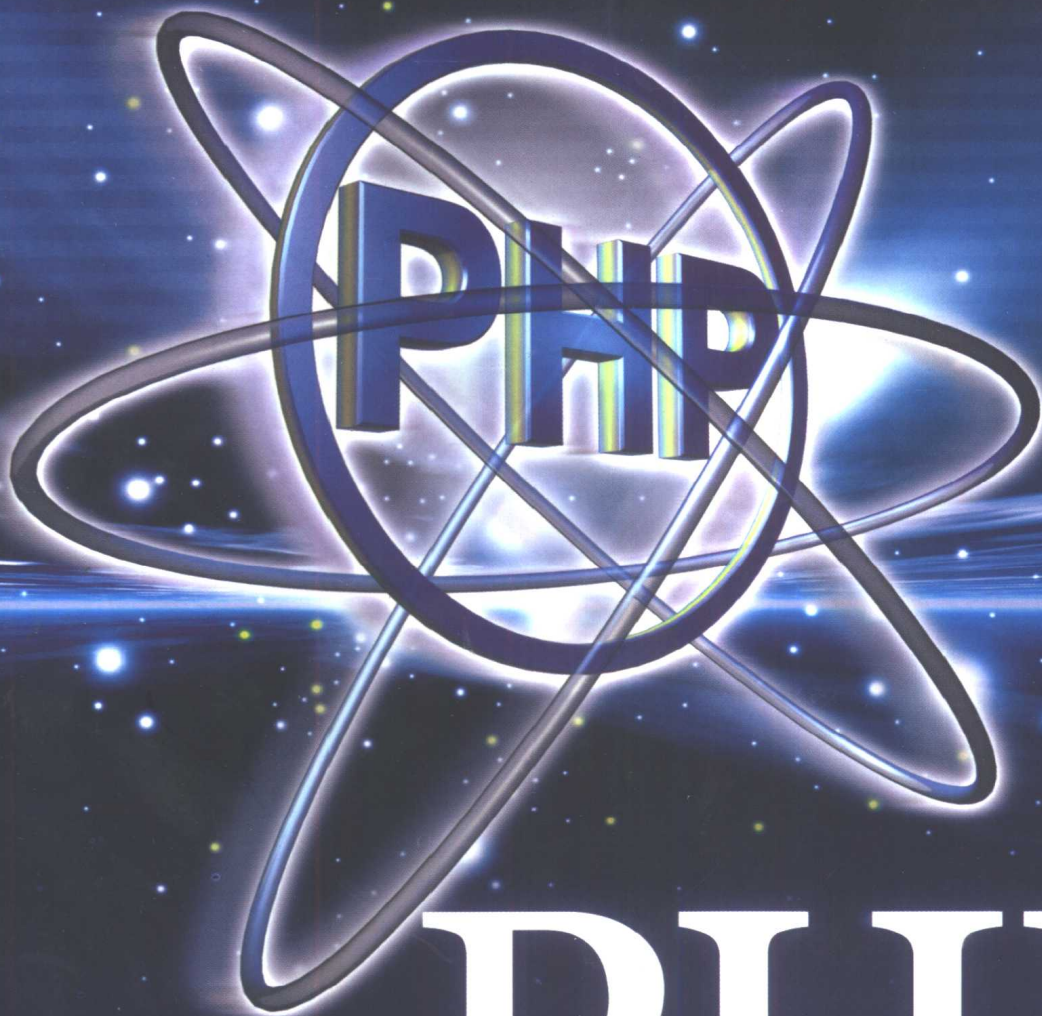




附光盘
CD-ROM

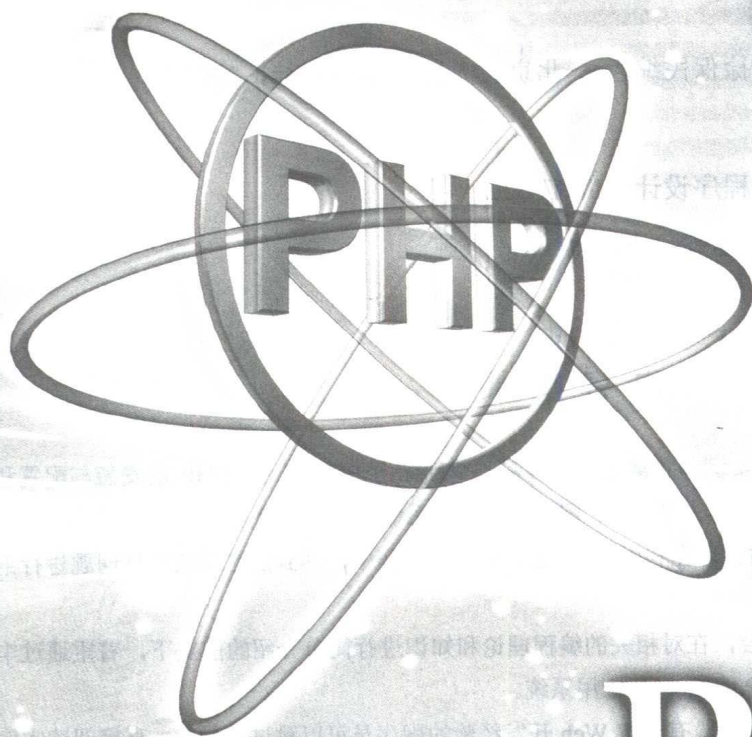


PHP

编程基础及 应用实例集锦

■ 徐保民 等 编著

人民邮电出版社
www.pptph.com.cn



PHP

编程基础及 应用实例集锦

■ 徐保民 等 编著

人民邮电出版社

图书在版编目 (CIP) 数据

PHP 编程基础及应用实例集锦/徐保民编著. —北京: 人民邮电出版社, 2001.5

ISBN 7-115-09291-5

I.P... II.徐... III.PHP 语言—程序设计 IV.TP312

中国版本图书馆 CIP 数据核字 (2001) 第 22906 号

内 容 提 要

全书共分 3 大部分: 基础篇、实例篇和常见问题解答。基础篇由 PHP 概述、PHP 的安装与配置和 PHP 语法 3 章构成, 实例篇由广告轮播、聊天室、留言板、投票系统和图书管理系统等 6 个精选的实例构成, 常见问题解答部分主要是针对读者在利用 PHP 进行编程的过程中, 遇到的这样或那样问题进行剖析和解答。

本书的特点是理论与实践相结合, 在对相关的编程理论和知识进行简单介绍的前提下, 着重通过丰富、实用的例子来说明如何利用 PHP 编制 Web 应用系统。

初学者可以从基础篇开始学习本书, 有一定 Web 开发经验的程序员可以跳过基础篇, 直接阅读实例篇和常见问题解答部分。

