

PHP & MySQL Web

网络编程

郭金锋 林宇 等编著

人民邮电出版社
www.pptph.com.cn

PHP & MySQL Web 网络编程

郭金锋 林 宇 等编著

人 民 邮 电 出 版 社



北航 C0533657

内容提要

PHP 是当今非常流行的 Web 服务器脚本语言之一,而 MySQL 是当今非常流行的网络数据库之一,它们的结合为建设功能强大的 Web 网站打下了坚实的基础。PHP 嵌入在 HTML 语句中,能够根据用户的要求,利用后台数据库中的数据灵活地、动态地生成网页。

本书的内容分为三个部分。第一篇介绍 PHP 语言的基本使用方法,包括 PHP 语言的基本概念、安装和基本语法;第二篇介绍 MySQL 数据库的基本原理,讲解 MySQL 数据库的安装、操作语言、操作函数等内容;第三篇结合前两篇的知识,用大量实例来讲解如何创建实际的 Web 应用程序。

书中含有大量的实例代码,内容详实,深入浅出,不仅适合于 Web 编程的初学者,对于有一定经验的 Web 程序员也有极好的参考价值。

PHP & MySQLWeb 网络编程

◆ 编 著 郭金锋 林 宇 等

责任编辑 梁 凝

◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号

邮编 100061 电子函件 315@pptph.com.cn

网址 <http://www.pptph.com.cn>

北京汉魂图文设计有限公司制作

北京鸿佳印刷厂印刷

新华书店总店北京发行所经销

◆ 开本:787×1092 1/16

印张:32.5

字数:814 千字

印数:1-5 000 册

2001 年 3 月第 1 版

2001 年 3 月北京第 1 次印刷

ISBN 7-115-09126-9/TP·2083

定价:48.00 元

前言

由于 PHP 语言和 MySQL 数据库的组合如此完美,能够很好地创建几乎所有的 Web 应用程序,并且使用灵活,代码可移植性高,因此作为一名 Web 开发人员,我很希望能够向读者推荐它们。PHP 的源程序代码是开放的,全世界很多人都在使用它进行网站建设,所以会不断有新鲜的东西补充到它的体内,其功能像滚雪球一样快速强大起来。这些,正是我们编著本书的原因。

本书基于边学习边动手编程的原则,采用由浅入深的方法,从最简单的程序出发,逐渐展开介绍 PHP 和 MySQL 的特点与使用方法。这种安排符合人们学习新知识的规律,使读者能够很快进入角色,亲手编写各种应用程序。

本书读者对象

本书假设读者已经对 HTML 语言有所了解,并有一定的 Internet 和 Web 知识。

本书既适合于刚刚开始进行 Web 开发的初学者,也同样适合有一定经验的程序员,这是由本书的内容安排特点所决定的。对于 Web 编程的初学者,可以从本书的开头开始学习本书,首先学习 PHP 的语言基础,并对 MySQL 数据库有所了解,在这个基础上进入实际的网站开发。而对于有经验的程序员,可以在大体了解第三章 PHP 速成的有关知识后进入第三篇的实际应用开发阶段,把主要精力放在更复杂的综合应用章节之上,在了解了基本的应用原理之后编写复杂的网络应用程序。

本书中有大量的实例可供读者在学习时参考,你必须实际运行一些例子才能真正理解相应的概念。而且,在编写实例的过程中,你会感受到很多乐趣。

本书的组织结构

本书分为三篇。第一篇讲述 PHP 语言基础,第二篇讲解 MySQL 数据库的使用,而第三篇是在前两部分的基础上开发实际的网络应用程序。

第一篇中含有 3 章。第一章“PHP 语言概述”讲述了 PHP 的基本概念和 workflows。第二章讲述 PHP 在各种操作系统中的安装方法。第三章使用大量的例子来详细讲解 PHP 的语法问题,使你能够在比较短的时间内达到“PHP 速成”。

第二篇也包括 3 章,第四章讲述 MySQL 数据库的基本知识,使你对它有清楚的了解;第五章教你如何使用 MySQL 数据库,你将学习到很多数据库的操作知识。第六章“MySQL 数据库的管理和数据库的安全”讲述管理数据库的工具和使用方法,并对数据库的安全问题进行详细讲解。

