

PHP程序设计 基础教程

- ◆ PHP入门与环境搭建
- ◆ PHP开发基础
- ◆ 运算符和表达式
- ◆ 流程控制语句
- ◆ PHP数组
- ◆ 函数
- ◆ 面向对象编程
- ◆ 字符串操作
- ◆ PHP和Web页面交互
- ◆ PHP会话控制
- ◆ PHP+MySQL开发实战
——学生成绩管理系统
- ◆ PHP+MySQL开发实战
——个人博客系统



张艳 庞海波 主编
许胜礼 丁玉涛 王兆庆 副主编

高等学校计算机应用规划教材

PHP 程序设计

基础教程

张 艳 庞海波 主 编
许胜礼 丁玉涛 王兆庆 副主编

清华大学出版社

北 京

内 容 简 介

本书全面讲述 PHP 程序开发的相关基础知识和详细技术。全书共分为 12 章, 深入介绍 PHP 入门与开发环境的安装和配置、PHP 相关的基本语法、运算符和表达式、流程控制语句、PHP 数组和函数、面向对象编程基础、字符串操作、PHP 和 Web 页面交互、PHP 会话控制, 最后给出完整的开发实例。

本书内容丰富、结构合理、思路清晰、语言简练流畅、示例翔实。本书可作为高等院校网站设计与制作及其相关专业、Web 编程专业的教材, 还可作为 Web 应用开发人员的参考资料。

本书对应的电子课件、习题答案和实例源文件可以到 <http://www.tupwk.com.cn/downpage> 网站下载。

本书封面贴有清华大学出版社防伪标签, 无标签者不得销售。

版权所有, 侵权必究。侵权举报电话: 010-62782989 13701121933

图书在版编目(CIP)数据

PHP 程序设计基础教程/张艳, 庞海波 主编. —北京: 清华大学出版社, 2018

(高等学校计算机应用规划教材)

ISBN 978-7-302-50057-5

I. ①P… II. ①张…②庞… III. ①PHP 语言—程序设计—高等学校—教材 IV. ①TP312.8

中国版本图书馆 CIP 数据核字(2018)第 088827 号

责任编辑: 胡辰浩 李维杰

装帧设计: 牛艳敏

责任校对: 孔祥峰

责任印制: 沈 露

出版发行: 清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址: 北京清华大学学研大厦 A 座 邮 编: 100084

社 总 机: 010-62770175 邮 购: 010-62786544

投稿与读者服务: 010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈: 010-62772015, zhiliang@tup.tsinghua.edu.cn

印 装 者: 三河市铭诚印务有限公司

经 销: 全国新华书店

开 本: 185mm×260mm 印 张: 18.75 字 数: 468 千字

版 次: 2018 年 6 月第 1 版 印 次: 2018 年 6 月第 1 次印刷

印 数: 1~3500

定 价: 58.00 元

产品编号: 077946-01

前 言

信息技术的飞速发展大大推动了社会的进步，已经逐渐改变了人类的生活、工作和学习方式。PHP 是全球最普及、应用最广泛的 Web 应用程序开发语言之一，多年来始终保持在最流行编程语言排行榜的前五位。PHP 是一种跨平台的、开源的服务器端嵌入式脚本语言，其简单易学的特点，在全球范围内受到广大程序员的认同和青睐。

在过去的十年间，PHP 已经从一套为 Web 站点开发人员提供的简单工具演变成完整的面向对象编程语言。在 Web 应用开发方面，PHP 现在可与 Java 和 C# 这样的主流编程语言抗衡，越来越多的公司为了给站点提供更加强大的功能而采用 PHP。PHP 的简单易学性和强大功能使其得到了广泛应用。

本书作者具有多年的开发和教学经验，筛选出适合教学的开发案例，详细介绍了 PHP 程序设计所涉及的重要知识。本书通过结合不同难度的案例，全面介绍了 PHP 程序开发技术。本书深入介绍了 PHP 入门知识及开发环境的安装和配置、PHP 相关的基本语法、运算符和表达式、流程控制语句、PHP 数组和函数、面向对象编程基础、字符串操作、PHP 和 Web 页面交互、PHP 会话控制，最后给出了两个完整的开发实例。在每一章末尾都安排了有针对性的思考练习题和编程题，有助于读者巩固所学的基本概念，并针对本章重点设计了编程题，有助于培养读者的实际动手能力、增强对基本概念的理解和实际应用能力。

本书内容丰富、结构合理、思路清晰、语言简练流畅、示例翔实。本书可作为高等院校网站设计与制作及其相关专业、Web 编程专业的教材，还可作为 Web 应用开发人员的培训和参考资料。

本书是集体智慧的结晶，其中，第 1、第 2 和第 4 章由张艳编写，第 3、第 5 和第 10 章由丁玉涛编写，第 6 至第 8 章由庞海波编写，第 9 和第 12 章由许胜礼编写，第 11 章由王兆庆和庞海波编写。另外，参加编写的人员还有王秀玲、陶永才、石育澄、巴阳、赵国桦、丁鑫、海朝阳、曹朝阳、张鑫倩、杨朝阳、火昊、任鹏程、王战红、贾圣杰、姚瑶、郭华杰、王亚敏等。

由于作者水平有限，本书难免有不足之处，欢迎广大读者批评指正。我们的信箱是 huchenhao@263.net，电话是 010-62796045。

本书对应的电子课件、习题答案和实例源文件可以到 <http://www.tupwk.com.cn/downpage> 网站下载。

作 者
2018 年 2 月

目 录

第 1 章 PHP 入门与环境搭建	1
1.1 PHP 概述.....	1
1.1.1 什么是 PHP.....	1
1.1.2 PHP 的版本.....	1
1.1.3 PHP 语言的优势.....	3
1.1.4 PHP 常用工具.....	3
1.2 PHP 程序的工作流程.....	5
1.2.1 PHP 的工作流程.....	5
1.2.2 PHP 服务器.....	6
1.3 PHP 集成运行环境工具的安装与配置.....	7
1.3.1 WampServer 的安装步骤.....	7
1.3.2 集成运行环境的配置.....	12
1.4 第一个 PHP 程序.....	21
1.4.1 开发第一个 PHP 网页.....	21
1.4.2 设置 PHP 网站.....	22
1.4.3 运行 PHP 网站.....	23
1.5 本章小结.....	23
1.6 思考与练习.....	24
第 2 章 PHP 开发基础	25
2.1 PHP 基本语法.....	25
2.1.1 PHP 标记符.....	25
2.1.2 PHP 注释.....	26
2.1.3 PHP 语句和语句块.....	27
2.1.4 PHP 编码规范.....	28
2.1.5 PHP 命名规则.....	29
2.2 常量.....	30
2.2.1 自定义常量.....	30
2.2.2 预定义常量.....	31
2.3 变量.....	32

2.3.1 变量的概念.....	32
2.3.2 变量的声明和使用.....	32
2.3.3 变量的作用域和生存周期.....	33
2.3.4 变量的生存周期.....	35
2.3.5 可变变量与变量的引用.....	36
2.4 数据类型.....	37
2.4.1 标量数据类型.....	37
2.4.2 复合数据类型.....	40
2.4.3 特殊数据类型.....	41
2.4.4 检测数据类型.....	42
2.5 数据类型的转换.....	43
2.5.1 自动类型转换.....	43
2.5.2 强制类型转换.....	45
2.6 数据的输出.....	46
2.6.1 print 和 echo.....	46
2.6.2 输出运算符“<?= ?>”.....	47
2.7 本章小结.....	47
2.8 思考和练习.....	47
第 3 章 运算符和表达式	49
3.1 算术运算符.....	49
3.1.1 常用算术运算符.....	50
3.1.2 累加、累减运算符.....	50
3.2 字符串运算符.....	51
3.3 赋值运算符.....	52
3.4 比较运算符.....	53
3.5 逻辑运算符.....	54
3.6 按位运算符.....	55
3.7 错误控制运算符.....	56
3.8 三元运算符.....	56
3.9 运算符的优先级和结合规则.....	57
3.10 表达式.....	57

