

Advanced PHP
For Web Professionals



Web 专家 PHP 高级编程

- 快速：构建复杂的、PHP 驱动的 Web 站点
- 便捷：探寻功能强大的 PHP 新技术
- 学习：基于 PHP-GTK 和 PEAR::DB 的全新技术
- 精通：解析 XML 文档、用户认证、表单处理、数据挖掘等等

Christopher Cosentino 著
崔玮 译



清华大学出版社

Web 专家 PHP 高级编程

Advanced PHP For Web Professionals

Christopher Cosentino 著

崔 玮 译



清华大学出版社

北京

Simplified Chinese edition copyright © 2003 by PEARSON EDUCATION ASIA LIMITED and TSINGHUA UNIVERSITY PRESS.

Original English language title from Proprietor's edition of the Work.

Original English language title: Advanced PHP For Web Professionals by Christopher Cosentino, Copyright © 2003

EISBN: 0-13-008539-1

All Rights Reserved.

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

This edition is authorized for sale only in the People's Republic of China (excluding the Special Administrative Region of Hong Kong and Macau).

本书中文简体翻译版由 Pearson Education 培生教育出版亚洲有限公司授权给清华大学出版社在中国境内（不包括中国香港、澳门特别行政区）出版发行。

北京市版权局著作权合同登记号 图字 01-2003-6901 号

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。

图书在版编目（CIP）数据

Web 专家 PHP 高级编程/康西蒂诺（Cosention, C.）著；崔玮译.

—北京：清华大学出版社，2003.12

书名原文：Advanced PHP For Web Professionals

ISBN 7-302-07654-5

I. W… II. ①康… ②崔… III. PHP 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字（2003）第 106646 号

出 版 者：清华大学出版社

地 址：北京清华大学学研大厦

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

邮 编：100084

社总机：(010) 62770175

客户服务：(010) 62776969

组稿编辑：成昊

文稿编辑：洪英

封面设计：杨月静

版式设计：关静

印 刷 者：北京市耀华印刷有限公司

发 行 者：新华书店总店北京发行所

开 本：异 16 印张：19 字数：403 千字

版 次：2003 年 12 月第 1 版 2003 年 12 月第 1 次印刷

书 号：ISBN 7-302-07654-5/TP · 5615

印 数：1 ~ 4000

定 价：32.00 元

内 容 提 要

PHP 是一种 HTML 嵌入式脚本语言，具有数据库访问速度快、运行效率高、性能稳定等特点，是开发 Web 应用程序的有力工具。

本书共 12 章，主要介绍 PHP 高级编程知识，涉及会话过程、数据库交互、文件处理、调试与排错、XML、图像处理、PHPGTK+ 等内容。书中利用大量的实例阐述 PHP 高级使用技巧，所有实例都有详尽的分析和注释。

本书实例丰富、实用性强，许多经验可以直接应用于网站建设，适于广大 Web 开发人员和 PHP 编程爱好者阅读参考。

前 言

本书目的

本书的目的是帮助读者更深一步掌握 PHP，学习一些不常用的功能，这些功能对创建应用程序很有帮助。希望本书所介绍的方法能够帮助读者轻松地编写属于自己的更容易使用的应用程序。

读者对象

本书比较适合那些已经掌握了一些 PHP 基础知识，并且希望了解更多高级技术的读者。我曾经收到过许多读者的邮件，都是关于我的第一本书——*Essential PHP for Web Professionals*（也是由 Prentice Hall 出版），他们非常喜欢书中简明的风格和丰富的例子。很多读者说他们还阅读过其他 PHP 编程图书，但直到读过我的第一本书，才真正掌握了 PHP 的工作过程和使用方法。他们的这种感觉让我无比欣慰。希望这本书也能让你有同样的感觉。

本书是专为那些对 PHP 有所了解，准备进一步学习的人员所写。本书不是一本参考书，也没有覆盖 PHP 高级概念的所有细节。但它确实已经超出了基础范围，介绍了很多帮助学习 PHP 编程的概念知识。