第三篇进入到实际开发阶段,它共有 8 章。第七章讲述如何使用 PHP 语言来编写 CGI 程序,详细说明了 CGI 的基本概念、工作原理等;第八章讲述如何使用 PHP 语言来操作 MySQL 数据库,并介绍了 phpMyAdmin 数据库管理软件的原理和使用。从第九章开始进入实际的网络应用,第九章讲述如何对网站用户进行管理,主要是对用户的身份进行认证;第十章讲解电子邮件的基本原理,并用实例说明如何使用 PHP 建立 Web 邮件系统。第十一章,“建立聊天室”,用实例说明如何使用 PHP 和 MySQL 数据库建立聊天室。第十二章讲述了网站调查

常用的投票程序。第十三章讲解网站讨论区的原理及其 PHP 和 MySQL 实现。最后，第十四章讲述如何建立一个网站，以及建立网站过程中需要注意的一些问题，包括用户会话管理、并发和网站安全问题。

本书由高波策划，郭金锋、林宇等执笔编写，由于时间和水平有限，不足之处在所难免，敬请读者斧正。

作者

目 录

第一篇 PHP 语言

第一章 PHP 语言概述	2
1.1 什么是 PHP	2
1.2 PHP 语言的发展历史	2
1.3 PHP 的特点	3
1.4 PHP 的工作机制	4
1.5 简单的 PHP 程序介绍	5
1.6 总结	9
第二章 建立 PHP 工作环境	10
2.1 安装 PHP 的准备工作	10
2.2 在 Linux 系统中安装 PHP	11
2.3 在 Windows 操作系统中安装 PHP	13
2.3.1 在 Win 98 操作系统中安装 PHP	13
2.3.2 在 Win 2000 下安装 PHP	16
2.4 总结	16
第三章 PHP 速成	17
3.1 PHP 语言基本语法规则	17
3.1.1 PHP 语言嵌入 HTML 的方式	18
3.1.2 在 PHP 语句中添加注释	19
3.1.3 PHP 语句标记	19
3.2 PHP 中的数据类型、运算符和表达式	19
3.2.1 PHP 语言中的数据类型	20
3.2.2 运算符和表达式	31
3.2.3 PHP 语言中的语句	41
3.2.4 小结	42
3.3 流程控制	42
3.3.1 程序的基本结构	43
3.3.2 if() ... else 语句	44
3.3.3 switch 语句	48
3.3.4 循环控制语句	50
3.3.5 break 语句和 continue 语句	54

3.3.6	exit 语句和 return 语句	55
3.3.7	小结	56
3.4	函数	56
3.4.1	函数的定义	57
3.4.2	函数的参数	58
3.4.3	return 语句	62
3.4.4	函数的调用	64
3.4.5	变量的作用字段和静态变量	68
3.4.6	小结	72
3.5	常用的系统函数	72
3.5.1	输入/输出函数	73
3.5.2	文件操作函数	78
3.5.3	数据操作函数	86
3.5.4	数学函数	102
3.5.5	时间和日期函数	106
3.5.6	图像函数	111
3.5.7	关于 PHP 函数库的说明	118
3.5.8	小结	120
3.6	输入/输出系统	120
3.6.1	信息的输入	121
3.6.2	信息的输出	124
3.6.3	使用 Cookies	125
3.6.4	PHP 中的环境变量	128
3.6.5	文件操作	128
3.6.6	小结	133
3.7	面向对象技术	134
3.7.1	面向对象的基本概念	134
3.7.2	在 PHP 语言中使用类和对象	139
3.7.3	面向对象总结	145
3.8	总结	145

第二篇 MySQL 数据库

第四章	MySQL 数据库概述	148
4.1	MySQL 数据库的特征	148
4.2	MySQL 数据库的安装	149
4.2.1	安装 MySQL 二进制代码分发	149
4.2.2	Linux RPM 安装	150
4.2.3	安装 MySQL 源代码分发	150
4.3	MySQL 数据库的基本结构	151