3.11	小结	58	5.4.2	用 foreach 循环访问数组中元素的键和值	88
3.12	思考和练习	58	5.4.3	用 foreach 循环修改数组中元素的值	89
第 4 章	流程控制语句	59	5.5	多维数组	90
4.1	流程控制概述	59	5.5.1	创建多维数组	90
4.2	条件控制语句	59	5.5.2	访问多维数组中的元素	92
4.2.1	if 语句	60	5.5.3	多维数组的循环访问	92
4.2.2	if...else 语句	61	5.6	数组的操作	94
4.2.3	if...elseif...else 语句	62	5.6.1	数组的排序	94
4.2.4	switch 语句	64	5.6.2	添加和删除数组元素	100
4.3	循环控制语句	66	5.6.3	数组的合并	105
4.3.1	while 循环语句	67	5.6.4	数组与字符串之间的转换	107
4.3.2	do-while 循环语句	68	5.6.5	把数组转换为变量列表	108
4.3.3	for 循环语句	69	5.7	本章小结	108
4.3.4	foreach 循环语句	71	5.8	思考和练习	109
4.4	跳转语句	72	第 6 章	PHP 函数	111
4.4.1	break 语句	72	6.1	创建和使用简单函数	111
4.4.2	continue 语句	74	6.2	创建和调用接受参数的函数	120
4.4.3	exit 语句	75	6.3	设置参数默认值	123
4.5	本章小结	75	6.4	创建和使用带有返回值的函数	126
4.6	思考和练习	75	6.5	理解变量的作用域	130
第 5 章	数组	77	6.6	本章小结	133
5.1	数组概述	77	6.7	思考和练习	134
5.2	创建数组	78	第 7 章	面向对象基础知识	135
5.3	访问数组中的元素	78	7.1	面向对象的基本概念	135
5.3.1	改变元素内容	79	7.1.1	类	135
5.3.2	用 print_r() 函数输出整个数组	80	7.1.2	对象	135
5.3.3	用 array_slice() 函数读取数组中的连续几个元素	82	7.1.3	面向对象编程的三大特点	135
5.3.4	统计数组中元素的个数	83	7.2	PHP 与对象	136
5.3.5	逐个访问数组中的元素	84	7.2.1	类的定义	136
5.4	用 foreach() 循环访问数组	87	7.2.2	成员方法	137
5.4.1	用 foreach 循环访问数组中每个元素的值	88			

7.2.3 类的实例化	137	8.6.8 格式化字符串	186
7.2.4 成员变量	138	8.6.9 分割字符串	187
7.2.5 类常量	139	8.6.10 合成字符串	188
7.2.6 构造方法和析构方法	140	8.7 本章小结	189
7.2.7 继承和多态的实现	143	8.8 思考和练习	189
7.2.8 “\$this->”和“: :” 的使用	146	第 9 章 PHP 与 Web 页面交互	191
7.2.9 数据隐藏	147	9.1 PHP Web 编程基础	191
7.2.10 静态变量(方法)	149	9.2 Web 表单	193
7.3 PHP 对象的高级应用	151	9.2.1 创建表单	193
7.3.1 final 关键字	151	9.2.2 认识表单元素	194
7.3.2 抽象类	152	9.3 PHP 与 Web 页面交互的 基本方法	195
7.3.3 接口的使用	153	9.3.1 访问和获取 Web 表单数据	195
7.3.4 克隆对象	155	9.3.2 Web 表单数据的 有效性验证	199
7.3.5 比较对象	157	9.3.3 Web 表单的安全性验证	202
7.3.6 检测对象类型	157	9.3.4 PHP 文件上传处理	204
7.3.7 魔术方法()	158	9.4 PHP 全局变量	207
7.4 面向对象的应用—— 中文字符串的截取类	163	9.4.1 全局变量\$_SERVER	208
7.5 本章小结	165	9.4.2 全局变量\$_GET	210
7.6 思考和练习	165	9.4.3 全局变量\$_POST	211
第 8 章 字符串	167	9.4.4 全局变量\$_SESSION	213
8.1 字符串简介	167	9.5 本章小结	217
8.2 引用字符串常量	168	9.6 思考和练习	217
8.3 单引号和双引号的区别	168	第 10 章 会话管理	219
8.4 输出字符串	169	10.1 用 cookie 保存页面状态	219
8.5 字符串的连接符	170	10.1.1 cookie 的组成	220
8.6 字符串操作	171	10.1.2 在 PHP 中设置 cookie	220
8.6.1 去除字符串的首尾空格 和特殊字符	171	10.1.3 在脚本中访问 cookie	221
8.6.2 转义、还原字符串数据	174	10.1.4 删除 cookie	222
8.6.3 获取字符串的长度	176	10.2 用 PHP 会话存储数据	225
8.6.4 截取字符串	178	10.2.1 创建会话	226
8.6.5 比较字符串	180	10.2.2 读取和写入会话数据	226
8.6.6 检索字符串	182	10.2.3 撤销会话	229
8.6.7 替换字符串	184		

10.3	本章小结	233	12.1.3	系统功能设计	254
10.4	思考和练习	234	12.1.4	文件组织	255
第 11 章	综合案例——		12.2	数据库设计	255
	学生成绩管理系统	235	12.2.1	数据库结构设计	256
11.1	需求分析	235	12.2.2	数据库连接程序设计	257
11.1.1	系统描述	235	12.3	首页设计	258
11.1.2	系统设计目标	235	12.3.1	首页布局	258
11.1.3	系统功能设计	236	12.3.2	首页实现	259
11.1.4	文件组织	236	12.4	用户注册模块设计	263
11.2	系统功能设计	237	12.5	博客文章模块设计	268
11.2.1	前台成绩查询系统	237	12.5.1	文章列表	269
11.2.2	后台数据管理系统	238	12.5.2	发表文章	269
11.3	数据库设计	239	12.5.3	查询文章	274
11.4	各模块功能描述	242	12.5.4	我的文章	278
11.5	本章小结	252	12.6	图片管理模块设计	281
11.6	思考和练习	252	12.6.1	显示图片	282
第 12 章	综合案例——		12.6.2	添加图片	283
	个人博客系统	253	12.6.3	浏览图片	288
12.1	个人博客系统分析	253	12.7	本章小结	292
12.1.1	系统描述	253	12.8	思考和练习	292
12.1.2	系统设计目标	254			

第1章 PHP入门与环境搭建

PHP 是一种跨平台、HTML 嵌入式的服务器端脚本语言，是全球普及、应用最广泛的 Web 应用程序开发语言之一。本章主要介绍 PHP 的入门知识、基本概念、工作流程，以及 PHP 开发工具的安装和环境配置，并且制作第一个 PHP 实例。