PHP 编程基础及应用实例集锦

◆ 编 著 徐保民 等

责任编辑 张立科

◆ 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>

读者热线: 010-67129212 010-67129211(传真)

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: 21.5

字数: 516 千字

2001 年 5 月第 1 版

印数: 1 - 6 000 册

2001 年 5 月北京第 1 次印刷

ISBN 7-115-09291-5/TP·2214

定价: 39.00 元(附光盘)

编者的话

PHP 是最流行的 Web 脚本语言之一，运行在 Web 服务器端，根据用户请求或服务端的数据产生动态网页。它嵌入在 HTML 文件中，功能强大，并内建了访问数据库的接口。它能够作为 Apache Web 服务器的模块执行，执行效率要高于普通的 CGI 程序。

内容和特点

本书由基础篇、实例篇以及常见问题解答 3 部分组成。基础篇由 PHP 概述、PHP 的安装与配置和 PHP 语法等 3 章构成。通过这几章的学习，使读者对 PHP 编程语言本身有一个比较全面的认识。实例篇由广告轮播、聊天室、留言板、投票系统和图书管理系统等 6 个精选的实例构成。这部分通过实例来讲述如何利用 PHP 开发 Web 应用程序和思考问题的方法。常见问题解答部分主要是针对人们在利用 PHP 进行编程的过程中，遇到的这样或那样问题进行剖析。