4.3.1	网络功能的实现和通信的方式	151
4.3.2	数据库的本地结构	152
4.3.3	数据表的构成	153
4.4	MySQL 支持的数据类型	153
4.4.1	列类型概述	153
4.4.2	字符串类型	155
4.4.3	数值列类型	160
4.4.4	日期和时间类型	161
4.4.5	NULL 空值	162
4.5	数据库设计的基本概念	162
4.5.1	数据模型设计的基本过程	162
4.5.2	简单的数据库设计的实例	163
4.5.3	连接到 MySQL 服务器	164
4.5.4	口令的安全问题	165
4.5.5	构建实践的数据库环境	166
4.5.6	查看创建的结果	172
4.6	总结	174
第五章	MySQL 数据库的使用	175
5.1	SQL 语言的基本概念	175
5.2	MySQL 中数据定义 SQL 语言的使用	175
5.2.1	创建、删除和选择数据库	176
5.2.2	创建、删除基本表	176
5.2.3	修改表的结构	183
5.2.4	使用 OPTIMIZE TABLE 优化表	185
5.2.5	创建、删除和修改索引	186
5.3	MySQL 中数据操作语句的使用	188
5.3.1	基本信息的检索	188
5.3.2	高级信息的检索	196
5.3.3	获取关于 SELECT 的信息	207
5.3.4	增加和删除记录	210
5.4	环境设定相关的 SQL 语句	216
5.4.1	设置服务器/客户连接的参数	216
5.4.2	数据刷新和终止线程	217
5.5	性能的优化	221
5.5.1	系统瓶颈	222
5.5.2	索引的选择	223
5.5.3	选择表类型	225
5.6	多媒体数据的组织方式	229
5.6.1	直接存放方式	229

5.6.2 间接存放的方式	230
5.7 MySQL 数据库的局限	231
5.7.1 事务的提交/会滚功能	231
5.7.2 嵌套查询	232
5.7.3 存储过程	233
5.7.4 触发器	233
5.7.5 视图	234
5.7.6 记录级的锁定和访问权限的设定	234
5.7.7 外部键和引用来增强数据的完整性检查	235
5.8 总结	235
第六章 MySQL 数据库管理和数据库安全	237
6.1 MySQL 数据目录的维护和管理	237
6.1.1 数据目录的位置和结构	237
6.1.2 MySQL 的状态文件	239
6.2 MySQL 数据库的管理工作	243
6.2.1 数据库管理员应当熟悉的内容	243
6.2.2 数据库服务器的启动和关闭	243
6.2.3 mysqladmin 的使用	250
6.2.4 MySQL 的存取权限系统	252
6.2.5 数据库用户帐号的管理	259
6.2.6 数据库日志文件的维护	266
6.2.7 数据库的备份和拷贝	267
6.2.8 检查和维护数据库表	273
6.2.9 安排数据库的维护任务	274
6.3 性能的优化	274
6.3.1 服务器参数调节	275
6.3.2 重定位数据目录	282
6.3.3 配置和安装多个服务器	284
6.4 总结	286

第三篇 PHP & MySQL Web 编程

第七章 用 PHP 编写 CGI 程序	289
7.1 CGI 的基本知识	289
7.1.1 使用 CGI 的原因	289
7.1.2 什么是 CGI	289
7.1.3 CGI 的工作方式	291
7.2 用 PHP 编写 CGI 程序	306
7.2.1 PHP 的 CGI 工作方式环境设置	306

