



卫 喆 陈争航 编著

PHP 5与MySQL 5 从入门到精通



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

<http://www.phei.com.cn>

TP312/2923

2008

PHP 5 与 MySQL 5 从入门到精通

卫 喆 陈争航 编著

電子工業出版社

Publishing House of Electronics Industry

北京 • BEIJING

内 容 简 介

本书采用循序渐进的方法介绍了用 PHP 5 与 MySQL 5 进行 Web 程序设计的基础知识,包括服务器环境的构建,PHP 基本语法,常用 PHP 函数库的使用方法,MySQL 5 安装、配置方法,SQL 语法,phpMyAdmin 的配置与使用,PHP 模板 Smarty,PHP 与 Ajax 结合使用,MySQL 视图,存储过程等。在介绍基础知识之后,通过两个完整的 Web 应用开发实例,全面介绍了使用 PHP 5 与 MySQL 5 进行 Web 应用开发的完整过程。

本书内容精简而翔实,针对使用 PHP 5 与 MySQL 5 进行 Web 应用开发必须掌握的基础和核心技术,进行了详细介绍。所选实例实用性强,极具代表性,第一个实例的功能逻辑较为简单,通过这个实例可以使读者掌握用 PHP 5 与 MySQL 5 进行 Web 开发的完整流程。第二个实例的功能逻辑比较复杂,功能模块较多,通过该实例介绍了复杂应用系统开发的过程,并介绍了数据库连接类、分页类等常用类的封装以及文件包含技术的应用,同时,还介绍了 Ajax 技术的应用。通过一易一难两个实例,读者能够迅速掌握用 PHP 5 与 MySQL 5 进行 Web 应用开发的核心实用技术。

本书面向初中级读者,旨在使读者快速掌握用 PHP 5 与 MySQL 5 进行 Web 应用开发所必需的基础与核心技术,适合作为编程爱好者自学和网站开发人员的参考用书。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有,侵权必究。

图书在版编目(CIP)数据

PHP 5 与 MySQL 5 从入门到精通 / 卫喆, 陈争航编著. —北京: 电子工业出版社, 2008.8

ISBN 978-7-121-06915-4

I. P… II. ①卫…②陈… III. ①PHP 语言—程序设计②关系数据库—数据库管理系统, MySQL 5 IV. TP311.138

中国版本图书馆 CIP 数据核字 (2008) 第 087262 号

责任编辑: 吴 源 易 昆

印 刷: 北京天竺颖华印刷厂

装 订: 三河市金马印装有限公司

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编: 100036

北京市海淀区翠微东里甲 2 号 邮编: 100036

开 本: 787×1092 1/16 印张: 34.75 字数: 880 千字

印 次: 2008 年 8 月第 1 次印刷

定 价: 61.00 元

凡所购买电子工业出版社图书有缺损问题, 请向购买书店调换。若书店售缺, 请与本社发行部联系。联系及邮购电话: (010) 88254888。

质量投诉请发邮件至 zltz@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线: (010) 88258888。

前 言

作为网站开发的黄金组合,PHP 与 MySQL 受到越来越多的 Web 应用开发者的欢迎,越来越多的网站在构建时选择采用 PHP 与 MySQL。一直以来,PHP 以简单的语法、丰富的库函数以及极快的脚本解释执行速度,赢得了许多开发者的青睐,而 PHP 5 是 PHP 发展史上的一个里程碑,它功能更强、更加健壮和稳定。

在 PHP 5 之前,PHP 的发展思路是尽可能为快速开发 Web 应用提供方便,所以在 PHP 5 之前的各个版本中大量引入了库函数,而语言模型的发展则相对缓慢。虽然 PHP 4 中加入了面向对象的设计,但其语言模型并不完善,限制了利用 PHP 完成大规模应用程序的能力。而 PHP 5 从根本上改变了 PHP 的上述弊端,引入了完备的对象模型、改进的语法设计,使 PHP 成为了设计完备、真正具有面向对象能力的脚本语言。如今,全世界有上千万的网站和接近数万家公司在用 PHP 语言,包括雅虎、Google、百度等著名网站,也包括德意志银行的网上银行、华尔街在线的金融信息发布系统等。目前有超过半数的 Ajax-enabled 和 Web 2.0 站点选择使用 PHP,PHP 正在成为“互联网最流行的开发语言”。

MySQL 是流行的开源关系型数据库管理系统之一,是一个高速、可靠、可扩展、易使用的数据库管理系统。MySQL 一直专注于速度、可靠性、扩展性和易用性,而且在发展过程中,不断引入了一些企业级特性,包括复制、全文搜索、事件、外键完整性、行级锁定等。同时在 MySQL 5 中,还进一步引入了视图、存储过程、触发器等高级特性,使得 MySQL 能满足企业级应用的需要。