取得代码

本书所有代码可以从页面 www.phptr.com/advancedweb/php 上找到。这个页面还有更新或勘误表，以及常见问题的解答。如有问题，请发邮件至：cosentino@digital-salvage.net。

条款

以下内容摘自 GNU 网站 <http://www.gnu.org>。

本书的程序都是自由软件；在遵守自由软件基金会的 GNU 通用公共许可证——第 2 版或更新版本条款的前提下，可以发布和/或修改程序。

有关复制、发布和修改的条款和条件

0. 此许可证适用于任何包含版权所有者的声明的程序和其他作品，版权所有者在声明中明确说明程序和作品可以在 GPL 条款的约束下发布。下面提到的“程序”指的是任何这样的程序或作品。而“基于程序的作品”指的是程序或者任何受版权法约束的衍生作品。也就是说，包含程序或程序的一部分的作品，可以是原封不动的，或经过修改的和/或翻译成其他语言的（程序）。（在下文中，翻译包含在修改的条款中。）每个许可证接受人（licensee）用“你”来称呼。

许可证条款不适用于复制、发布和修改以外的活动。这些活动超出这些条款的范围。运行程序的活动不受条款的限止。仅当程序的输出构成基于程序作品的内容时，这一条款才适用（如果只运行程序就无关）。是否普遍适用取决于程序具体用来做什么。

1. 只要你在每一副本上明显和恰当地出版版权声明和不承担担保的声明，保持此许可证的声明和没有担保的声明完整无损，并和程序一起给每个其他的程序接受者一份许可证的副本，你就可以用任何媒体复制和发布你收到的原始的程序的源代码。

你可以为转让副本的实际行动收取一定费用。你也有权选择提供担保以换取一定费用。

2. 你可以修改程序的一个或几个副本或程序的任何部分，以此形成基于程序的作品。只要你同时满足下面的所有条件，你就可以按前面第一款的要求复制和发布这一经过修改的程序或作品。

a) 你必须在修改的文件中附有明确的说明：你修改了这一文件及具体的修改日期。

b) 你必须使你发布或出版的作品（它包含程序的全部或一部分，或包含由程序的全部或部分衍生的作品）允许第三方作为整体按许可证条款免费使用。

c) 如果修改的程序在运行时以交互方式读取命令，你必须使它在开始进入常规的交互使用方式时打印或显示声明，包括适当的版权声明和没有担保的声明（或者你提供担保的声明）；用户可以按此许可证条款重新发布程序的说明；并告诉用户如何看到这一许可证的副本。（例外的情况：如果原始程序以交互方式工作，它并不打印这样的声明，你的基于程序的作品也就不需要打印声明）。

这些要求适用于修改了的作品的整体。如果能够确定作品的一部分并非程序的衍生产品，可以合理地认为这部分是独立的，是不同的作品。当你将它作为独立作品发布时，它

不受此许可证和它的条款的约束。但是当你将这部分作为基于程序的作品的一部分发布时，作为整体它将受到许可证条款约束。准予其他许可证持有人的使用范围扩大到整个产品。也就是每个部分，不管它是谁写的。

因此，本条款的意图不在于索取权利；或剥夺全部由你写成的作品的权利。而是履行权利来控制基于程序的集体作品或衍生作品的发布。

此外，将与程序无关的作品和该程序或基于程序的作品一起放在存储体或发布媒体的同一卷上，并不导致将其他作品置于此许可证的约束范围之内。

3. 你可以以目标码或可执行形式复制或发布程序（或符合第 2 款的基于程序的作品），只要你遵守前面的第 1、2 款，并同时满足下列 3 条中的 1 条。

a) 在通常用作软件交换的媒体上，和目标码一起附有机可读的完整的源码。这些源码的发布应符合上面第 1、2 款的要求。或者

b) 在通常用作软件交换的媒体上，和目标码一起附有给第三方提供相应的机器可读的源码的书面报价。有效期不少于 3 年，费用不超过实际完成源程序发布的实际成本。源码的发布应符合上面的第 1、2 款的要求。或者