7.2.2 用 PHP 编写 CGI 程序实例	307
7.3 用 PHP 编写 CGI 程序的注意事项	315
7.3.1 URL 编解码问题	315
7.3.2 CGI 程序的安全性	316
7.3.3 CGI 程序的调试	316
7.3.4 CGI 程序和其他的应用程序	319
7.4 总结	319
第八章 用 PHP 操作 MySQL 数据库	321
8.1 开发网络应用的一般步骤	321
8.1.1 一般应用设计环节	321
8.1.2 数据和处理	322
8.2 MySQL 数据库的 PHP 接口函数	324
8.2.1 数据库服务器连接	325
8.2.2 数据库操作函数	326
8.2.3 从数据库中获取信息的操作函数	328
8.2.4 错误处理函数	331
8.3 phpMyAdmin 的使用说明	333
8.3.1 安装 phpMyAdmin 软件	333
8.3.2 使用 phpMyAdmin 常见的问题	337
8.3.3 phpMyAdmin 的设计分析	338
8.4 使用 MySQL 数据库的 PHP 程序实例	340
8.5 总结	347
第九章 用户管理程序	348
9.1 Apache 服务器上的用户身份认证	348
9.1.1 Apache 身份认证的基本原理	348
9.1.2 Apache 身份认证设置	349
9.2 用 PHP 实现身份认证	353
9.2.1 HTTP 认证原理	353
9.2.2 PHP 实现基本认证	353
9.2.3 PHP 结合 MySQL 数据库实现身份认证	355
9.3 用户管理	357
9.3.1 数据表格	358
9.3.2 系统基本函数	358
9.3.3 用户信息登记程序	362
9.3.4 用户登录程序	366
9.3.5 用户找回密码程序	368
9.4 总结	375

第十章 电子邮件处理程序	377
10.1 电子邮件基本知识	377
10.1.1 电子邮件协议	377
10.1.2 电子邮件系统工作原理	378
10.1.3 电子邮件的基本结构	379
10.2 用 PHP 编写发信程序	382
10.3 用 PHP 编写邮件处理程序	390
10.3.1 PHP 邮件处理程序原理	390
10.3.2 PHP 电子邮件操作函数	391
10.3.3 一个在线读信程序	392
10.4 在线信箱管理	408
10.5 总结	408
第十一章 建立聊天室	410
11.1 聊天室基本原理	410
11.2 聊天室的数据准备	411
11.3 聊天室程序设计	412
11.3.1 聊天室主页面设计	414
11.3.2 用户聊天信息记录程序	416
11.3.3 聊天信息显示程序	419
11.3.4 在线用户显示程序	421
11.4 总结	422
第十二章 投票统计程序	423
12.1 投票统计程序的原理和数据准备	423
12.2 投票程序	425
12.3 投票结果显示程序	428
12.4 总结	431
第十三章 创建讨论区	432
13.1 讨论区系统的原理和数据准备	432
13.2 发表文章处理程序	434
13.3 文章列表显示程序	436
13.4 文章详细信息查看程序	439
13.5 总结	442
第十四章 建设自己的网站	444
14.1 网站建设的一般步骤	444
14.2 会话管理	445
14.2.1 会话的概念	445

14.2.2	Session 使用实例	446
14.2.3	Session 和 Cookie 的区别和联系	451
14.3	管理并发	452
14.3.1	并发的产生	452
14.3.2	并发的处理	453
14.4	网站安全	460
14.4.1	PHP 程序安全问题描述	460
14.4.2	两种导致安全问题的方式	460
14.4.3	处理表单不合理数据	461
14.4.4	输入合理输出不合理	464
14.4.5	来自于内部的安全问题	466
14.5	总结	466
附录 A	PHP 相关参考	467
A.1	PHP 安装配置选项说明	467
A.2	PHP 语言中运算符说明	471
附录 B	MySQL 支持的 SQL 函数、工具和环境设置参考	474
B.1	MySQL 支持的 SQL 函数	474
B.1.1	用在 SELECT 和 WHERE 子句中的函数	474
B.1.2	用在 SELECT 和 WHERE 子句中的函数	477
B.1.3	在 GROUP BY 子句中使用的函数	488
B.2	MySQL 的标准控制选项和选项文件、环境变量	489
B.2.1	MySQL 支持的标准控制选项	489
B.2.2	MySQL 使用的选项文件	490
B.2.3	MySQL 使用的环境变量	490
B.3	数据库管理工具	491
B.3.1	mysqladmin 的使用	491
B.3.2	myisamchk 和 isamchk 的使用	492
B.3.3	myisampack 和 pack_isam 的使用	494
B.4	服务器启动工具	495
B.4.1	mysqld 的使用	495
B.4.2	safe_mysqld 的使用	496
B.4.3	mysql.server 的使用	497
B.5	数据备份和数据导入工具	497
B.5.1	mysqldump 的使用	497
B.5.2	mysqlimport 的使用	499
B.6	数据库访问工具	500
B.6.1	mysql 的使用	500
B.6.2	mysqlshow 的使用	502