本章的主要学习目标：

- 了解 PHP 基础知识
- 掌握 PHP 语言的基本概念
- 掌握常用 PHP 开发工具的安装和环境配置

1.1 PHP 概述

1.1.1 什么是 PHP

PHP 是 Hypertext Preprocessor(超文本预处理器)的缩写，是全球最流行的 Web 应用程序开发语言之一。PHP 是一种跨平台、HTML 嵌入式的服务器端脚本语言，和微软的 ASP 颇有几分相似，都是一种在服务器端执行的嵌入 HTML 文档的脚本语言；混合了 C、Java 和 Perl 等现代编程语言的长处以及 PHP 自创的新语法，语法简单、易于学习、功能强大、灵活易用，目标就是让网页开发人员快速编写出动态的网页；用 PHP 做出的动态页面与用其他编程语言做出的相比，具有更快的执行速度，因为 PHP 充分利用了服务器的性能，其执行引擎还会将用户经常访问的 PHP 程序驻留在内存中，用户再次访问这个程序时就不需要重新编译程序了，直接执行内存中的代码即可，这也是 PHP 高效率的体现之一；PHP 支持几乎所有流行的数据库以及操作系统，完全不必考虑跨平台问题；PHP、Apache 和 MySQL 的组合已成为 Web 服务器的一种配置标准。

1.1.2 PHP 的版本

PHP 最初只是一个用 Perl 语言编写的简单程序，用来统计网站的访问者。经过慢慢地完善，在 2000 年 5 月发布了官方正式版本——基于该引擎并结合了更多新功能的 PHP 4.0。

1. PHP/FI

1995 年，Rasmus Lerdorf 创建了一套简单的 Perl 脚本，用来跟踪访问他个人主页的人们的信息，并把它取名为“Personal Home Page Tools”，简称 PHP/FI。后来 Rasmus 用 C 语言

对它进行了重写，做出了一个可以访问数据库、开发简单的动态 Web 程序的工具。Rasmus 发布了 PHP/FI 的源代码，以便每个人都可以使用它，同时大家也可以修正它的 Bug 并且改进它的源代码。PHP/FI 后续版本 2.0 于 1997 年 11 月正式发布，但是那时只有几个人在为该工程撰写少量代码，它仍然只是少数人的工程。

2. PHP 3.0

1998 年 6 月正式发布了官方 PHP 3.0 版，PHP 3.0 是类似于当今 PHP 语法结构的第一个版本。Andi Gutmans 和 Zeev Suraski 在为所大学的项目中开发电子商务程序时发现，PHP/FI 2.0 功能明显不足，于是他们重写了代码，这就是 PHP 3.0。考虑到 PHP/FI 已存在的用户群，从 PHP/FI 2.0 的名称中移去了暗含“本语言只限于个人使用”的部分，最终被命名为简单的缩写“PHP”。PHP 3.0 除给最终用户提供数据库、协议和 API 的基础结构外，它强大的可扩展性还吸引了大量的开发人员加入并提交新的模块，这也是 PHP 3.0 取得巨大成功的关键。PHP 3.0 中的其他关键功能包括对面向对象的支持以及更强大和协调的语法结构。

3. PHP 4.0

1998 年的冬天，在 PHP 3.0 官方发布后不久，Andi Gutmans 和 Zeev Suraski 开始重新编写 PHP 代码，以提高复杂程序运行时的性能和 PHP 自身代码的模块性。虽然 PHP 3.0 的新功能和广泛的第三方数据库、API 的支持使得编写这样的程序成为可能，但是 PHP 3.0 没有高效处理如此复杂程序的能力。在 1999 年中期，新的被称为“Zend Engine”（这是 Zeev 和 Andi 的缩写）的引擎首次引入 PHP，基于该引擎并结合更多新功能的 PHP 4.0 于 2000 年 5 月发布了官方正式版本。

4. PHP 5.0

在 2004 年 6 月份的时候，PHP 的发展到达了第二个里程碑。带有二代 Zend Engine 的 PHP 5.0 正式发布，PHP 5.0 引入了新的对象模型和大量新功能，而且性能明显增强。直到 2008 年，很多程序都不再支持 PHP 4.0 版本，取而代之的是 PHP 5.0。

5. PHP 6.0

PHP 5.0 发布后，收到最多的反馈内容就是在 PHP 中缺少对编码转换的支持。在 Andrei Zmievski 的领导下，PHP 中嵌入了 ICU 库，使文本字符串以 Unicode-16 的方式呈现。这一举动对 PHP 本身以及用户的编码方式产生了巨大的改变，所以 PHP 6.0 应运而生了。但是由于这一改变跨越较大，开发人员无法很好地理解所发生的改变，并且转换导致性能下降，另外 2009 年发布的 PHP 5.3，还有 2010 年发布的 PHP 5.4，几乎涵盖了所有从 PHP 6.0 移植而来的功能；因此，在 2010 年这一工程就停止了，直到 2014 年也没有被人们所接受。

6. PHP 7.0

2014 年至 2015 年期间，PHP 7.0 正式发布了。PHP 7.0 最主要的目标就是通过重构 Zend Engine，使 PHP 的性能更加优化，同时保留语言的兼容性。由于是对引擎加以重构，因此 PHP 7.0 的引擎目前已是第三代 Zend Engine。

1.1.3 PHP 语言的优势

PHP 能够迅速发展，并得到广大使用者的喜爱，主要原因是 PHP 除了拥有一般脚本具有的功能外，还具备自身的优势，具体如下：

- 源代码完全公开：事实上，所有的 PHP 源代码都可以获得。读者也可以通过 Internet 获得所需要的源代码，快速修改并利用。
- 完全免费：和其他技术相比，PHP 本身是免费的。读者使用 PHP 进行 Web 开发无须支付任何费用。
- 语法结构简单：PHP 结合了 C 语言和 Perl 语言的特色，编写简单，方便易懂；可以嵌入到 HTML 语言中，实用性强，更适合初学者。
- 跨平台性强：PHP 是运行在服务器端的脚本，可以运行在 Linux 和 Windows 等操作系统下。
- 效率高：PHP 消耗相当少的系统资源，并且程序开发快、运行快。
- 强大的数据库支持：支持目前所有的主流和非主流数据库，使 PHP 的应用对象非常广泛。
- 面向对象：在 PHP 中，面向对象有了很大的改进，PHP 完全可以用来开发大型商业程序。

1.1.4 PHP 常用工具

制作 PHP 动态网站可分为两个方面：一是网站的界面设计，主要是用浏览器能理解的代码及图片设计网页；二是使用 PHP 语言进行网站程序设计和代码实现，用来实现网站的新闻管理、与用户进行交互等各种功能。

1. 网页设计工具

(1) Dreamweaver

Dreamweaver 是网页制作“三剑客”之一，其功能更多体现在对 Web 页面的设计上。随着 Web 语言的发展，Dreamweaver 的功能早已不再仅限于网页设计方面，而是更多支持各种 Web 应用流行的前后台技术的综合应用。Dreamweaver 对 PHP 的支持十分到位，不但对 PHP 的不同方面清晰地表示，并且给予足够的编程提示，使编程过程相当流畅。

(2) Squire

HTML5 现在已经成为最流行的标记语言，拥有成熟的社区和广泛的浏览器支持，HTML5 完备的功能和强大的拓展性使得设计师和开发者可以点石成金。更多的可控元素，更自由的交互设计，变化随心的动效，丰富生动的多媒体，都可以借助 HTML5 一手掌控。Squire 是一款 HTML5 富文本编辑器，兼容不同浏览器的标准，轻巧灵活，让你制作网页如同写文档一般轻松。