c) 和目标码一起，附有你收到的发布源码的报价信息（这一条款只适用于非商业性发布，而且你只收到程序的目标码或可执行代码和按 b) 款要求提供的报价）。

作品的源码指的是对作品进行修改最优先择取的形式。对可执行的作品讲，完整的源码包括：所有模块的所有源程序，加上有关的接口的定义，加上控制可执行作品的安装和编译的 script。作为特殊例外，发布的源码不必包含任何常规发布的供可执行代码在上面运行的操作系统的主要组成部分（如编译程序、内核等）。除非这些组成部分和可执行作品结合在一起。

如果采用提供对指定地点的访问和复制的方式发布可执行码或目标码，那么，提供对同一地点的访问和复制源码可以算作源码的发布，即使第三方不强求与目标码一起复制源码。

4. 除非你明确按许可证提出的要求去做，否则你不能复制、修改、转发许可证和发布程序。任何试图用其他方式复制、修改、转发许可证和发布程序是无效的，而且将自动结束许可证赋予你的权利。然而，对那些从你那里按许可证条款得到副本和权利的人们，只要他们继续全面履行条款，许可证赋予他们的权利仍然有效。

5. 你没有在许可证上签字，因而你没有必要一定接受这一许可证。然而，没有任何其他东西赋予你修改和发布程序及其衍生作品的权利。如果你不接受许可证，这些行为是法律禁止的。因此，如果你修改或发布程序（或任何基于程序的作品），就表明你接受这一许可证以及它的所有有关复制、发布和修改程序或基于程序的作品条款和条件。

6. 每当你重新发布程序（或任何基于程序的作品）时，接受者自动从原始许可证颁发者那里接到受这些条款和条件支配的复制、发布或修改程序的许可证。你不可以对接受者

履行这里赋予他们的权利强加其他限制。你也没有强求第三方履行许可证条款的义务。

7. 如果由于法院判决或违反专利的指控或任何其他原因（不限于专利问题）的结果，强加于你的条件（不管是法院判决、协议或其他）和许可证的条件有冲突。他们也不能用许可证条款为你开脱。

在你不能同时满足本许可证规定的义务及其他相关的义务时，作为结果，你可以根本不发布程序。例如，如果某一专利许可证不允许所有那些直接或间接从你那里接受副本的人们在不付专利费的情况下重新发布程序，惟一能同时满足两方面要求的办法是停止发布程序。

如果本条款的任何部分在特定的环境下无效或无法实施，就使用条款的其余部分，并将条款作为整体用于其他环境。

本条款的目的不在于引诱你侵犯专利或其他财产权的要求，或争论这种要求的有效性。本条款的主要目的在于保护自由软件发布系统的完整性。它是通过通用公共许可证的应用来实现的。许多人坚持应用这一系统，已经为通过这一系统发布大量自由软件作出慷慨的奉献。作者/捐献者有权决定他/她是否通过任何其他系统发布软件。许可证持有人不能强制这种选择。

本节的目的在于明确说明许可证其余部分可能产生的结果。

8. 如果由于专利或者由于有版权的接口问题使程序在某些国家的发布和使用受到限制，将此程序置于许可证约束下的原始版权拥有者可以增加限制发布地区的条款，将这些国家明确排除在外。并在这些国家以外的地区发布程序。在这种情况下，许可证包含的限制条款和许可证正文一样有效。

9. 自由软件基金会可能随时出版通用公共许可证的修改版或新版。新版和当前的版本在原则上保持一致，但在提到新问题时或有关事项时，在细节上可能出现差别。

每一版本都有不同的版本号。如果程序指定适用于它的许可证版本号以及“任何更新的版本”，你有权选择遵循指定的版本或自由软件基金会以后出版的新版本。如果程序未指定许可证版本，你可选择自由软件基金会已经出版的任何版本。