本书的特点是理论与实践相结合，在对相关的编程理论和知识进行简单介绍以后，通过丰富、实用的例子来说明如何利用 PHP 编制 Web 应用系统。

配套光盘内容简介

PHP 文件

书中讲述的各个实例的源文件。通过查看 Index.txt 文件，获取每个文件的简单说明和相应源代码文件名。

PHP 辅助软件

几个常用的 PHP 软件。通过查看 Index.txt 文件，可以获取每个文件的简单说明和相应文件名。

附录

包含 PHP 函数参考以及 MySQL 语言参考手册等内容。通过查看 Index.txt 文件，可以获取每个文件的简单说明和相应源代码文件名。

编者
2001 年 4 月

目 录

第 1 章	PHP 概述.....	1
1.1	PHP 的来历.....	1
1.2	功能特点.....	2
1.3	PHP 与其他 CGI 的比较.....	3
1.4	第一个 PHP 应用程序.....	4
第 2 章	PHP 的安装与配置.....	6
2.1	准备工作.....	6
2.2	系统安装.....	6
2.2.1	Linux 操作平台下 PHP 系统的安装流程.....	7
2.2.2	Windows 操作平台下 PHP 系统的安装流程.....	10
2.2.3	PHPLIB 的安装流程.....	13
2.3	PHP 的使用配置.....	13
2.3.1	PHP 的编译配置详细说明.....	14
2.3.2	PHP.INI 配置详细说明.....	18
第 3 章	PHP 的语法.....	25
3.1	数据类型.....	25
3.1.1	常量类型.....	25
3.1.2	变量类型.....	26
3.1.3	变量的使用.....	29
3.2	运算符.....	36
3.2.1	算术运算符.....	36
3.2.2	逻辑运算符.....	38
3.2.3	比较运算符.....	39
3.2.4	字符串运算符.....	39
3.2.5	位运算符.....	40
3.2.6	赋值运算符.....	41
3.2.7	其他运算符.....	41
3.3	PHP 表达式.....	42
3.4	控制语句.....	43
3.4.1	条件选择语句.....	43
3.4.2	循环控制语句.....	48

3.4.3 其他控制语句	50
3.5 函数与对象	53
3.5.1 函数	53
3.5.2 对象	57
3.6 其他	71
3.6.1 嵌入方法	71
3.6.2 文件包含	71
3.6.3 程序注释	72
第 4 章 PHP 的简单实例	73
4.1 用户认证	73
4.1.1 使用服务器内置功能进行认证	73
4.1.2 使用 PHP 进行认证	74
4.2 计数器	76
4.2.1 计数	77
4.2.2 计数的显示	80
4.2.3 用 session 实现的计数器	85
4.2.4 基于 PNG 格式的计数器	86
4.3 动态图形生成	87
4.3.1 线形图形	88
4.3.2 柱形图	94
4.3.3 饼形图	96
4.3.4 3D 饼形图	99
4.4 分页显示	103
4.5 邮件收取	106
4.6 新闻抓取	115
4.7 简单的搜索引擎	122
4.8 意见反馈表	124
第 5 章 应用实例——广告轮播	131
5.1 广告上传的实现	131
5.2 记录广告条点击次数的方法	137
5.3 一个完整的广告轮播实例	137
5.3.1 数据库的建立	138
5.3.2 通用函数的建立	139
5.3.3 广告管理模块的实现	141
5.3.4 广告点击模块的实现	150
5.3.5 广告显示模块的实现	151

第 6 章 应用实例二——聊天室	154
6.1 显示界面	154
6.2 用户管理	156
6.3 在线人数的实现	162
6.4 表情和动作的实现	163
6.5 密谈的实现	165
6.6 主动与被动更新	166
6.7 异常离线的处理方法	171
6.8 输入信息的发送	171
6.9 一个完整的聊天室实例	173
6.9.1 用户登录模块的实现	173
6.9.2 用户输入模块的处理	174
6.9.3 显示模块的实现	177
6.9.4 显示界面的实现	178
第 7 章 应用实例三——留言板	179
7.1 数据库的建立	179
7.2 通用函数的实现	180
7.3 主控模块的实现	180
7.4 留言模块的实现	182
7.5 搜索留言模块的实现	188
7.6 查看留言模块的实现	189
7.7 删除及修改留言模块的实现	199
7.8 回复留言模块的实现	205
第 8 章 应用实例四——投票系统	211
8.1 数据库的建立	211
8.2 通用函数的实现	212
8.3 在线申请投票模块的实现	216
8.3.1 产生投票页面	216
8.3.2 投票信息合法性检验	220
8.4 投票信息入库模块的实现	223
8.5 显示投票结果模块的实现	226
第 9 章 应用实例五——图书管理系统	229
9.1 数据库的建立	229
9.2 通用函数的建立	231
9.3 主控模块的实现	234
9.4 用户管理模块的实现	235