2. PHP 代码开发工具

1) 文本编辑工具

Windows 系统自带的记事本是一款体积小、启动快、占用内存小、容易使用、具备最基

本的文本编辑功能的工具。

UltraEdit 是一套功能强大的文本编辑器，可以编辑文本、十六进制值、ASCII 码，完全可以取代 Windows 记事本，并且内建了英文单词检查、C++ 及 VB 指令突显等功能。该软件还附有 HTML 标签颜色显示、搜索替换以及无限制的还原功能，可以满足用户的一切编辑需要。

2) IDE

IDE 是集成开发环境(Integrated Development Environment)的英文简称，是集成了代码编写功能、分析功能、编译功能、调试功能于一体的软件开发套件。目前常用于 PHP 的 IDE 包括以下几种：

Notepad++: Notepad++ 是一款 Windows 环境下免费开源的代码编辑器，支持的语言包括 C、C++、Java、C#、XML、HTML、PHP、Javascript 等。Notepad++ 不仅有语法高亮显示功能，也有语法折叠功能，并且支援宏以及扩充基本功能的外挂模组。

PHPEdit: PHPEdit 是 Windows 环境下一款优秀的 PHP 脚本 IDE。该软件为快速、便捷地开发 PHP 脚本提供了多种工具，其功能包括：语法关键词高亮；代码提示、浏览；集成 PHP 调试工具；帮助生成器；自定义快捷方式；150 多个脚本命令；键盘模板；报告生成器；快速标记；插件等。

phpDesigner: phpDesigner 是 Linux 环境下十分流行的免费 PHP 编辑器，小巧且功能强大。它以 Linux 环境下的 gedit 文本编辑器为基础，是专门用来编辑 PHP 和 HTML 的编辑器，可以显著标识 PHP 和 HTML、CSS 以及 SQL 语句。在编写过程中提供函数列表参考、函数参数参考、搜索和检测语法等。

Zend Studio: 由 Zend 科技开发的一个针对 PHP 的全面开发平台，这个 IDE 融合了 Zend Server 和 Zend Framework，并且融合了 Eclipse 开发环境。Eclipse 是最早适用于 Java 的 IDE，缘于优良的特性和对 PHP 的支持，成为极具影响力的 PHP 开发工具，是最好的 PHP IDE 之一。Zend Studio 具备功能强大的专业编辑工具和调试工具，支持 PHP 语法高亮显示，支持语法自动填充功能，支持书签功能，支持语法自动缩排和代码复制功能，内置强大的 PHP 代码调试工具，支持本地和远程两种调试模式，支持多种高级调试功能。Zend Studio 可以在 Linux、Windows、Mac OS X 环境下运行。

PHP 开发工具有很多，但是建议使用记事本等轻型编辑器进行前期的学习，这不仅是因为程序体积小、安装方便、消耗系统资源少，最重要的是把代码完完整整通过敲击键盘按键编辑出来，有利于加强对 PHP 语法规则的记忆和理解。

3. PHP 集成运行环境工具

为了建立 PHP 动态网站，首先需要搭建 PHP 的开发和运行环境。对新手来说，一般选择在 Windows 平台上使用 Apache、MySQL 和 PHP 这种搭配组合，Apache 是类似 IIS 的 Web 服务器软件，MySQL 是数据库，这种组合也称 WAMP(W 代表 Windows、A 代表 Apache、M 代表 MySQL、P 代表 PHP)。下面介绍几款在 Windows 环境下可以使用的 WAMP 集成工具。

WampServer: WampServer 集成了 Apache、MySQL、PHP、phpmyadmin，支持 Apache 的 mod_rewrite 操作，PHP 扩展和 Apache 操作只需要在菜单中操作就可以，省去了修改配置文件的麻烦。可以单独开启 Apache 或 MySQL，也支持中文界面。

APMServ: APMServ 是一款拥有图形界面的绿色软件，无须安装，具有灵活的移动性，只要单击 APMServ 的启动按钮即可自动进行相关设置，拥有跟 IIS 一样便捷的图形管理界面。

XAMPP: XAMPP 是一款具有中文说明，但不支持中文界面的集成环境。XAMPP 不仅仅针对 Windows，也适用于 Linux 等其他操作系统；缺点是集成功能较多，不支持中文界面，操作不容易，安全设定较繁琐。

本书之所以选择将 PHP 作为动态网站的开发语言，主要是考虑到 PHP 语法结构简单、易学，而动态网站开发语言的编程思想是十分相似的，每种语言基本上都定义了一些服务器与浏览器之间交互信息的方法，只要能深刻掌握其中一种，再去学习其他语言就很容易了。另外，WampServer、AppServ 等集成环境的出现使配置 PHP 的 Web 服务器也变得更加简单，初学者能在短时间内学会 Web 应用程序的开发流程。

1.2 PHP 程序的工作流程

1.2.1 PHP 的工作流程

一个完整的 PHP 系统由以下几部分组成：

- **操作系统:** 网站运行服务器所使用的操作系统。PHP 语言具有跨平台性，可以运行在任何操作系统上，如 Windows、Linux 等。
- **服务器:** 搭建 PHP 运行环境时所选择的服务器。PHP 支持多种服务器软件，包括 Apache、IIS 等。
- **PHP 包:** 实现对 PHP 文件的解析和编译。
- **数据库系统:** 实现系统中数据的存储。PHP 支持多种数据库系统，包括 MySQL、SQL Server、Oracle 以及 DB2 等。
- **浏览器:** 用于浏览网页。因为 PHP 在发送到浏览器时被解析器编译成其他的代码，所以 PHP 对浏览器没有任何限制。

图 1-1 展示了 PHP 页面运行的全过程：

第一步，将 PHP 代码传递给 PHP 包，请求对 PHP 包进行解析并编译。

第二步，服务器根据 PHP 代码请求读取数据库，调用数据。

第三步，服务器与 PHP 包共同根据数据库中的数据或其他运行变量，将 PHP 代码解析为普通的 HTML 代码。

第四步，将解析后的代码发送给浏览器，浏览器对代码进行分析，获取可视化内容。

第五步，用户通过访问浏览器浏览网站内容。

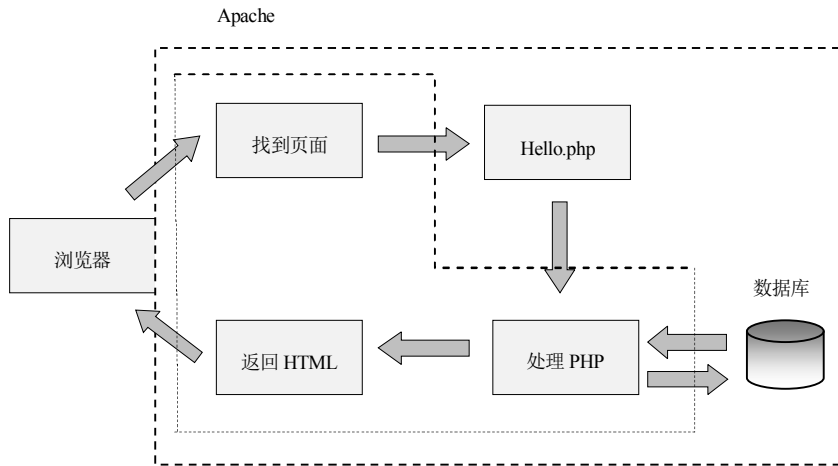


图 1-1 PHP 的工作原理

1.2.2 PHP 服务器

1. PHP 预处理器

PHP 预处理器的功能是解析 PHP 代码，主要是将 PHP 程序代码解析为文本信息，而且这些文本信息中也可以包含 HTML 代码。

2. Web 服务器

Web 服务器也称为 WWW(World Wide Web)服务器，功能是解析 HTTP。当 Web 浏览器向 Web 服务器发送一个 HTTP 请求时，PHP 预处理器会对该请求对应的程序进行解析并执行，然后 Web 服务器会向浏览器返回一个 HTTP 响应，该响应通常是一个 HTML 页面，包含用户请求的信息，供用户浏览。