10. 如果你愿意将程序的一部分结合到其他自由程序中，而它们的发布条件不同，写信给作者，要求准予使用。如果是自由软件基金会加以版权保护的软件，写信给自由软件基金会。我们有时会作为例外的情况处理。我们的决定受两个主要目标的指导。这两个主要目标是：我们的自由软件的衍生作品继续保持自由状态，以及从整体上促进软件的共享和重复利用。

没有担保

11. 由于程序准予免费使用，在适用法准许的范围内，对程序没有担保。除非另有书面说明，版权所有者和/或其他提供程序的人们“一样”不提供任何类型的担保，不论是明确的还是隐含的，包括但不限于隐含的适销和适合特定用途的保证。全部的风险，如程序的质量和性能问题都由你来承担。如果程序出现缺陷，你承担所有必要的服务、修复和改正的费用。

12. 除非适用法或书面协议的要求，在任何情况下，任何版权所有者或任何按许可证条款修改和发布程序的人们都不对你的损失负有任何责任。包括由于使用或不能使用程序引起的任何一般的、特殊的、偶然发生的或重大的损失（包括但不限于数据的损失，或者数据变得不精确，或者你或第三方的持续的损失，或者程序不能和其他程序协调运行等）。即使版权所有者和其他人提到这种损失的可能性也不例外。

目 录

第 1 章 PHP 概述	1
1.1 概述.....	1
1.2 PHP 句法.....	2
1.3 变量.....	2
1.4 运算符.....	4
1.4.1 算术运算符.....	4
1.4.2 赋值运算符.....	4
1.4.3 关系运算符.....	4
1.4.4 拼接字符串运算符.....	5
1.5 数组.....	5
1.6 if/then 语句.....	6
1.7 switch 语句.....	7
1.8 for 循环.....	8
1.9 foreach 循环.....	8
1.10 while 循环.....	9
1.11 do while 循环.....	9
1.12 用户定义函数.....	10
1.13 PHP 的面向对象程序设计.....	10
1.14 phpinfo().....	12
1.15 补充资料.....	12
第 2 章 会话管理	13
2.1 PHP4 会话管理介绍.....	13
2.2 在 php.ini 里初始设置会话管理.....	14
2.2.1 在 Windows 平台上设置 php.ini.....	14
2.2.2 在 Linux 平台上设置 php.ini.....	14
2.2.3 一般问题.....	14
2.3 启动会话.....	15

2.4	会话文件的内容	16
2.5	会话期间跨页面跟踪变量	17
2.6	注销会话变量	21
2.7	结束会话	22
2.8	一个简单的基于会话的购物车	24
第 3 章	多种数据库交互	31
3.1	PHP4 与高级数据库的交互	31
3.2	PHP 的特定数据库函数	32
3.2.1	连接数据库服务器	33
3.2.2	选择正确的数据库	34
3.2.3	数据库查询、插入、读取或删除数据	34
3.2.4	获得查询结果并返回给用户	34
3.3	多种数据库的 DBX-PHP 支持	35
3.3.1	在 Linux 下打开 DBX 功能	35
3.3.2	在 Windows 下打开 DBX 功能	36
3.3.3	DBX 函数	36
3.3.4	使用 DBX	40
3.4	创建支持多种数据库的应用程序	44
3.5	使用 PEAR::DB	58
3.5.1	PEAR——PHP 扩展与应用储存库	59
3.5.2	下载 PEAR	59
3.5.3	从并发版本系统 (CVS) 下载 PEAR	59
3.5.4	安装 PEAR	60
3.5.5	基础应用	60
第 4 章	改善表单处理	67
4.1	改善表单介绍	67
4.2	post 方法和 get 方法	68
4.3	改进脚本程序	72
4.3.1	检查空格	72
4.3.2	检查最少字符数	73
4.3.3	使用更多的错误描述信息	73
4.4	表单域的检验	74