9.4.1	用户注册模块.....	235
9.4.2	用户登录模块.....	239
9.4.3	管理员登录模块.....	241
9.5	新书入库模块的实现.....	243
9.6	借阅模块的实现.....	247
9.6.1	借书登录模块.....	247
9.6.2	借书登记模块.....	248
9.6.3	还书处理模块.....	252
9.6.4	续借处理模块.....	253
9.6.5	借书登记处理模块.....	253
9.7	书籍查询模块的实现.....	255
附录	常见问题解答.....	263
A.1	数据库常见问题.....	263
A.1.1	MySQL 与 Oracle、Informix 有何差别.....	263
A.1.2	如何在 Windows 2000 环境下把 MySQL 安装成自动的服务器.....	263
A.1.3	如何在 MySQL 数据库中为用户设置密码.....	263
A.1.4	如何在 MySQL 中创建数据库和表.....	264
A.1.5	如何增大 MySQL 的最大连接数.....	264
A.1.6	如何获取到 MySQL 数据库中枚举列的全部可能值.....	264
A.1.7	如何对 MySQL 数据库中的日期数据进行比较.....	265
A.1.8	如何利用 MySQL 的函数实现 PHP 中时间的格式化输出.....	266
A.1.9	如何在 PHP 中实现基于数据库的用户权限 / 模块管理.....	267
A.1.10	如何利用 PHP 访问数据库.....	269
A.1.11	如何在 PHP 程序中通过 ODBC 调用数据库.....	271
A.1.12	如何用 PHP 调用数据库的存储过程.....	274
A.1.13	如何不使用 OCI 8 接口实现 PHP 和 Oracle 的连接.....	275
A.1.14	如何使用 ADO 访问数据库.....	276
A.1.15	如何在 PHP 中调用 JDBC.....	277
A.1.16	如何利用 PHP 操纵 LOB 类型的数据.....	278
A.2	编程常见问题.....	281
A.2.1	在 PHP 程序中, echo、print、printf 和 sprintf 之间有何区别.....	281
A.2.2	如何区分 PHP 的 require 和 include 语句的作用.....	281
A.2.3	如何把错误信息隐藏起来.....	282
A.2.4	如何巧妙地使用 PHP 的错误处理.....	283
A.2.5	如何在程序里获取 php.ini 文件的一个配置的值.....	284
A.2.6	如何在用户自定义函数中读取一个标准的 CGI 变量.....	284
A.2.7	如何用 PHP 生成自己的 LOG 文件.....	284
A.2.8	如何利用 PHP 实现目录管理与文件预览.....	286

A.2.9	如何在 PHP 中从一个页面重定向到另外一个页面	288
A.2.10	如何将 PHP 的结果输出到非 PHP 页面中	289
A.2.11	如何实现页面的自动跳转	290
A.2.12	如何利用 PHP 绘制一个可以更换文字的按钮	291
A.2.13	如何利用 PHP 创建菜单	292
A.2.14	如何利用 JavaScript 和 PHP 实现网页制作中双下拉菜单的动态生成 ..	294
A.2.15	如何使 PHP 能处理多个同名复选框	296
A.2.16	如何利用邮件来发送表单数据	297
A.2.17	如何才能删除数组的最后一个元素	299
A.2.18	如何对一段长的文字进行分解	299
A.2.19	如何利用 PHP 打开指定的网页	300
A.2.20	如何判断 PHP 中的变量是否已初始化	300
A.2.21	如何判断某个变量是否已定义	301
A.2.22	如何利用 getdate () 函数获取时间和日期	301
A.2.23	如何检测 Cookie 是否生效	302
A.2.24	如何读取用 Cookie 设置的变量值	303
A.2.25	如何实现跨域名的 Cookie	303
A.2.26	如何只让某个 IP 访问此页而其他人访问另一页	305
A.2.27	如何实现图像的缩放	305
A.2.28	如何使 PHP 4 调用自己编写的 COM 组件	306
A.2.29	如何将一个字符串转换成不超过特定长度的字符串	307
A.2.30	如何检测用 POST 和 GET 方法提交的表单中的变量	307
A.2.31	如何利用 PHP 生成动态 WAP 页面	309
A.2.32	如何用 PHP 实现对上传的 ZIP 文件的解压	310
A.2.33	如何实现文件上载	312
A.2.34	如何使用 PHP 建立 GIF	312
A.2.35	如何动态改变 PHP 的错误输出等级	313
A.2.36	如何实现 PHP 和 JavaScript 的混合编程	314
A.2.37	如何利用 PHP 对 XML 文件进行解析	315
A.2.38	如何在 PHP 中实现 ASP 的 Application 功能	318
A.2.39	如何利用 PHP 过滤换行符	320
A.2.40	如何利用 PHP 获取页面中的所有链接地址	320
A.2.41	如何使用 PHP 来给网页做导航条	321
A.3	其他问题	323
A.3.1	php_admin_value 和 php_value 之间有何异同	323
A.3.2	ASP 与 PHP 之间有何区别	323
A.3.3	PHP 与 PERL 之间有何区别	323
A.3.4	PHP 与 Cold Fusion 之间有何区别	323