学习程序设计最有效、最快速的方法是:在掌握必要的基础知识后,通过实际的应用实例掌握程序设计的完整流程和开发技巧。本书选择用 PHP 5 与 MySQL 5 进行 Web 应用开发,结合简短的实例,对基础和核心技术进行详细介绍,使读者能够迅速掌握必要的基础知识。之后,通过一易一难两个完整的应用实例,展示了用 PHP 5 与 MySQL 5 进行 Web 应用开发的完整流程,以及相应的设计方法、技巧。

本书的顺利出版源于许多人的努力,而不仅仅是封面上的两位作者。在这里要感谢电子工业出版社相关同志的帮助、支持与理解,没有他们,这本书不可能顺利出版。还要感谢我们的家人、同事、朋友,感谢你们给予的支持。

由于作者水平有限,书中疏漏和不妥之处在所难免,恳请广大读者批评指正。

为了方便读者阅读,本书配套资料请登录“华信教育资源网”(<http://www.hxedu.com.cn>),在“资源下载”频道的“图书资源”栏目下载。

目 录

第 1 章 PHP 5 基础.....	1	2.5 表达式.....	47
1.1 简介.....	1	2.6 运算符.....	47
1.1.1 PHP 是什么.....	1	2.6.1 算术运算符.....	48
1.1.2 初步认识 PHP 代码.....	1	2.6.2 字符串运算符.....	50
1.2 PHP 的安装.....	2	2.6.3 比较运算符.....	51
1.2.1 Linux 下安装 Apache 和 PHP.....	2	2.6.4 逻辑运算符.....	52
1.2.2 Windows 下安装 Apache 和 PHP.....	4	2.6.5 位运算符.....	52
1.3 PHP 配置.....	7	2.6.6 赋值运算符.....	52
1.3.1 PHP 配置文件.....	8	2.6.7 数组运算符.....	53
1.3.2 PHP 常用配置选项.....	8	2.6.8 类型运算符.....	55
1.3.3 加载扩展库.....	11	2.6.9 错误控制运算符.....	56
1.4 本章小结.....	12	2.6.10 执行运算符.....	56
第 2 章 网站开发语法基础.....	13	2.6.11 运算符的优先级.....	56
2.1 基本语法.....	13	2.7 流程控制.....	58
2.2 数据类型.....	15	2.7.1 if 结构.....	58
2.2.1 布尔型.....	17	2.7.2 switch 结构.....	61
2.2.2 整型.....	18	2.7.3 while 循环.....	65
2.2.3 浮点型.....	21	2.7.4 for 循环.....	68
2.2.4 字符串.....	22	2.7.5 foreach 语句.....	70
2.2.5 数组.....	27	2.7.6 declare 语句.....	71
2.2.6 对象.....	28	2.7.7 exit 和 return 语句.....	72
2.2.7 资源.....	29	2.8 包含代码.....	72
2.2.8 NULL.....	30	2.9 本章小结.....	76
2.2.9 类型转换.....	30	第 3 章 函数.....	78
2.3 变量.....	31	3.1 用户自定义函数.....	78
2.3.1 变量的赋值.....	31	3.1.1 函数调用与函数定义.....	79
2.3.2 预定义变量.....	33	3.1.2 递归函数.....	80
2.3.3 变量作用域.....	38	3.2 函数的参数.....	81
2.3.4 可变变量.....	43	3.2.1 按值传递参数.....	81
2.4 常量.....	44	3.2.2 通过引用传递.....	81
2.4.1 自定义常量.....	44	3.2.3 默认参数值.....	83
2.4.2 魔术常量.....	46	3.2.4 可变长度参数列表.....	84
		3.3 函数返回值.....	86