目前市面上的 Web 服务器也有很多种，常见的有开源的 Apache 服务器、微软的 IIS 服务器、Tomcat 服务器等，本书使用的是 Apache 服务器。

3. 数据库服务器

数据库服务器是用于提供数据查询和数据管理服务的软件，这些服务主要有数据查询、数据的增加/删除/修改等操作、查询优化、事务管理、数据安全等。

常见的数据库服务器有 MySQL、Oracle、SQL Server、Access 等，其中 MySQL 以其强大的功能、使用和安装便捷、较快的运行速度而备受中小型网站青睐，本书也使用该数据库。

1.3 PHP 集成运行环境工具的安装与配置

对于初学者来说, Apache、PHP 以及 MySQL 的安装和配置较为复杂, 这时可以选择 WAMP(Windows+Apache+MySQL+PHP)集成安装环境以快速安装的配置 PHP 服务器, 集成安装环境就是将 Apache、PHP 以及 MySQL 等服务器软件整合在一起, 免去单独安装和配置服务器带来的麻烦, 实现 PHP 开发环境的快速搭建。

目前比较常用的集成安装环境是 WampServer 和 AppServ, 它们都集成了 Apache 服务器、PHP 预处理器以及 MySQL 服务器。本书以 WampServer 为例介绍 PHP 服务器的安装和配置。

1.3.1 WampServer 的安装步骤

WampServer 是一款由法国软件开发人员开发的、应用在 Windows 环境下的 Apache Web 服务器、PHP 解释器以及 MySQL 数据库的整合软件包, 它使开发人员避免将时间花费在烦琐的环境配置过程上, 从而腾出更多精力去做开发。这个软件是完全免费的, 可以从其官方网站 <http://www.wampserver.com/> 下载到最新的版本, 如图 1-2 所示。



图 1-2 下载 WampServer

本书采用的是 64 位版 Windows 7 系统, 使用的 WampServer 版本是 WampServer 3.0.6 中文 64 位版, 其中集成了包括 Apache 2.4.23、PHP 5.6.25/7.0.10、MySQL 5.7.14 等在内的软件。

WampServer 3.0.6 中文 64 位版软件可以通过常用的中文搜索引擎进行查找和下载, 但需要注意的是下载其 64 位版本, 下载和安装 32 位版本可能出现意想不到的错误。下载的软件名由 WampServer 的版本、适用的操作系统平台、集成的 Apache、MySQL 和 PHP 软件版本等组成, 中间以“_”作为分隔符, 例如 wampserver3_x64_apache2.4.17_mysql5.7.9_php5.6.16_php7.0.0, 其中 wampserver3 是软件的版本系列, x64 表示 Windows 系列的 64 位操作系统平台, apache2.4.17 表示 Apache Web 服务器版本, mysql5.7.9 是 MySQL 数据库版本, php5.6.16 和 php7.0.0 表示软件版本支持的 PHP 解释器版本。

集成运行软件的安装步骤如下：

(1) 双击下载的 WampServer 软件，会出现如图 1-3 所示的安装初始界面，安装软件支持英语和法语界面，默认是 English 语言界面。



图 1-3 选择语言界面

(2) 单击 OK 按钮，进行软件的版权信息设置，如图 1-4 所示。

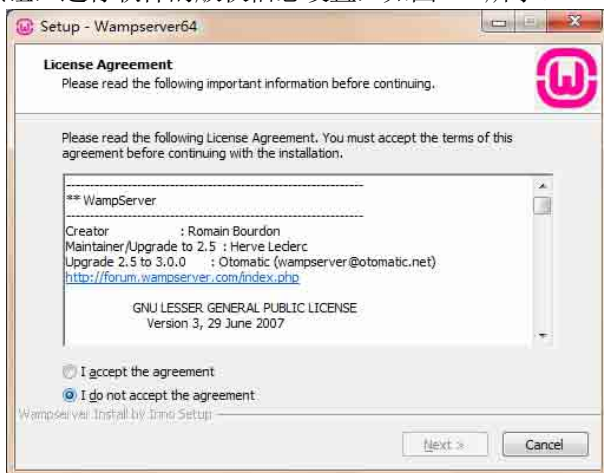


图 1-4 设置版权信息界面

(3) 选中 I accept the agreement 后单击 Next 按钮，进入软件的安装环境确认界面，如图 1-5 所示。

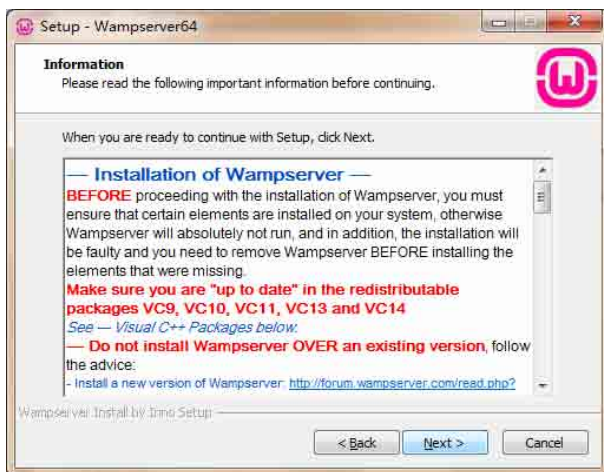


图 1-5 安装环境确认界面

(4) 单击 Next 按钮进入软件的安装目录选择界面, 阅读安装软件所需要的最小硬盘空间, 默认安装在 C 盘根目录下, 如图 1-6 所示。

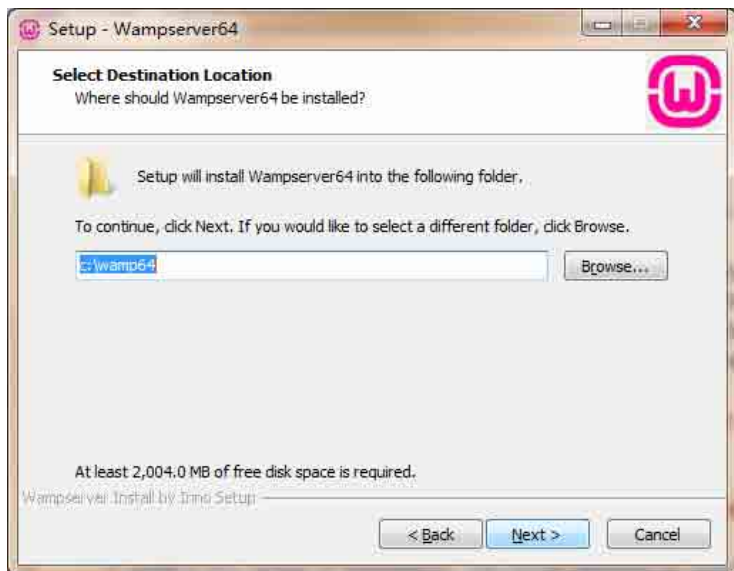


图 1-6 目录选择界面

(5) 使用默认安装目录或修改安装目录后, 单击 Next 按钮进入软件的快捷方式存放目录选择界面, 如图 1-7 所示。默认存放在“开始”菜单中的“程序”目录下, 也可以修改到其他目录下。