A.3.5	如何直接读取请求的信息头.....	324
A.3.6	如何在 PHP 里启动一个进程以后让它在后台运行.....	324
A.3.7	如何在 PHP 里使用 SSI.....	324
A.3.8	如何才能在 Aache 中使用 PHP 4.....	324
A.3.9	如何才能在 IIS 中使用 PHP 4.....	325
A.3.10	如何才能同时使用 PHP 3 和 PHP 4.....	325
A.3.11	如何获知来访者的 URL 地址.....	325
A.3.12	如何将形如 XXX.YYY.WWW.ZZZ 的 IP 地址转变为整数.....	325
A.3.13	如何提高 PHP 程序的运行效率.....	326
A.3.14	如何在 PHP 中执行系统外部命令.....	326
A.3.15	如何在 Linux/UNIX 下定时运行 PHP 脚本.....	328
A.3.16	如何使用 PHPLIB 进行 Session 的管理和认证.....	328

第1章 PHP 概述

PHP 是一种面向 Internet/Intranet 的编程语言。它以一种 HTML 内嵌式语言的形式出现,可以在多种平台和多种 Web 服务器中使用。本章主要介绍 PHP 的来历以及功能特点,并将 PHP 和其他 CGI 语言进行比较,最后给出一个简单的 PHP 程序,使你对 PHP 有一个初步的认识。

1.1 PHP 的来历

PHP (Hypertext Preprocessor) 是一种服务器端 HTML 嵌入式脚本描述语言,类似 Windows NT 下的 IIS 上的 ASP,它的出现使得在 UNIX 上快速地开发动态网页成为现实。PHP 大量采用 C、Java 和 Perl 语言的语法,并加入了具有 PHP 自己特征的语法,使它可以比其他 CGI 或者 Perl 更快速地执行动态网页。

PHP 从 1994 年秋天开始孕育,创始人是 Rasmus Lerdorf。早期没有发布的版本用在 Rasmus Lerdorf 自己的网页上,跟踪有谁访问过他的在线个人简历。被其他人使用的第一个版本是在 1995 年发布的,当时叫做 Personal Home Page Tools。它包含了一个非常简单的语法分析引擎,只能理解一些指定的宏和一些 Home Page 后台的常见功能,如留言板、计数器等。在 1995 年中期,Rasmus Lerdorf 重新编写了这个语法分析引擎并且命名为 PHP/FI (Form Interpreter)。FI 来源于他写的另外一个可以接受 HTML 表单数据的程序包。他组合了 Personal Home Page Tools 脚本和 Form Interpreter,并且加入了对 mSQL 的支持,于是 PHP/FI2.0 诞生了,自此奠定了 PHP 在动态网页开发上的基础。此后 PHP/FI 便以一种令人惊奇的速度传播开来,人们开始大量使用它编写程序。到 1996 年底,世界上已经有至少 15 000 个站点在使用 PHP/FI,到 1997 年底,使用 PHP/FI 的 Web 网站已超过 50 000 个站点。

在 1997 年中,PHP 的发展发生了一个重大的转折。这就是 PHP 的开发从 Rasmus 个人的爱好升级到一群程序员有组织的工作。再后来,这个解析器被 Zeev Suraski 和 Andi Gutmans 重写,通过这次全面的重写,大量 PHP/FI 的功能被移植到 PHP 3 中,并且形成了 PHP 3 的雏形。