3.4	变量函数	87	5.2.6	HTML 元素	126
3.5	匿名函数	88	5.2.7	处理反斜杠	128
3.6	内建函数	90	5.2.8	比较字符串	129
3.7	本章小结	90	5.2.9	子串	130
第 4 章	类与对象	91	5.2.10	字符串查找	131
4.1	面向对象编程	91	5.2.11	字符串的分解和合并	134
4.2	PHP 的对象模型	91	5.2.12	格式化字符串	135
4.3	对象的创建和使用	92	5.2.13	分解 URL	136
4.3.1	创建对象	92	5.3	正则表达式	137
4.3.2	访问属性和方法	92	5.3.1	基本模式匹配	137
4.3.3	复制对象	93	5.3.2	字符簇	138
4.3.4	对象的比较	94	5.3.3	重复出现	139
4.3.5	自动加载对象	95	5.3.4	POSIX 正则表达式的处理	
4.4	类的定义	96	函数		140
4.4.1	类的属性	96	5.3.5	Perl 兼容正则表达式的处理	
4.4.2	类的方法	97	函数		141
4.4.3	范围解析操作符 (::)	98	5.4	本章小结	142
4.4.4	静态成员	99	第 6 章	数组	143
4.4.5	定义常量	101	6.1	数组基础	143
4.4.6	继承	102	6.1.1	创建数组	143
4.4.7	抽象类和抽象方法	104	6.1.2	访问数组成员	143
4.4.8	接口	105	6.2	输出一个数组	144
4.4.9	构造函数和析构函数	106	6.3	数组操作	145
4.4.10	访问限制	109	6.3.1	遍历数组	145
4.5	重载	112	6.3.2	导出数组	148
4.5.1	属性重载	113	6.3.3	分割数组	151
4.5.2	方法重载	115	6.3.4	合并数组	153
4.6	对象串行化	116	6.3.5	数组的键值处理	154
4.7	本章小结	118	6.3.6	反转数组	155
第 5 章	字符串	119	6.3.7	获取部分数组成员	157
5.1	字符串常量	119	6.3.8	数组排序	158
5.1.1	转义序列	119	6.3.9	数组的交集	161
5.1.2	解析变量	120	6.3.10	数组的差集	162
5.2	字符串操作	122	6.3.11	过滤数组成员	163
5.2.1	访问单个字符	123	6.3.12	交换数组的键值	166
5.2.2	求字符串长度	123	6.3.13	处理整个数组	166
5.2.3	去除空白字符	123	6.4	本章小结	168
5.2.4	字符串大小写变换	125	第 7 章	PHP 高级应用技术	169
5.2.5	ASCII 编码转换	125	7.1	上传文件	169

7.1.1 上传文件的表单	169	9.2 安装 MySQL	222
7.1.2 PHP 处理上传文件	169	9.2.1 在 Linux 上安装 MySQL	222
7.1.3 上传文件实例	170	9.2.2 在 Windows 上安装 MySQL	223
7.1.4 上传文件类	172	9.3 配置 MySQL	224
7.2 图像处理	176	9.4 可视化管理工具	
7.2.1 GD 库的加载	176	phpMyAdmin	225
7.2.2 GD 库函数	177	9.4.1 phpMyAdmin 安装	225
7.2.3 用 PHP 制作水印	180	9.4.2 phpMyAdmin 配置	228
7.3 Ajax 技术	187	9.5 SQL 基础	229
7.3.1 Ajax 技术简介	187	9.5.1 SQL 简介	229
7.3.2 Ajax 的工作原理	188	9.5.2 MySQL 支持的 SQL 语句	230
7.3.3 XMLHttpRequest 类	188	9.6 数据库的操作	230
7.3.4 Ajax 与 PHP 结合	192	9.6.1 创建数据库	230
7.4 Socket 编程	195	9.6.2 选择数据库	232
7.4.1 Socket 库的加载	195	9.6.3 删除数据库	232
7.4.2 Socket 原理	195	9.7 数据表的操作	233
7.4.3 Socket 函数和用法	196	9.7.1 创建表	233
7.4.4 Socket 实例	198	9.7.2 修改表	237
7.5 文件系统	202	9.7.3 删除表	237
7.5.1 文件路径处理	203	9.8 数据的操作	238
7.5.2 操作目录	204	9.8.1 添加记录	238
7.5.3 操作文件	206	9.8.2 修改记录	239
7.6 本章小结	209	9.8.3 删除记录	240
第 8 章 PHP 模板	210	9.9 查询数据	240
8.1 PHP 模板简介	210	9.9.1 选择列	241
8.2 Smarty 模板引擎的特性	210	9.9.2 WHERE 子句	241
8.3 Smarty 模板引擎的安装简介	211	9.9.3 GROUP BY 子句	243
8.4 Smarty 模板引擎的基本语法	211	9.9.4 ORDER BY 子句	244
8.5 Smarty 模板引擎实例	212	9.9.5 LIMIT 子句	246
8.6 Smarty 模板高级应用	216	9.10 数据库, 表信息查询	246
8.6.1 嵌套输出	216	9.10.1 SHOW 语句	246
8.6.2 表格输出	218	9.10.2 DESCRIBE / EXPLAIN 语句	248
8.7 本章小结	220	9.11 PHP 中的 MySQL 操作函数	248
第 9 章 MySQL 快速入门	221	9.12 本章小结	250
9.1 MySQL 简介	221	第 10 章 MySQL 数据库管理	251
9.1.1 MySQL 特性	221	10.1 启动 MySQL 服务器	251
9.1.2 MySQL 的应用	222	10.1.1 通过命令行启动	251
9.1.3 MySQL 的管理	222	10.1.2 通过 Windows 服务启动	251
9.1.4 MySQL 的存储引擎	222		