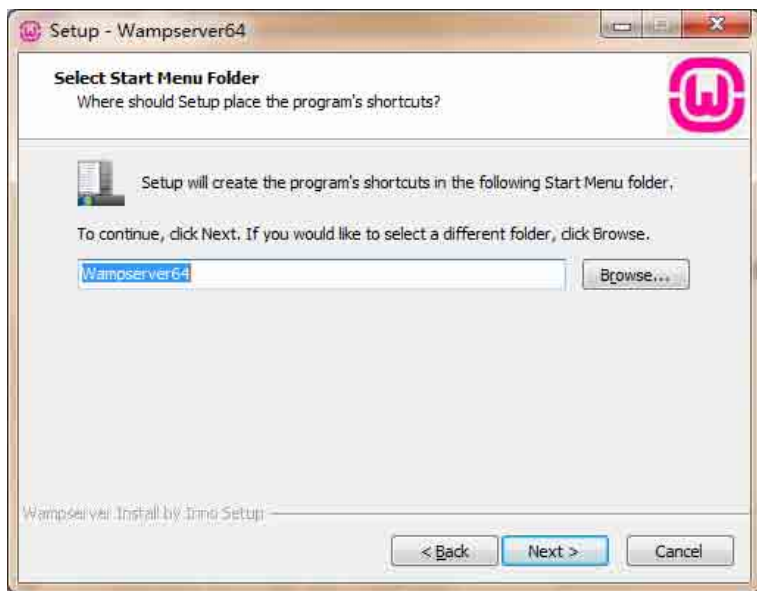


图 1-7 设置快捷方式界面

(6) 单击 Next 按钮进入安装信息确认界面, 如图 1-8 所示。

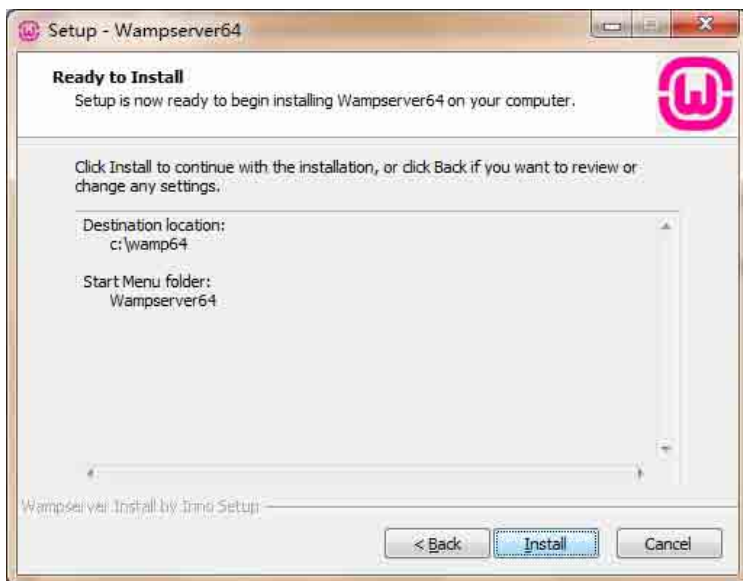


图 1-8 安装信息确认界面

(7) 确认安装信息后，可单击 **Install** 按钮开始正式安装，如图 1-9 所示。也可以单击 **Back** 按钮返回上一步，修改安装目录和软件快捷方式的存放目录。

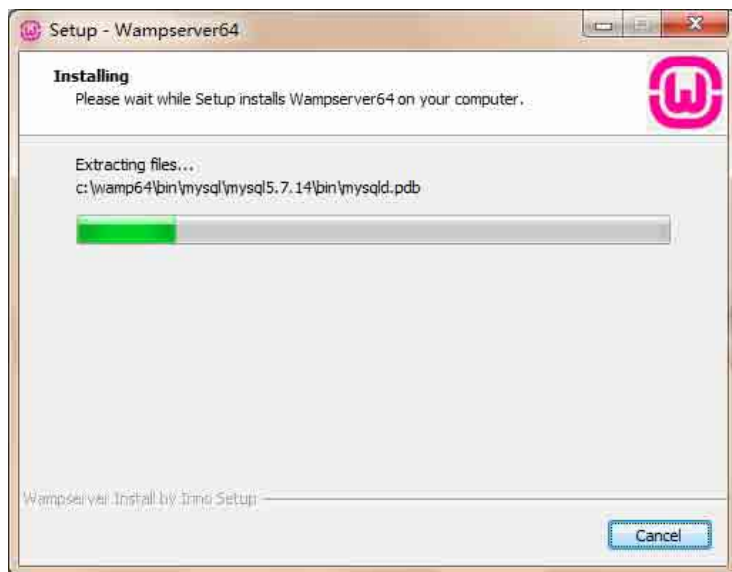


图 1-9 安装界面

(8) 在软件安装过程中会弹出两个对话框，如图 1-10 和图 1-11 所示，询问用户对 WampServer 默认使用的浏览器和代码编辑软件，默认使用微软的 **Internet** 浏览器作为默认浏览器，使用微软操作系统自带的记事本作为代码编辑器，单击“是”接受默认选项，或者单击“否”不接受默认选项。



图 1-10 浏览器选择界面



图 1-11 代码编辑器选择界面

(9) 如果软件在安装过程提示丢失特定的 DLL 文件，那么需要下载并安装所需 DLL 文件，然后重新安装软件。软件基本安装完毕后，会出现如图 1-12 所示的信息提示界面，包括 phpMyadmin 默认的用户名和密码、WampServer 的菜单操作等信息。

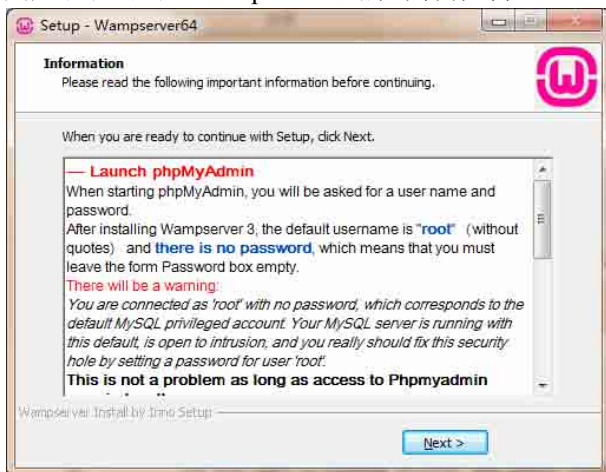


图 1-12 信息提示界面

(10) 单击 Next 按钮后，出现软件安装完成界面，如图 1-13 所示。

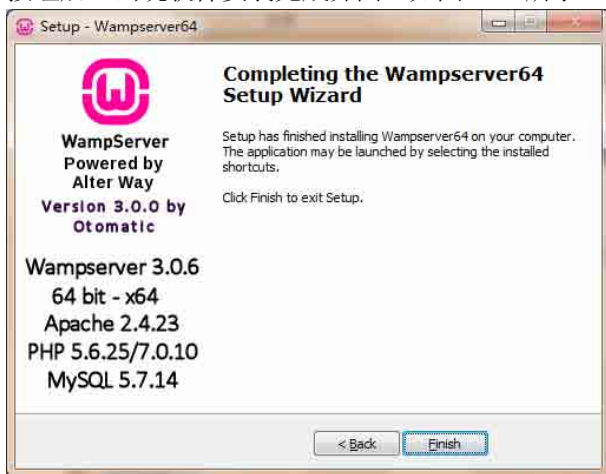


图 1-13 软件安装完成界面

(11) 单击 Finish 按钮就可以完成全部安装操作，双击桌面上的软件快捷方式“Wampserver64”即可打开软件，如图 1-14 所示。



图 1-14 软件快捷方式

1.3.2 集成运行环境的配置

1. 第一次使用集成运行软件

1) 集成运行软件的运行状态

在成功安装 WampServer 后，双击如图 1-14 所示的软件快捷方式，可以在状态栏找到软件图标，会显示 3 种颜色(如图 1-15 所示)，不同的颜色代表不同的含义。如果是红色，表示 Apache 服务器和 MySQL 服务器均未能正常运行；如果是橙色，表示 Apache 服务器或 MySQL 服务器中有一个没有正常运行；如果是绿色，表示两种服务器均正常运行。