PHP 3 跟 Apache 服务器紧密结合、不断地更新及加入新的功能以及它基本上支持所有主流与非主流数据库的特点,再加上具有较高的运行效率,使得在 1999 年中 PHP 的使用站点超过了 150 000 个。

目前,PHP 的第 4 代 Zend 核心引擎已经进入测试阶段。整个脚本程序的核心大幅改动,使得程序的执行速度能够满足更高的要求。经过最佳化处理,其运行效率已比传统 CGI

或者 ASP 等程序有更好的表现。而且还提供了更强的新功能、更丰富的函数库支持。PHP 将在 Web CGI 的领域上，掀起新的革命。

1.2 功能特点

在服务器性能满足要求的情况下，PHP 语言的功能是非常强大的。具体讲，它具有如下功能特点：

1. 对数据库的广泛支持

通过 PHP 语言所提供的丰富数据库函数，可以方便地存取 Oracle、Sybase、MS SQL、Adabase D、MySQL、mSQL、PostgreSQL、dBase、FilePro、Unix dbm、Informix/Illustra 等类型的数据库，以及任何支持 ODBC 标准的数据库，使得用 PHP 编写基于数据库的网页变得非常简单。这是它迅速走红的原因之一。

2. 对网络通信协议的广泛支持

PHP 所支持的网络通信协议包括：与电子邮件相关的 IMAP 协议和 POP3 协议、网管系统 SNMP 协议、网络新闻 NNTP 协议、账号共用 NIS 协议、全球信息网 HTTP 协议以及 Apache 服务器和目录协议 LDAP 等。

3. 良好的移植性

用 PHP 开发的 Web 后端 CGI 程序，可以很容易地移植到不同操作平台上。例如，起初在 Linux 环境下建立的网站，在系统负荷过高时，可以快速地将整个系统移到 SUN 工作站上，不用重新编译 CGI 程序。面对快速发展的 Internet，这是长期规划的最好选择。

4. 良好的开放性和可扩展性

所有的 PHP 源码及文档都可以免费复制、编译、打印和分发。用户用 PHP 编写的任何程序都属于用户自己，可以按照自己的意愿进行处理，而不需要付任何版税。就 PHP 本身而言，用户可以无限制地发布自己编写的程序。

5. 运行效率高

与其他的解释性语言相比，PHP 系统消耗较少的系统资源。特别是当 PHP 作为 Apache Web 服务器的一部分时，运行代码不需要调用外部二进制程序，服务器解释脚本也不需要承担任何额外负担。

6. 基于服务器端

由于 PHP 是在 Web 服务器端运行的，因此 PHP 不能直接与运行在客户端的 Web 浏览器中的 Java applets、ActiveX 或 Javascript 进行比较。然而可以很容易地将这些语言与 PHP 相结合在一起。使用 PHP 可以很方便地生成所需要的任何 HTML 代码，当然也能激活 Java applets 和 ActiveX 控件，还可以动态生成 JavaScript 语句。

7. 强大的字符串处理以及文档处理能力

PHP 语言对数学运算、时间处理、文件系统、字符串处理等功能提供了大量的函数，供程序员调用。再加上它是免费的系统，使得效益与成本比，近乎等于无限大。

1.3 PHP 与其他 CGI 的比较

当今，制作网页的动态语言也越来越多，主要流行的有以下几种：ASP、PHP、JSP 等，它们之间互有优缺点，如表 1-1 所示。

表 1-1 PHP、ASP 及 JSP 的比较

程序界面	PHP	ASP	JSP
操作系统	均可	Win32	均可
Web 服务器	多种	IIS	多种
执行效率	快	快	快
稳定性	佳	中等	一般
开发时间	短	短	较短
修改时间	短	短	较短
程序语言	PHP	VB	Java
网页结合	佳	佳	差
学习门槛	低	低	较低
函数支持	多	少	不定
系统安全	佳	极差	佳
使用网站	极多	多	多