10.2 命令行下管理 MySQL	252	12.4.1 模板类的包装	278
10.2.1 查看 mysql.exe 的选项	252	12.4.2 分页类	279
10.2.2 启动 mysql.exe	253	12.5 数据库的准备	284
10.2.3 退出 mysql.exe	254	12.5.1 创建数据库	284
10.3 MySQL 账户管理	254	12.5.2 数据库连接	286
10.3.1 添加 MySQL 账户	254	12.6 权限系统	286
10.3.2 删除 MySQL 账户	256	12.6.1 添加管理员	287
10.3.3 修改密码	256	12.6.2 删除管理员	289
10.4 MySQL 权限系统	257	12.6.3 管理员列表	290
10.4.1 权限系统的工作原理	257	12.6.4 后台登录	294
10.4.2 访问控制	258	12.6.5 管理员退出	297
10.4.3 GRANT 语句和 REVOKE 语句	259	12.7 文章分类管理	298
10.5 本章小结	262	12.7.1 添加文章分类	298
第 11 章 MySQL 5 进阶	263	12.7.2 删除文章分类	300
11.1 MySQL 的 5 新特性	263	12.7.3 编辑文章分类	300
11.2 存储过程	263	12.7.4 文章分类列表	300
11.2.1 存储过程简介	263	12.7.5 文章分类管理文件	300
11.2.2 创建存储过程	264	12.8 文章管理	307
11.2.3 调用存储过程	267	12.8.1 添加文章	307
11.2.4 查看存储过程	269	12.8.2 编辑文章	311
11.2.5 删除存储过程	269	12.8.3 删除文章	314
11.3 触发器	270	12.8.4 文章列表	315
11.3.1 创建触发器	270	12.8.5 文章管理文件	316
11.3.2 删除触发器	271	12.9 后台管理框架	321
11.4 视图	271	12.9.1 首页设计	321
11.4.1 创建视图	271	12.9.2 菜单文件	322
11.4.2 修改视图	272	12.9.3 后台首页界面	326
11.4.3 删除视图	273	12.10 前台输出	326
11.5 PHP 调用存储过程	273	12.10.1 文章分类显示	326
11.5.1 mysqli 扩展	273	12.10.2 文章列表	328
11.5.2 使用 mysqli 扩展调用存储 过程	273	12.10.3 文章正文	334
11.6 本章小结	276	12.10.4 前台首页	341
第 12 章 文章发布系统	277	12.2 本章小结	342
12.1 需求描述	277	第 13 章 电子商务系统	343
12.2 系统设计	277	13.1 需求描述	343
12.3 代码框架	277	13.2 系统设计	343
12.4 类库的准备	278	13.3 代码框架	344
		13.4 类库的准备	344
		13.4.1 数据库类的封装	344

13.4.2 分页类的封装	356	13.10.1 指定服务器的价格方 案列表	442
13.4.3 模板类封装	356	13.10.2 添加价格方案	447
13.5 数据库准备	356	13.10.3 修改价格方案信息	451
13.6 后台管理员	361	13.10.4 删除价格方案	453
13.6.1 权限判断	361	13.10.5 查找价格方案	454
13.6.2 添加管理员	366	13.10.6 批量修改价格方案	465
13.6.3 管理员列表	368	13.10.7 价格方案管理类	471
13.6.4 删除管理员	372	13.10.8 价格方案管理菜单	475
13.6.5 界面人性化和退出后台	373	13.11 客户订单系统	476
13.7 用户管理	377	13.11.1 前台页面分割	476
13.7.1 用户注册	377	13.11.2 前台首页	483
13.7.2 用户管理	393	13.11.3 登录下单	487
13.7.3 坏单用户列表	401	13.12 订单管理	506
13.7.4 用户菜单栏	402	13.12.1 添加订单	506
13.7.5 用户操作类	403	13.12.2 后台订单列表	512
13.8 游戏管理	406	13.12.3 结算订单	516
13.8.1 添加游戏	406	13.12.4 标记坏单	519
13.8.2 游戏列表	414	13.12.5 分类列表	520
13.8.3 修改游戏信息	415	13.12.6 订单管理类	521
13.8.4 游戏管理类	419	13.12.7 订单管理菜单	523
13.8.5 游戏管理菜单栏	420	13.13 后台管理框架	524
13.9 服务器管理	421	13.14 前台用户管理	525
13.9.1 指定游戏的服务器列表	421	13.14.1 客户订单查询	525
13.9.2 添加服务器	426	13.14.2 客户个人资料修改	533
13.9.3 修改服务器信息	428	13.14.3 修改密码	538
13.9.4 删除服务器	431	13.14.4 前台用户管理菜单	542
13.9.5 查找服务器	432	13.15 前台首页	543
13.9.6 服务器管理类	438	13.16 本章小结	544
13.9.7 查找服务器菜单	441		
13.10 价格方案管理	442		

