.7 PEAR配置选项	471
第27章 XML Web服务和SOAP	441	28.4 范例: 使用日期类	472
27.1 XML Web服务	441	28.4.1 安装	472
27.1.1 建立阶段	441	28.4.2 基本使用	472
27.1.2 进入XML Web服务	441	28.4.3 更多范例	473
27.1.3 查找Web服务	442	28.5 小结	474
27.2 Web服务如何工作	443	第29章 开发和部署	475
27.2.1 SOAP	443	29.1 编码标准	475
		29.1.1 关于样式的担心	475

29.1.2	开发编码标准文档	476	30.5.3	错误设置	515
29.1.3	争论	479	30.5.4	数据库设置	515
29.1.4	其他注意事项	479	30.6	小结	515
29.2	源代码控制	480	第31章	预约管理器	517
29.2.1	我的目的是什么	480	31.1	概述	517
29.2.2	它们如何工作	480	31.2	安装和运行范例	518
29.2.3	选择源代码控制系统	483	31.3	结构和页面流程	519
29.2.4	处理源代码控制	484	31.3.1	页面结构	519
29.3	测试	485	31.3.2	数据库布局	521
29.3.1	为什么要费力进行测试	485	31.3.3	UI策略	522
29.3.2	单元测试	486	31.3.4	完整文件清单	524
29.3.3	性能和负载测试	489	31.4	查看代码	525
29.3.4	bug控制	491	31.4.1	AppointmentManager类	525
29.4	部署	491	31.4.2	处理日期和时间	529
29.4.1	测试服务器	491	31.4.3	处理表单和页面流程	532
29.4.2	过程的脚本编写和自动化	491	31.4.4	显示星期和月	538
29.4.3	部署到运营服务器	492	31.5	建议/练习	541
29.5	小结	493	31.5.1	改变天视图和周视图	541
第五部分 范例工程和更多概念					
第30章	开发成功Web应用程序的策略	495	31.5.2	星期一到星期日的星期显示	542
30.1	单实例对象	495	31.5.3	删除或者移动预约	542
30.2	会话管理	497	31.5.4	转换为PEAR Date类	542
30.2.1	配置	497	31.5.5	允许重叠预约	542
30.2.2	安全性	498	31.5.6	构成多用户系统	542
30.2.3	将所有东西组合在一起	499	31.6	小结	542
30.3	整体的错误处理方式	501	第32章	博客引擎	543
30.3.1	用户错误与应用程序错误	501	32.1	概述	543
30.3.2	替换默认的处理程序	503	32.2	安装和运行范例	544
30.3.3	向用户显示错误	506	32.3	结构和页面流程	545
30.3.4	创建新的异常类	508	32.3.1	页面布局	545
30.4	数据库连接管理	509	32.3.2	数据库结构和说明	546
30.4.1	更好的途径	510	32.3.3	UI策略	548
30.4.2	最佳途径	512	32.3.4	完整文件清单	549
30.4.3	改进了的新的字符串转义函数	514	32.4	查看代码	550
30.5	PHP配置设置	514	32.4.1	用户界面生成	550
30.5.1	总体设置	514	32.4.2	用户管理	553
30.5.2	多字节字符串设置	515	32.4.3	跟踪登录的用户	559
			32.4.4	管理条目和评论	564
			32.5	建议/练习	570

32.5.1 改进主页上的用户清单	571	33.4.2 实现购物车	586
32.5.2 允许匿名评论	571	33.4.3 结账流程	589
32.5.3 分层的评论	571	33.4.4 提交订单	598
32.5.4 用事务进行用户创建	571	33.4.5 安全性	604
32.5.5 实现新的strip_tags函数	571	33.5 付款处理	604
32.6 小结	572	33.6 建议/练习	605
第33章 电子商务应用程序	573	33.6.1 跳过发货信息	605
33.1 概述	573	33.6.2 发送订单处理	605
33.2 安装和运行范例	574	33.6.3 管理页面	605
33.3 结构和页面流程	575	33.6.4 订单状态/取消订单	605
33.3.1 范例的代码结构	575	33.7 小结	606
33.3.2 页面布局	577		
33.3.3 数据库结构	577		
33.3.4 UI策略	581		
33.3.5 完整文件清单	582		
33.4 查看代码	584		
33.4.1 浏览产品	584		