ASP 是基于 Web 的一种编程技术，可以说是 CGI 的一种。它可以完成以往 CGI 程序的所有功能，如计数器、留言簿、公告板、聊天室等等。

ASP 操作数据库的功能非常强大。如果使用基于 NT 的 Internet 建立 MIS 应用，采用 ASP 技术应该是首选方案。用 ASP 编写出的系统，完全可以达到原来使用 Client/Server 结构的网络数据库系统的水平。使用 ASP 最大的好处是：在对程序进行升级时，只需要修改服务器上的 ASP 文件，而不需要修改客户端程序。

另外，ASP 可以轻松地实现对页面内容的动态控制，根据不同的浏览者，显示不同的页面内容。而浏览者一点觉察不出来，就像为他自己专门制作的页面一样。

JSP (Java Server Page) 是 Sun 公司推出的新一代站点开发语言，它完全解决了目前 ASP 和 PHP 的一个通病，即脚本级执行。JSP 可以在 Servlet 和 JavaBean 的支持下，实现功能强大的站点程序。

PHP 则秉承 Linux 的 GNU 风格，借助于源码公开，它迅速成为世界上目前应用最为广泛的站点制作语言之一。借助于 C++ 的语法形式，引用类的概念，使得代码的可重复性应用变得异常简单。加上它与 Linux、Apache 和 MySql 的紧密配合，关键性的应用也没有问题，例如有名的 Sina 就采用 PHP。由于没有一个非常有实力的公司专门给它做支持，因此相关的学习资料比较少，目前在国内使用它的站点比较多，但是专门讲解它的站点就比较少，这给初学者带来了很多的麻烦。

1.4 第一个 PHP 应用程序

为了对 PHP 有一个初步的认识,下面来看一个简单的用 PHP 编写的“hello, world”程序。

1. <html>
2. <head>
3. <title>第一个 PHP 程序</title>
4. </head>
5. <body>
6. <?php
7. echo "hello, world\n";
8. ?>
9. </body>
10. </html>

上述 10 行程序在 PHP 中不需经过编译等复杂的过程,只要将它放在可执行 PHP 语法的服务器中,将它存成文件 `helloworld.php` 即可。在用户的浏览器端,只要在地址(Location)中输入 `http://localhost/helloworld.php`,就可以在浏览器上看到“hello, world”字符串,如图 1-1 所示。



图 1-1 “hello,world”程序的运行结果

可以看到,这个程序中起显示作用的只有 3 行,其他 7 行都是标准的 HTML 语法。而它在返回浏览器时和 JavaScript 或 VBScript 完全不一样,PHP 的程序没有被传到浏览器,只是在浏览器上显示短短的几个字“hello, world”。

第 6 行及第 8 行,分别是 PHP 的开始及结束的嵌入符号。第 7 行才是服务器端执行的

程序行。在这个例子中，“\n”与在 C 语言中的含义一样，代表换行的意思。这正好表明 PHP 是一个混合多种语言而成的编程语言，而 C 语言正是其含量最多的语言。在一个表达式结束后，要加上分号代表结束。

为了使你对 PHP 程序的执行过程有一个更深入的理解，下面就“hello, world”程序的执行过程作一个详细的介绍。

- (1) 用户在浏览器的地址栏中输入要访问的 PHP 文件名，即 `http://localhost/helloworld.php`，然后按回车键，触发这个 PHP 文件请求。
- (2) 浏览器将这个 PHP 请求传送给支持 PHP 的 Web 服务器。
- (3) Web 服务器接受这个请求，并从内存或硬盘中取出用户要访问的 PHP 文件，即 `helloworld.php`。
- (4) Web 服务器将 `helloworld.php` 文件发送给 PHP 解释程序。
- (5) PHP 解释程序对 `helloworld.php` 文件从头到尾进行扫描分析，并生成相应的静态 HTML 页面。
- (6) Web 服务器将生成的 HTML 文件回送到用户的浏览器。
- (7) 用户所在浏览器对该 HTML 文件进行解释执行，并将执行结果显示在屏幕上。

第 2 章 PHP 的安装与配置

本章首先介绍 PHP 在 Linux 和 Windows 操作系统下的安装流程，然后介绍如何配置 PHP，以便使它更好地工作。

2.1 准备工作

在安装 PHP 作为 Web 服务器的一部分时，既可以用 UNIX (Linux) 操作系统，也可以用 Windows NT/95 等 Win32 API 平台。当然，大部分的人都会选取 UNIX 作为 PHP 的执行平台，而 Windows NT 的用户大多数都会选择 IIS + ASP。实际上，Linux + Apache + PHP 应是最经济的选择，因为这样的组合几乎是不花钱的，成本与效益比也是最好的选择。而许多成功网站的经验，更是采用这种组合最好的佐证。