第1章 PHP 5 基础

本章内容包括:

- PHP 简介
- PHP 安装
- PHP 配置

1.1 简介

学习任何语言,最好的方法是从概貌开始。在开始学习使用 PHP 编写 Web 应用程序之前,我们有必要对 PHP 的起源、特点、能做什么、如何安装与配置等内容进行简单了解。本节将围绕这些内容展开,使读者对 PHP 有一个总体的认识,并希望以此提高读者进一步学习 PHP 的热情。

1.1.1 PHP 是什么

PHP——是英文 Hypertext Preprocessor (超文本预处理器)的缩写,是一种广泛使用的服务器端编程语言,用于开发动态网页。它是一种开源的、跨平台的、独立于架构的、解释的、面向对象的、快速的、简单易学的、健壮的、安全的、性能优越的用于 Web 服务器端动态网页开发的语言。目前已有超过上千万的网站和数万家公司,以及数百万的程序开发人员在使用它。它是目前动态网页开发中使用最为广泛的语言之一。

PHP 是一种 Web 服务器端开发语言,用于快速进行动态网页开发。早期的 Web 应用服务器端程序是用 Perl 或 C 语言直接编写 CGI (Common Gateway Interface, 公共网关接口) 程序,或者利用 Web 服务器自带的 API,或者采用第三方解决方案。这些方法均有各自的强项,但在运行速度、数据库操作方面均不能满足 Web 应用不断增长的需求。因此,PHP 与 ASP、JSP 语言以其优异的性能、强大的数据库操作功能迅速成为主流的 Web 服务器端开发语言。PHP 与 ASP、JSP 一起,几乎覆盖了所有的 Web 服务器端应用的开发。而 PHP + Apache + MySQL 是一个完全免费的、性能优异的 Web 服务器应用开发组合,已经成为绝大多数中小型网站的应用解决方案。

与 ASP 和 JSP 不同,PHP 是一个开放源代码的项目,所以没有购买许可证的费用或限制使用的问题。你可以使用 PHP 来运行小的、非赢利性站点,或者运行十亿美元的电子商务网站,而且成本都是零。不仅如此,如果想要或需要修改 PHP,可以修改它。PHP 并没有得到 GPL (GNU 通用公共许可证) 的许可,但它自己的许可证允许重新分发代码和 / 或二进制文件。

1.1.2 初步认识 PHP 代码

了解了 PHP 是什么之后,现在就来看看 PHP 代码是什么样子。下面就给出一个简单的例

子程序，对 PHP 进行最初的认识。

例 1.1 第一个 PHP 程序(I_am_php.php)。

```
1  <html>
2  <head>
3  <title>Example</title>
4  </head>
5  <body>
6  <?php
7  echo "I am PHP!";
8  ?>
9  </body>
10 </html>
```

鉴于后面将对 PHP 代码进行详细介绍，这里仅简单给出上述程序的运行结果，即最后在浏览器中输出：I am PHP!。

从上述代码清单中可以看到，用户可以直接在 Web 页面中输入 PHP 命令代码，因而不需要用任何特殊的开发环境。在 Web 页面中，所有 PHP 代码都被放置在“<?php”和“?>”中。此外，用户还可以选择使用诸如<SCRIPT LANGUAGE="php"></SCRIPT>等的形式。PHP 引擎会自动识别并处理页面中所有位于 PHP 定界符之间的代码。

1.2 PHP 的安装

要运行 PHP 程序，需要先准备好三个东西：

- Web 服务器。本书中，我们选择 Apache 作为 Web 服务器，可以在 Apache 官方网站 www.apache.org 上面下载到。

- PHP 解释器。本书中，我们选择 PHP 5.2，在 PHP 官方网站 (www.php.net) 上面可以下载到。

- Web 浏览器。现在常用的 IE 浏览器以及 Firefox 都可以满足要求。

大多数常规 PHP 安装的通常是与 Linux 或各种 UNIX 上的 Apache 一起运行的 PHP 模块。但这不是唯一的选择，由于 Apache 和 PHP 良好的跨平台性，它们还可以在很多操作系统上进行安装，下面，我们着重介绍 Apache 2.2 和 PHP 5.2 在 Linux 以及 Windows 下的安装。