附 录

附录A 安装/配置	607
附录B 等效的数据库功能	624
附录C 推荐的读物	629

第一部分 PHP基础

第1章 开始了解PHP

现在，你已经使PHP运行在你的服务器或者工作站上了，并且准备好开始使用PHP编写脚本。如果还没有完成以上准备，或者还有问题没有解决，请阅读附录A“安装/配置”，查阅更加详细的信息。在本章中，我们将使用PHP编写第一个程序，并且学习使用这种语言的基础知识。

在本章的课程中，我们将：

- 编写几个基本的PHP程序，并且学习如何输入和处理PHP脚本
- 了解语言是如何构造的以及如何注释代码的基础知识
- 学习PHP中一些比较常用的和基础的数据类型
- 学习将在本书中使用的一些有用的函数

1.1 第一个PHP程序

PHP是一种解释语言（interpreted scripting），就是说它的语言引擎简单地运行你编写的脚本，不需要通过必须的中间步骤把脚本编译或者转换为二进制格式。构成Web应用程序的大部分脚本将存放到和你的HTML内容文件相同的位置。虽然很多人还是使用旧式的扩展名.php3和.shtml，但是一般情况下，包含PHP脚本的文件将使用扩展名.php。把这些文件存放到什么位置，取决于你如何配置Web站点以及Web服务器使用什么扩展名把文件传递给PHP解释器。我们把这个位置称为文档根（document root）。

当Web服务器接收到对已适当命名的文件的请求时，它会把请求传送给PHP引擎，然后PHP引擎将逐行处理文件。对于文件中不包含PHP代码的行（典型情况下，就是包含HTML标记的那些行），服务器不会处理它们，服务器会执行它发现的任何PHP脚本。了解了这些情况之后，我们来编写第一个程序，它混合了PHP和HTML。

```
<html>
<head>
  <title>My First PHP Program</title>
</head>
<body>

<?php
  echo "Hello Everybody!";
?>
</body>
</html>
```

如果我们把这个文件称为firstprogram.php，并且把它存放到我们的文档根目录中，然后就可以输入这个文件的URL（例如http://localhost/gettingstarted/firstprogram.php）。然后我们会看到类似图1-1的情况。

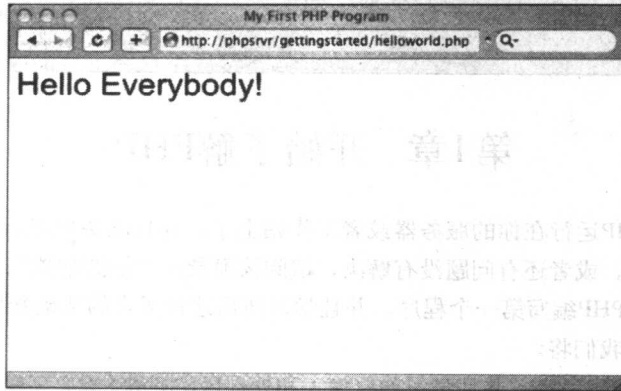


图1-1 运行第一个PHP程序

如果遇到问题，脚本没有正确地运行，应该检查下面几个问题：

- 确保正确地输入了脚本文件的URL，确保服务器名称是合法的。
- 不要忘记路径gettingstarted和Web站点的文档根是相关联的。所以，如果我们Web站点的根目录是c:\inetpub\wwwroot或者/home/samplesite/www，那么前面例子中我们的目录就应该是c:\inetpub\wwwroot\gettingstarted或者/home/httpd/www/gettingstarted。
- 如果看到屏幕上只输出了脚本的代码，可能的原因是Web服务器没有被配置为把.php文件发送给PHP解释器。
- 如果出现错误，声称找不到主机，那么应该确保Web服务器确实在运行。更多的信息参见附录A。

文件的大部分是简单的HTML文本。就像我们说过的，PHP语言引擎输出它发现的任何文本，并且查找PHP代码段落。在前面的文件中，PHP代码是用下面的标记区分出来的：

```
<?php  
...  
?>
```

当PHP语言引擎看到这两个标记时，它会处理脚本并且采取所有适当的操作。到达脚本部分的结尾时，会恢复到HTML的输出。

为了更多地了解PHP，我们将在下一个程序中编写短小的脚本，它使用一个有用的小工具。这个称为phpinfo的函数的使用方法如下：

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