使用 Linux 操作系统，可以选择不同的 Linux 套件，包括 Slackware Linux、RedHat、OpenLinux、SuSE 等，这方面的软件是很容易而且很便宜就可以买到。当然，也可以去各大 FTP 站点下载免费的、完整的安装系统。

Apache 服务器是目前大多数 Web 网站所采用的服务器。可以到 <http://www.apache.org> 下载最新版的程序及相关文件。

PHP 则可以去它的官方网站 <http://www.php.net> 下载所需要的程序。

虽然目前 Windows NT 或者 Windows 98 等 Win32 系统平台也能安装 PHP 及 Apache 服务器，但在 UNIX 下，PHP 和 Apache 可以跑得更快、更好。

当然，若想使用商业化的系统平台，SUN、IBM、HP、DEC、SGI、NEC 等公司都提供有相关的 UNIX 或者是 Windows NT 的系统平台，加上具有高安全性的 Apache 服务器 Stronghold，或是其他支持 SSL 的 Apache 版本。这种组合，能满足大多数商业化的需求。而 PHP 就扮演着快速方便的 CGI 角色，让客户对网站的服务品质更加满意。

2.2 系统安装

尽管操作平台、Web 服务器，以及后台数据库的组合有许多种，但是系统安装过程大同小异。下面主要介绍 Linux 和 Windows 操作平台下 PHP 系统的安装流程，以及 PHPLIB 的安装流程。

2.2.1 Linux 操作平台下 PHP 系统的安装流程

下面以 Redhat Linux 6.2+Oracle 8.1.6i+ Apache 1.3.14+PHP 4.0.3 为例,说明 Linux 操作平台下 PHP 系统的安装流程。

其安装过程如下:

1. 安装 Oracle 前的准备

(1) 安装 JRE1.3.0

- [root@wapdb /root]# #以 root 身份登录
- [root@wapdb /root]# chmod 755 j2re-1_3_0-linux-rpm.sh
#改变 j2re-1_3_0-linux-rpm.sh 的权限
- [root@wapdb /root]# ./ j2re-1_3_0-linux-rpm.sh #运行 j2re-1_3_0-linux-rpm.sh
在同级目录下可得到 JRE 的 RPM 包
- [root@wapdb /root]# rpm -i j2re-1_3_0-linux-rpm #安装 JRE1.3.0

(2) 创建 Oracle、Apache 用户组

- [root@wapdb /root]# #以 root 方式登录
- [root@wapdb /root]# groupadd oracle #创建 oracle 组
- [root@wapdb /root]# useradd oracle -g oracle #在 oracle 组中建新用户 oracle
- [root@wapdb /root]# useradd apache -g oracle #在 oracle 组中建新用户 apache
- [root@wapdb /root]# passwd oracle #修改 oracle 用户口令

注意:

创建 Oracle 组和 oracle 用户是安装 Oracle 所必需的。创建 Oracle 用户后, Linux 系统会自动在/home 目录下创建/home/oracle 目录,供 Oracle 用户使用,Oracle 用户对此目录拥有所有的权限。

(3) 编辑.bash_profile 文件

- [root@wapdb /root]# cd /home/oracle #进入目录
- [root@wapdb /root]# vi .bash_profile #打开文件
- 在文件.bash_profile 内,输入以下内容:

```
# Setup ORACLE environment #设置 Oracle 运行环境
NLS_LANG="SIMPLIFIED CHINESE_CHINA.ZHS16CGB231280";export NLS_LANG
JAVA_HOME=/usr/java/jre1.3
BASH_ENV=$HOME/.bashrc
ORACLE_HOME=/data/oracle/app/oracle/product/8.1.6
export ORACLE_HOME
ORACLE_SID=wap
export ORACLE_SID
# Set up the search paths #设置搜索路径
PATH=$PATH: $HOME/bin: $JAVA_HOME/bin:
$ORACLE_HOME: $ORACLE_HOME/bin: $ORACLE_HOME/dbs
export USERNAME BASH_ENV PATH ORACLE_HOME JAVA_HOME
```