4.5 使用\$HTTP_GET_VARS 和\$HTTP_POST_VARS	78
4.6 完整的表单处理	80
4.7 动态变量	85
第 5 章 实例：简单的购物车系统	90
5.1 简介	90
5.2 购物车的功能	90
5.2.1 简单的购物界面	90
5.2.2 简单的执行和管理界面	92
5.3 搭建购物车系统	94
5.4 代码	94
5.4.1 cart.sql	95
5.4.2 cart.php	96
5.4.3 cart_inc.php	99
5.4.4 dbmanage.php	108
5.4.5 admin.php	118
5.4.6 checkout.php	125
5.4.7 finishorder.php	131
第 6 章 文件	135
6.1 概述	135
6.2 打开和关闭文件	135
6.2.1 打开文件	135
6.2.2 读取文件	136
6.2.3 关闭文件	137
6.3 文件操作应用	137
6.4 数据采集	141
6.5 解析大型文本文件	144
第 7 章 PHP 的认证方式	157
7.1 用户认证概述	157
7.2 生成口令	158
7.3 根据文本文件认证用户	160
7.4 根据 IP 地址认证用户	165
7.5 利用 HTTP 认证方法认证用户	167

7.6 通过数据库查询认证用户	171
第 8 章 错误管理	175
8.1 概述	175
8.2 错误报告	176
8.2.1 PHP 中的错误类型	176
8.2.2 php.ini 中的错误报告设置	177
8.2.3 error_reporting()	177
8.2.4 php.ini 设置: display_errors	178
8.2.5 为开发和测试设置 php.ini	178
8.2.6 错误日志	180
8.2.7 自定义错误处理函数	181
8.2.8 触发错误	184
8.2.9 跟踪用户登录	187
第 9 章 实例: 一个会议记录的应用系统	194
9.1 简介	194
9.2 构建会议管理应用系统	195
9.3 代码	196
9.3.1 meeting.sql	196
9.3.2 meeting.php	197
9.3.3 meeting_inc.php	215
9.3.4 admin.php	218
第 10 章 PHP 与 XML	223
10.1 概述	223
10.2 创建一个 XML 解析器	224
10.2.1 定义 XML 解析器	225
10.2.2 定义元素处理程序	225
10.2.3 定义字符处理程序	225
10.2.4 启动解析器	226
10.2.5 清除	226
10.3 解析并转换成 XML 文档	229
10.4 导出数据库内容到一个 XML 文件	241

第 11 章 用 PHP 生成动态图像	245
11.1 概述	245
11.2 GD 库	245
11.3 打开 GD 支持	246
11.3.1 Windows 下打开 GD 支持	246
11.3.2 Linux 下打开 GD 支持	247
11.4 用 GD 生成图像	248
11.4.1 直线和文本	248
11.4.2 为图像添加 TrueType 字体	251
11.4.3 向基本图像中添加文本	253
11.4.4 用 GD 生成矩形	257
第 12 章 PHPGTK	265
12.1 PHPGTK 介绍	265
12.2 安装 PHPGTK	266
12.2.1 说明	266
12.2.2 在 Windows 上安装	266
12.2.3 在 Linux 上安装	267
12.3 创建基本的 GTK 对象	269
12.3.1 创建 PHPGTK 窗口	269
12.3.2 添加一个 GTK 按钮	272
12.3.3 修改基本窗口组件的外观	273
12.3.4 文本输入	274
12.3.5 hbox 和 vbox	275
12.3.6 pack_start	275
12.4 组合窗口组件	277
12.5 一个简单的 PHPGTK 应用程序	283

第 1 章

PHP 概述

本章内容

- 概述
- PHP 句法
- 变量
- 运算符
- 数组
- if/then 语句
- switch 语句
- for 循环
- foreach 循环
- while 循环
- do while 循环
- 用户定义函数
- PHP 的面向对象程序设计
- phpinfo()
- 补充资料