B.7 测试和故障报告工具	502
附录 C PHP 和 MySQL 相关网络资源	504

第一篇

PHP 语言

本书的第一篇是 PHP 语言部分，详细介绍了 PHP 语言的基本知识，为读者建立了 Web 编程的基础。学习本篇内容只需要了解一些 HTML 语言的基本知识就可以了，如果你对 C 语言和 Perl 脚本比较熟悉，那么学习这一部分是比较简单的，它们之中有很多相同的地方，这是因为 PHP 从这两种语言中继承了很多东西。

本篇按照由浅入深的方式编写。没有任何 Web 编程经验的读者，可以按照本书的顺序，在一步步学习之后成为 Web 编程的高手。

本篇第一章讲述 PHP 的基本概念，读者可以从这里了解到什么是 PHP，它是怎么工作的等基础性知识。对于经验丰富的编程者，快速浏览这一章，大体了解一下就可以了。而对于 Web 开发的新手，笔者建议认真学习这一部分，逐步树立 Web 编程中的一些基本概念，以便在后面的编程实践中应用。

第二章中，讲解了如何建立 PHP 语言的工作环境。按照不同的工作平台，给出了不同的安装方案，供读者在建立运行环境时参考。

第三章 PHP 速成是本篇的核心部分。这一章从 PHP 最基本的语法开始讲起，介绍了 PHP 语言中的数据类型、运算符、表达式等基本语言要素。从流程控制开始讲解程序的构造方法，随后讲述 PHP 语言中的函数。函数是 PHP 语言的功能核心，本章详细说明了函数的声明、参数、调用方法，并介绍了大量的常用系统函数，这些函数是编程的功能基础。对于整个 Web I/O 系统的工作，在 3.6 节中进行了介绍，其中包括网络 I/O、文件系统工作、Cookie 的使用等内容。最后在 3.7 节中讲述了面向对象编程思想的一些基本概念，为读者进行建立基础，并说明了该思想在 PHP 语言中的实际应用。

第一章 PHP 语言概述

1.1 什么是 PHP

PHP(Hypertext Preprocessor: 超文本预处理程序)是一种服务器端 HTML - 嵌入式脚本描述语言。它用来在网站的服务器端快速有效地创建动态网页,是当今非常流行的一种服务器脚本语言,其执行效率比 ASP 或者传统的 CGI 要更高一些。

读者如果已经对 Cold Fusion (CFM)或者 Active Server Pages (ASP)有所了解,那么你一定知道服务器端 HTML-嵌入式脚本描述语言是怎么一回事了。如果你并不了解这是怎样的一个概念,那么可以理解为:这是一种编程的语言,它的语句写在 HTML 文件中,和标准的 HTML 语句共同存在;Web 服务器在调用这些文件时,能够利用包含在服务器端的 PHP 解释器模块动态地创建网页。举一个最简单的例子,你可以利用 PHP 语句让同一个网页在一天的不同时间显示不同的内容。内嵌有 PHP 语句的网页实际上是将标准的 HTML 文件进行了拓展,使其具有一些附加特征,Web 服务器在调用这些 PHP 文件的时候首先对它们进行解释,就像用 FrontPage 或者其他网页编辑工具创建和更新静态 HTML 网页一样,产生一个 HTML 文件。

1.2 PHP 语言的发展历史

PHP 的雏形产生于 1994 年秋天,它的创始人是 Rasmus Lerdorf,他最初设计 PHP 的目的是用来和访问他在线简历的人保持联系。1995 年初,用户可以使用的第一个 PHP 出现了,当时它的名字是 Personal Home Page Tools (PHP Tools),简称 PHP 1.0。那时作者只是把它当作一个简单的主页制作工具来使用,仅仅实现了访客留言簿和访客计数器等一些简单的功能。