图 1-15 WampServer 软件的 3 种运行状态

一般情况下，橙色代表 Apache 服务器未能正常运行，造成这种情况的最常见原因是 80 端口被其他应用程序占用，重新为 Apache 服务器指定端口即可解决。另外也可能是因为 Apache 服务器的某些服务未能正常安装造成，重新安装即可解决。

2) 更改操作界面语言

默认状态下，操作界面的语言是英文，可以用鼠标右键单击状态栏中的软件图标，在弹出的菜单中选择“Language”，在子菜单中选择“chinese”，如图 1-16 所示，将软件的操作界面改为简体中文版。



图 1-16 更改 WampServer 软件的操作界面语言

3) 测试 80 端口

用鼠标右键单击软件图标，在弹出的菜单中选择“Tools”，进入二级菜单后，选择“Test Port 80”。在弹出的命令行界面中，会显示端口 80 被占用的具体信息，如果信息显示 80 端口被 PHP 应用程序占用，就需要为 PHP 的运行开辟其他的端口。单击“Test port 80”菜单项下的“Use a Port other than 80”，会弹出一个对话框，默认会使用 8080 端口，如图 1-17 所示，单击 OK 按钮就会将原来的端口改为 8080 端口。然后在 Tools 菜单中，会出现一个新的子菜单“Test port used: 8080”，如图 1-18 所示，单击会在命令行界面中显示类似于 80 端口的端口 8080 被占用的具体信息。如果使用非 80 端口的其他端口(如 8080)，访问时就必须在域名后加上端口号，如 <http://localhost:8080>。

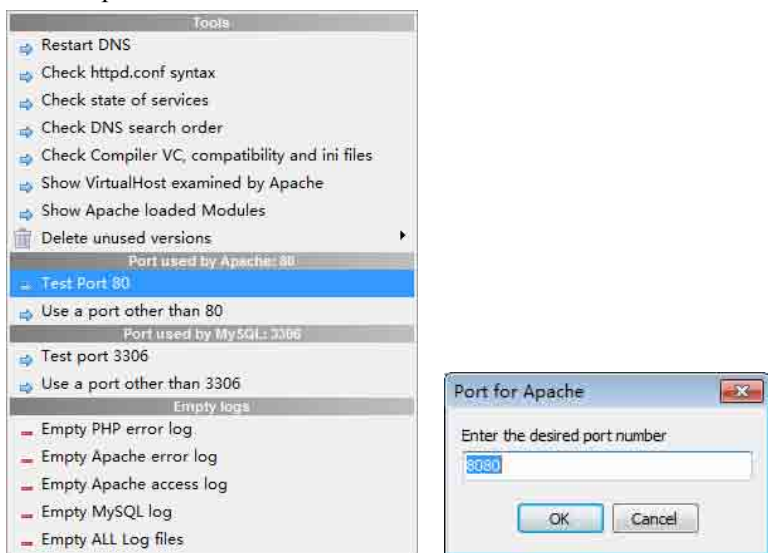


图 1-17 为 Apache 服务器指定其他端口

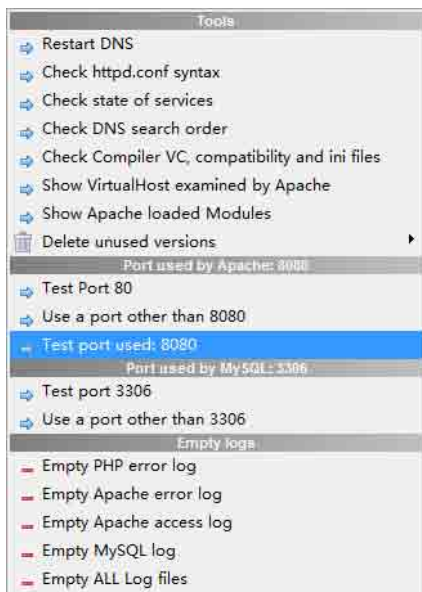


图 1-18 测试自定义的端口

4) 测试集成运行软件安装是否成功

在桌面右下角的状态栏中用鼠标左键单击软件图标，在弹出的菜单中选择 Localhost 命令，如图 1-19 所示。如果能看到如图 1-20 所示的网页，则表示 WampServer 软件安装基本成功。



图 1-19 测试默认网站是否正常运行

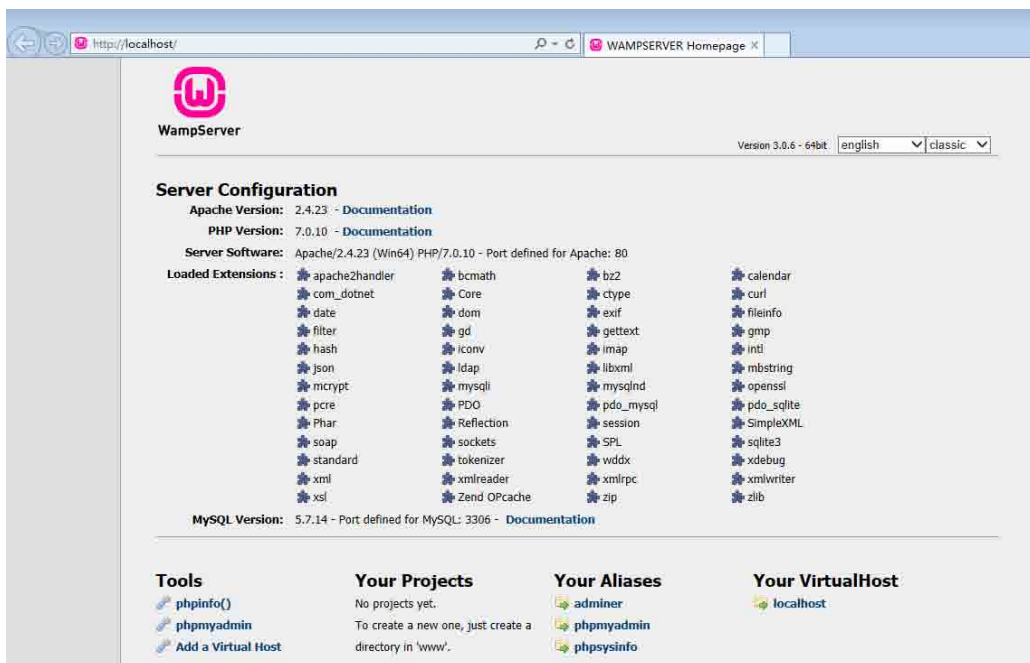


图 1-20 默认网站测试页的运行界面

2. 集成运行软件的常用设置

1) phpMyAdmin

phpMyAdmin 是一个用 PHP 编写的软件工具，可以通过 Web 方式控制和操作 MySQL 数据库。通过 phpMyAdmin 可以对数据库进行全面操作，例如建立、复制和删除数据等，MySQL 数据库的管理就会变得相当简单。单击图 1-19 中的“phpMyAdmin”，可以进入如图 1-21 所示的界面；默认情况下，登录用户名为 root，密码为空，单击“执行”按钮进入如