1.2.1 在 Linux 下安装 Apache 和 PHP

下面介绍的安装方式为直接通过源代码进行安装。在准备好 Apache 和 PHP 的源代码包之后，就可以输入下面的步骤代码进行 Apache 和 PHP 的基本安装了：

```
1  gzip -d httpd-2_2_NN.tar.gz
2  tar xvf httpd-2_2_NN.tar
3  gunzip php-NN.tar.gz
```

```

4  tar -xvf php-NN.tar
5  cd httpd-2_2_NN
6  ./configure --enable-so
7  make
8  make install

```

注意

现在已经将 Apache 2.2.NN 安装在 /usr/local/apache 目录下了。本安装支持装载模块和标准的 MPM prefork 模块。之后, 可以使用如下命令启动 Apache 服务器: /usr/local/apache2/bin/apachectl start。如果成功, 可以停止 Apache 服务器并继续安装 PHP: /usr/local/apache2/bin/apachectl stop。

```

9  cd ../php-NN

```

提示

输入上面的第 9 条语句后, 可以为你的 PHP 定制不同的选项, 例如要加载哪些扩展库。使用 ./configure --help 可列出可用的选项。下面的步骤语句只是简单地配置 Apache 1 与 MySQL 支持。你的 apxs 的路径可能不同。

```

10 ./configure --with-mysql --with-apsx=/www/bin/apsx
11 make
12 make install

```

注意

如果在安装之后你决定修改配置选项, 那么只需重复输入以上最后三条语句, 重新启动 Apache 就可以使新模块生效, 不需要重新编译 Apache。注意除非特别指出, “make install” 总是会安装上 PEAR、各种 PHP 工具例如 phpize、PHP CLI 以及其他项目。

```

13 cp php.ini-dist /usr/local/lib/php.ini

```

注意

输入以上命令即建立了你自己的 php.ini 文件。你可以编辑 php.ini 来设置 PHP 选项。如果你想把 php.ini 放在其他目录, 可在第 10 条步骤语句中加上以下内容: --with-config-file-path=/path。你也可以选择 PHP 推荐的配置文件 php.ini-recommended。需要注意的是, 不同的配置文件将影响 PHP 的运行状况。

提示

你已经分别安装好了 Apache 服务器和 PHP 的运行程序, 现在需要把二者结合起来, 通过编辑 httpd.conf 来加载 PHP 模块。在 LoadModule 语句右边的路径必须指向系统中 PHP 模块的

路径。以上的“make install”步骤可能已经在 httpd.conf 中添加了这些内容，但仍然需要检查一下。

```
14 LoadModule php5_module libexec/libphp5.so
15 AddModule mod_php5.c
16 AddType application/x-httpd-php .php
```

注意

上面的语句就确保了 Apache 会把后缀名为 .php 的文件当做 PHP 文件来解析，当然你也可以增加解析其他后缀名文件的功能，如解析后缀名为 .phtml 的文件，用下面的语句实现：AddType application/x-httpd-php .php .phtml。注意用空格分隔不同类型的文件。

17 正常重启 Apache

1.2.2 在 Windows 下安装 Apache 和 PHP

Apache 是当前最流行的网站服务器之一，在互联网上有非常广泛的应用，其中 Apache 2.2 是最新的分支，可以在 Windows NT 4.0、Windows 2000、Windows XP 或者 Windows Server 2003 上运行，需要注意的是，Apache 2.2 不能在 Windows 9x 下运行。下面就在 Windows XP 下搭建我们的测试平台，首先从 <http://httpd.apache.org> 网站上下载 Apache，目前它的最新版本是 Apache 2.2.4。然后下载 Win32 Binary (MSI Installer) 版本：apache_2.2.4-win32-x86-no_ssl.msi。这是一个 Windows 安装程序，双击运行该程序，进行安装。

安装完成后，Apache 将自动启动，如果启动成功，则会在任务栏右下角显示 Apache 监视器 (ApacheMonitor.exe) 图标，表示 Apache 运行正常。

默认情况下，每次计算机启动时，Apache 会自动运行，如果 Apache 没有运行，可以通过开始菜单或者执行安装目录下的 \Apache2.2\bin\httpd.exe 文件来启动，而 Apache 监视器则可以通过 \Apache2.2\bin\ApacheMonitor.exe 来启动。

成功启动 Apache 服务器之后，可以在浏览器地址栏输入 <http://localhost/> 或者 <http://127.0.0.1/> 来访问本地网站服务器，运行结果为如图 1.1 所示的页面，表示 Apache 已正常工作。这里“localhost”和“127.0.0.1”表示本机地址。