1995 年年中, Rasmus 重新编写了整个 PHP 解释器,并把它命名为 PHP/FI 2,这里的 FI 指的是他编写的另外一个 HTML 表单处理软件包 Form Interpreter。在这个版本里面,他把个人主页制作工具和表单集成工具合并,同时还加入了对 mSQL 数据库的支持,这使 PHP/FI 在动态网页开发方面的能力大增,从而在社会上广为流传,为人们所熟识。到 1996 年底,大约有 15000 个 Web 站点使用了 PHP/FI 技术;而到 1997 年,使用 PHP/FI 的 Web 站点增长到 50000 多个。

1997 年,PHP 的发展到了一个新的阶段,一些程序员加入到 PHP 的开发设计中来,和 Rasmus 一起共同改进 PHP 技术。他们启动了第三版的开发计划,开发组中加入了 Zeev Suraski 和 Andi Gutmans 等人,而第三版就定名为 PHP3.0。他们重新改写了 PHP 的解释器,把大量的 PHP/FI 功能移植到 PHP3 中,修补了一些漏洞,并在功能方面作了一些改进,这奠定了 PHP3 的基础。此后,由于 PHP3 的源代码是完全开放的,不断有新的函数库添加到 PHP3 中。PHP3 在不停地更新,这使它无论在 UNIX 平台、Win32 系统还是新兴的 Linux 平台上都在不断地增加新的功能。

1998 年年中，市场上已经有大量的 PHP3 商用产品，例如 C2 公司的 StrongHold Web Server 和 RedHat Linux 中已经捆绑了 PHP/FI 解释器或 PHP3 解释器。根据 NetCraft 的保守估计，在那个时候，PHP 已经被世界上 150000 多个站点所使用。

现在，第四版 PHP 已经投入使用，它修正了 PHP3 的很多缺陷，使程序具有更快的执行速度，可以满足更高的要求。在修正之后，PHP4 在效率方面比传统的 CGI 或者 ASP 的优势更加明显，而且，它还有更强劲的新功能、更加丰富和不断更新的函数库。无论你是刚刚涉足 Web CGI 领域的新手还是专业的 Web 管理者，PHP 都很值得学习，它将成为你学习 Web 网络编程的必修课程之一。

1.3 PHP 的特点

一种语言之所以能够存在和发展，并具有强大的生命力，总是因为它有不同于（或者优于）其他语言的特点。PHP 语言的特点是：

(1) 语言简单、紧凑，使用灵活、方便。PHP 语言是 C、Java、Perl 等多种语言的结合，其中以 C 语言为主。作为一种语言，PHP 拥有丰富的运算符、运算类型，表达式类型多样化，灵活使用各种运算符可以实现在其他脚本语言中比较难以实现的运算。数据结构丰富，用户可以定义自己的类和对象，实现自己想要的数据结构和相应的操作函数。

(2) 语法限制不太严格，程序设计的自由度很大。例如，在 PHP 语言中，字符、字符串、整形数值和实型数值变量在使用时都无需事先进行类型声明，简化了变量的使用。PHP 语言允许程序编写者有较大的自由度，放宽了语法检查，正是因为这一点，PHP 语言才具有其他语言所不具有的灵活性。同时，这也需要程序员比较仔细地检查程序，保证其正确性。

(3) PHP 嵌入在 HTML 文档中。由于任何进行 Web 编程的人员都很熟悉 HTML 语言，所以学习使用 PHP 语言是一件很容易的事。

(4) 良好的可移植性。与微软公司提供的 ASP 不同，用 PHP 编写的程序不需要修改或者仅作一点点很少的修改就能够跨操作平台使用。这是因为 PHP 是一种可以运行在多种操作系统平台上的编程语言，它可以使用在 UNIX、Win 98/2000/NT、Macintosh 和 Linux 平台上。

(5) 运行效率高。一方面，PHP 是一种服务器端脚本语言，它是在服务器端运行的。它的程序可以编写得很大，结构可以很复杂，而这些不会对客户浏览器产生运行速度上的影响。另一方面，与其他的解释性脚本语言相比，由于 PHP 作为服务器的一个模块在服务器上运行，因此用 PHP 编写的程序运行速度更快，效率更高，且占用较少的系统资源。