1.1 概述

本书假定用户已经具备了一些 PHP 编程经验。这不是一本 PHP 编程入门书籍，书中介绍的是一些 PHP 更高级的使用方法，这在那些网上的入门书籍或教程中是找不到的。

虽然如此，这里还是要对 PHP 做一些简要的介绍，主要是针对想快速温习 PHP 基础

知识的用户，或是已经熟悉另一种类似语言（如 PERL）的用户。

1.2 PHP 句法

PHP 是一种嵌入 HTML 页面的语言（尽管可以不这么做）。大多数 PHP 页面是 PHP 代码和 HTML 代码的混合体。当 Web 服务器读取 PHP 页面，它会寻找 PHP 代码的起始标签（`<?php`）和结束标签（`?>`），以区分 HTML 代码。

如果配置了 `php.ini` 文件使其支持短标签（通常默认是支持的），那么就可以使用句法 `<?和?>` 代替 `<?php 和?>`。此外，还可以配置 `php.ini` 文件支持 ASP 风格的标签 `<%和%>`。该功能默认设置是关闭的，它的惟一目的好像就是允许某些 HTML 编辑器能够识别中间代码以区别 HTML 代码，在这种情况下，编辑器不会把 HTML 句法规则强加到 PHP 代码中。

下面是一个 PHP 代码嵌入到 HTML 中的简单例子：

```
<h1>Welcome To The Web Site Of <? echo $company; ?>!</h1>
```

以上代码经过 Web 服务器的解析后，会在 PHP 代码的位置替换成公司名称再输出。

通常，如果使用了开始或结束括号，单行 PHP 代码应该以分号结束（这不是必须的，但对于 `if/then` 语句来说，以分号结束是很有意义的）。

例如：

```
<?
echo "<p>a line of code";
echo "<p>another line of code";
?>
```

1.3 变量

也可以把短的 PHP 代码放入 HTML 中，例如在打印变量值时：

```
<h1>Today's Date is <? = $date ?>.</h1>
```

PHP 的 `<? =` 后紧跟一个变量，相当于 `echo()` 函数的缩写。

PHP 脚本程序中可以包含注释。Web 服务器会忽略这些注释，PHP 代码注释的内容不会发往浏览器。注释有 3 种格式：

- `#` —— 类似于 PERL 的注释；`#` 符号后为注释内容。
- `//` —— 类似于 JavaScript 的注释；`//` 符号后为注释内容。
- `/*` 和 `*/` —— 两个符号之间为注释内容。与 C 语言的注释风格相同。

下面是一个 PHP 代码中包含注释的例子：

```
<?
echo "Hello"; #prints out "Hello"
echo "Hello"; //prints out "Hello";
/* The following
prints out "Hello" */
echo "Hello";
?>
```

最常见的注释是使用//或/*和*/符号。#符号几乎很少见，尽管它也是有效的注释符。PHP 变量用\$符号表示。把字符串值“Hello World”赋给变量\$a，只需这样写：

```
$a = "Hello World";
```

字符串必须加引号，字符串里也可以包含变量：

```
$a = 4;
$string = "The value of a is $a";
// $string = "The Value of a is 4";
```

PHP 变量无需预先声明，也不需要类型定义。注意，如果还未给一个变量赋值就使用它，会得到一个关于使用了未定义变量的错误警告（取决于 php.ini 中的错误报告设置，参见第 8 章）。例如：

```
$a = 4;
$c = $a + $b;
// $c = 4, but a warning appears "Warning: Undefined
variable..".
```

警告不会终止脚本程序的继续执行。但如果忘记了行代码结尾处的分号，就会产生一个解析错误，从而导致脚本程序停止运行。

因为 PHP 变量没有类型，所以无需担心赋值语句会产生错误，这点不同于 C 语言。例如：

```
$a = 4;
$b = "5";
$c = $a + $b;
// $c = 9;
```

PHP 还支持布尔变量，可以给布尔变量赋值 1 或 0，也可以赋值 true 或 false：

```
$a = true;
```