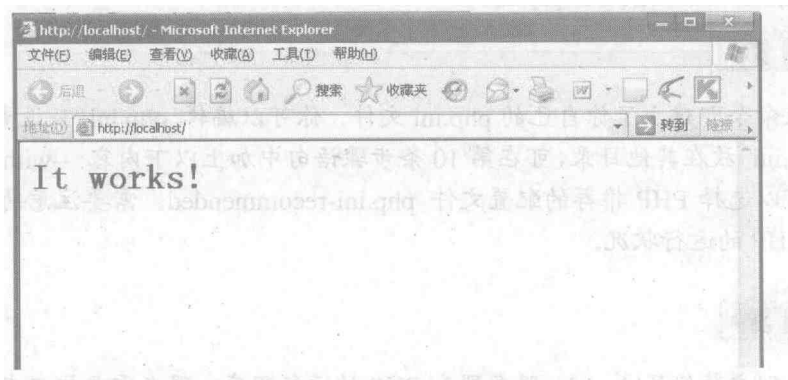


图 1.1 Apache 正常运行页面

接下来,在 PHP 官方网站 www.php.net 上下载 PHP 5.2.4 的 Win 32 版本二进制包: `php-5.2.4-Win32.zip`, 下载完成之后,我们把它解压到 `c:\php` 目录下,当然,也可以把它解压到计算机上的其他位置,只需要在下面的操作中注意将路径做相应修正就可以了。PHP 5.2.4 的二进制包解压出来之后,可以看到其目录结构如下:

```
c:\php
|
|--dev
| |
| |php5ts.lib
|
|--ext                -- PHP 的扩展 DLL 文件
| |
| |php_bz2.dll
| |
| |php_cpdf.dll
| |
| |..
|
|--extras
| |
| |++mibs                -- SNMP 支持文件
| |
| |+++openssl            -- Openssl 支持文件
| |
| |++pdf-related        -- PDF 支持文件
| |
| |l-mime.magic
|
|--pear                -- PEAR 的初始复本
|
|
|l-go-pear.bat        -- PEAR 安装脚本
|
|l-fdftk.dll
|
|..
|
|l-php-cgi.exe        -- CGI 可执行文件
```

```

|
| -l-php-win.exe          -- 执行脚本时不弹出命令提示行窗口
|
| -l-php.exe             -- CLI 可执行程序 - 仅使用在命令行脚本
|
|
| -l-..
|
| -l-php.ini-dist        -- 默认的 php.ini 设置
|
| -l-php.ini-recommended -- 推荐的 php.ini 设置
|
|
| -l-php5activescript.dll
|
| -l-php5apache.dll
|
| -l-php5apache2.dll
|
| -l-..
|
| -l-php5ts.dll          -- 核心 PHP DLL 文件
|
| -l-...

```

提示

在 Apache 配置文件里面编辑 Windows 路径的时候，所有的反斜杠都要转换成斜杠，如 c:\directory\file.ext 必须写成 c:/directory/file.ext。

PHP 在 Apache 下面可以通过 CGI 方式或者 Apache 模块方式来运行，由于 CGI 方式的效率不够高而且这种方式下服务器防御攻击的能力相对低下，一般采用 Apache 模块方式来运行 PHP。要把 PHP 以 Apache 模块的方式运行，只需要在你的 Apache 配置文件 httpd.conf 中加入下面的语句即可：

```

LoadModule php5_module "c:/php/php5apache2_2.dll"
AddType application/x-httpd-php .php
PHPIniDir "c:/php/"

```

注意

在这里，需要确认 PHP 目录里面存在动态链接库文件 php5apache2_2.dll；我们指定了读取 PHP 配置文件 php.ini 的位置为 c:\php\，现在需要在 c:\php\ 中创建 PHP 配置文件 php.ini，一般的做法是将 php.ini-dist 更名为 php.ini，作为 PHP 的默认配置文件。注意，切记以你实际所用

的 PHP 目录代替文中的 c:/php/。

通过上面的操作，我们成功配置了 PHP 5 的运行平台，重启 Apache 服务器之后，就可以正常运行 PHP 脚本了。下面我们在服务器目录中创建 info.php 文件，输入如下代码：

```
1  <?php
2  phpinfo();
3  ?>
```

提示

phpinfo()是一个 PHP 内建函数，生成一个展示 PHP 安装情况的完整信息的 HTML 页。可以使用该函数查看是否安装了特定的扩展，或 php.ini 文件是否被定制。