(6) 强大的数据库支持和丰富的函数库。PHP 支持的数据库有：

Adabas D	DBA	DBase	dbm	filePro
Informix	InterBase	mSQL	Microsoft SQL Server	Solid
MySQL	Sybase	PostgreSQL	Oracle 8	Oracle

PHP 语言中有大量的和上述数据库的接口函数，可以方便地存取数据库中的数据。另外，PHP 还可以存取任何支持 ODBC 标准的数据。PHP 拥有丰富的函数库，可以灵活地支持网络输入/输出、处理数据、进行数学运算、分析和创建图像等等。随着 PHP 的广泛使用，它的

函数库将会得到迅速的扩充，能够提供更多的、功能更强的函数。

(7) PHP 的源代码是完全免费的。任何人都可以访问 PHP 的 Web 站点，下载完整的源程序代码和程序文档。它用于 Windows 操作平台的二进制代码也是免费的，可以从网上下载到。

(8) PHP 是开放的。用户可以按照自己的意愿编写程序，在 GNU 许可的条件下自由地发布，提供给其他人使用，同样可以自由地使用别人编写的程序。这样一来，PHP 的功能将会随着时间的推移而滚雪球式地强大起来。

以上简要介绍了 PHP 的一般特点，至于 PHP 语言的其他特点将结合在后面的章节中进行介绍。由于 PHP 语言具有这些其他语言所不具有的特点，因此 PHP 在当今的 Web 编程领域大受欢迎，人们开始热衷于使用 PHP 语言来开发他们的网络站点，这从另一方面又促进了 PHP 的进一步发展。

1.4 PHP 的工作机制

为了更好地理解 PHP 是怎样工作的，首先让我们来看一下 Web 网站是怎样进行工作的。

- (1) 用户在浏览器的地址栏中填上要访问的主页地址，按回车键触发浏览网页的申请。
- (2) 浏览器将申请发送到诸如 Apache、IIS 或者 PWS 等 Web 服务器上。
- (3) Web 服务器接收到这些申请，并根据.htm 或.html 后缀名识别到这是 HTML 文件。
- (4) Web 服务器从当前硬盘或内存中读取正确的 HTML 文件，然后将它送回用户浏览器。
- (5) 用户浏览器解释 HTML 文件，并将解释结果显示在用户浏览器上。

当然，这种浏览访问的过程可能会更加复杂一些，例如，Form 中的内容以及查询字串的传递等等。在大多数情况下，对于各种典型的 Web 服务器来说，这个过程是按部就班执行的，Web 服务器接收浏览文件申请并从内存或者硬盘中读取正确的文件，然后发送往用户浏览器。

PHP 改变了这一切，在 Web 服务器仍然支持静态 HTML 主页的前提下，利用 PHP 语言，可以根据用户的要求在 Web 服务器上动态地建立新的主页。

为了便于理解 PHP 页面和静态 HTML 页面的区别，我们将浏览的整个过程分成以下步骤：

- (1) 用户在浏览器的地址栏中填入 PHP 页面文件名称（文件名称以.php 为后缀），按回车键触发这个请求。
- (2) 浏览器将这个 PHP 页面请求发送给 Web 服务器。
- (3) Web 服务器接收到这个请求，并根据其文件名称后缀识别到这是个 PHP 请求。
- (4) Web 服务器从硬盘或者内存中取出正确的 PHP 文件。
- (5) Web 服务器将这个文件发送到 PHP 解释模块中进行解释。
- (6) PHP 解释模块对该文件从头到尾进行解释，并根据 PHP 语句的命令要求，产生相应的静态 HTML 文件。
- (7) Web 服务器将这个 HTML 页面发送给客户浏览器。
- (8) 用户浏览器解释 HTML 文件，并将解释结果显示在用户浏览器上。

【注意】

为了清晰地说明问题，以上步骤已经进行了很大的简化，一个 PHP 文件浏览请求并不一定每一次都需要重新解释，如果再次接收以前的那个请求而且没有任何变化，Web 服务器将会从 Cache(缓存)中提出结果而不是再次运行解释模块。