在浏览器中输入正确的路径，即可以看到 info.php 的输出页面。如图 1.2 所示为输出页面的一部分。

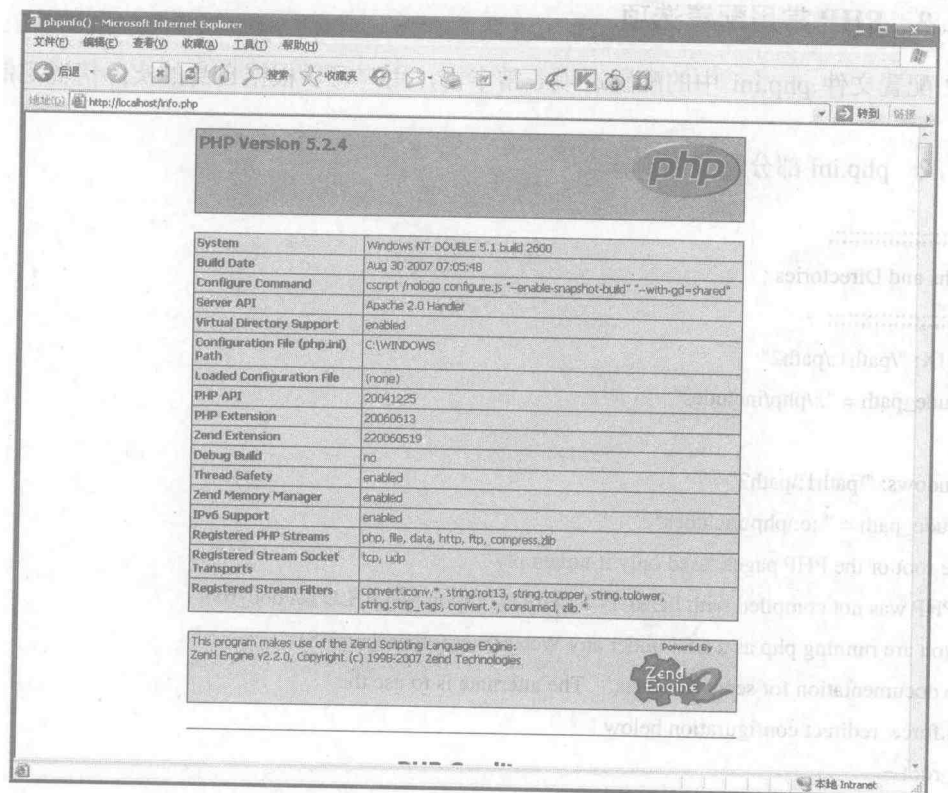


图 1.2 info.php 的部分输出页面

1.3 PHP 配置

现在可以使用已安装好 PHP 的 Apache 服务器了。如果你想改变 PHP 的配置，只需要修改 php.ini 文件并重启 Apache 服务器。切记每次修改了 PHP 的设置都需要重启 Apache 服务器来

使这些改变生效。

1.3.1 PHP 配置文件

配置文件 `php.ini` 在 PHP 启动时被读取。对于作为服务器模块版本的 PHP，仅在 Web 服务器启动时读取一次。`php.ini` 的搜索路径如下（按顺序）：

- Apache 2 中的 `PHPIniDir` 指定的路径。
- `HKEY_LOCAL_MACHINE\SOFTWARE\PHP\IniFilePath`（Windows 注册表位置）。
- `PHPRC` 环境变量。
- Web 服务器目录或 PHP 所在目录（Windows 下其他情况）。
- Windows 目录（`c:\windows` 或 `c:\winnt`）。

本文中，由于在前面的 Apache 配置文件 `httpd.conf` 中，我们已经指定了 `PHPIniDir` 项目，所以服务器会读取 `c:\php\php.ini` 作为 PHP 配置文件。你可以通过 `info.php` 输出页面中的 `Loaded Configuration File` 项的值来判断现在服务器读取的 PHP 配置文件是哪一个文件。

1.3.2 PHP 常用配置选项

PHP 配置文件 `php.ini` 中的配置选项非常丰富，用户可以根据自身的实际情况定制和优化 PHP。

例 1.2 `php.ini` 部分配置选项。

```

.....
; Paths and Directories ;
.....
; UNIX: "/path1:/path2"
include_path = "./php/includes"
;
; Windows: "\path1;\path2"
include_path = ".;c:\php\includes"
; The root of the PHP pages, used only if nonempty.
; if PHP was not compiled with FORCE_REDIRECT, you SHOULD set doc_root
; if you are running php as a CGI under any Web server (other than IIS)
; see documentation for security issues.  The alternate is to use the
; cgi.force_redirect configuration below
doc_root =
; The directory under which PHP opens the script using /-username used only
; if nonempty.
user_dir =
; Directory in which the loadable extensions (modules) reside.
extension_dir = "."
extension_dir = "C:\php\ext\"
```