图 1-22 所示的界面。phpMyAdmin 是用 PHP 语言开发的管理 MySQL 数据库的开源程序，使用 phpMyAdmin 可以对 MySQL 数据库进行新建、删除、编辑、数据备份、数据导入等操作。

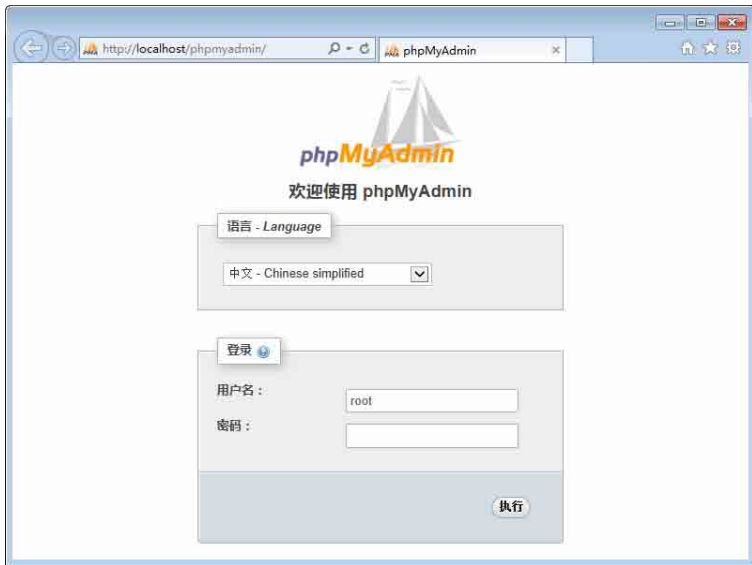


图 1-21 phpMyAdmin 登录界面



图 1-22 phpMyAdmin 配置界面

2) 网站主目录

在如图 1-19 所示的菜单中，单击“www 目录(W)”可以进入网站主目录，如图 1-23 所示。其中，index.php 文件是网站的首页，在浏览器的地址栏中输入“http://localhost”并回车后，打开的 WampServer 测试页就是该文件的运行结果。

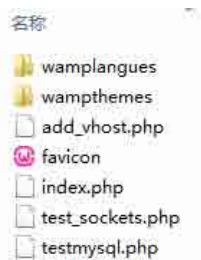


图 1-23 WWW 目录结构

3) 更改 PHP 版本

当前 WampServer 软件版本中内置了 5.6.25 和 7.0.10 两个版本的 PHP 解析器，可用鼠标左键单击软件图标，在弹出的菜单中选择“PHP”，进入二级菜单中后选择“Version”，进入下一级菜单后选择“5.6.25”或“7.0.10”，如图 1-24 所示，即可实现 PHP 在“5.6.25”和“7.0.10”版本之间的切换。



图 1-24 更改 PHP 版本

3. 集成运行环境的基本设置

Apache 没有图形化的服务器配置界面，只能通过修改配置文件进行设置，对 Apache 服务器的所有设置都是通过修改配置文件的代码来实现的。配置文件为纯文本文件，可以使用记事本等软件打开。

1) 修改 WWW 目录为指定目录

不管是学习阶段，还是日后搭建自己的站点进行建设或测试，如果不想将自己的网站放在默认的 WWW 目录下，而希望将个人制作的网站文件与 WWW 目录中的文件进行区分管理的话，可以在其他目录下建立新的站点，并将 localhost 域名指向的网站文件由原来的 WWW 目录所在路径修改为新站点所在路径。例如：可以将 localhost 指向的网站目录从原来的 C:\wamp64\www，修改为新网站所在路径 D:\Test。具体方式如下：

首先使用鼠标左键单击状态栏中的软件图标，在弹出的菜单中选择“Apache”，如图 1-25 所示，在弹出的子菜单中单击“httpd.conf”。



图 1-25 打开 httpd.conf 文件

在用记事本打开的 httpd.conf 文件中，使用 Ctrl+F 组合键打开“查找”对话框，在“查找内容”编辑框中输入“DocumentRoot”，对当前记事本中的第 261 行和第 262 行进行修改。将原文件中的：

```
DocumentRoot "${INSTALL_DIR}/www"  
<Directory "${INSTALL_DIR}/www/">
```

修改为：

```
DocumentRoot "D:/test"  
<Directory "D:/test/">
```

需要注意的是，Windows 下表示路径的“\”在这里必须改为“/”，原 httpd.conf 文件中出现的\${INSTALL_DIR}代表软件的安装目录 C:\wamp64\。

然后，重新使用鼠标左键单击状态栏中的软件图标，在 Apache 菜单的子菜单中选择“httpd-vhosts.conf”。在用记事本打开的 httpd-vhosts.conf 文件中，找到“DocumentRoot”和“Directory”后按照下面的要求进行修改：

将源文件中的：

```
DocumentRoot C:/wamp64/www  
<Directory "C:/wamp64/www/">
```

修改为：

```
DocumentRoot D:/test  
<Directory "D:/test/">
```



最后，使用 Ctrl+S 组合键分别保存对两个文件的上述修改，使用鼠标左键单击软件图标，选择“重新启动所有服务”，使刚才的修改生效。

2) 修改默认首页

当在浏览器的地址栏中输入诸如“http://localhost”的 URL 时，Apache 默认情况下会按照 index.php、index.php3、index.html、index.htm 的优先顺序在当前网站的根目录下查找，如

果 `index.php` 文件不存在，Apache 会尝试查找 `index.php3` 文件，依此类推。当目录下不存在默认文档，且用户仅指定要访问的目录但没有指定要访问目录下的哪个文件时，Apache 以超文本形式返回目录中的文件和子目录列表(虚拟目录不会出现在目录列表中)，如图 1-26 所示。

Index of /

<u>Name</u>	<u>Last modified</u>	<u>Size</u>	<u>Description</u>
 1/	2017-06-08 22:09	-	
 2/	2017-06-08 22:09	-	

Apache/2.4.23 (Win64) PHP/7.0.10 Server at localhost Port 80

图 1-26 以超文本形式显示目录和文件

如果用户想要修改打开首页文件的优先级，或者想要增加新的首页文件，可以使用鼠标左键单击状态栏中的软件图标，选择“Apache”子菜单中的“`httpd.conf`”，在打开的文件中查找“`DirectoryIndex`”，找到第 279 行，如下所示：

```
DirectoryIndex index.php index.php3 index.html index.htm
```

修改时需要注意，如果要修改首页文件，可将 `index.php` 修改为想要的文件名，也可以在 `index.php` 前添加新的文件名。例如，可以增加新的文件名 `default.php` 作为优先级最高的首页，如下所示：

```
DirectoryIndex default.php index.php index.php3 index.html index.htm
```

注意 `default.php` 和 `DirectoryIndex` 之间，以及 `default.php` 与 `index.php` 之间要用英文空格进行分隔，修改后需要保存并重新启动所有服务。

3) 添加虚拟目录

每个站点都有主目录，又称根目录，代表站点的主目录一旦建立，默认情况下主目录中的文件以及所有子目录中的文件都可以被用户访问。一般来说，一个站点的内容应当维护在一个单独的目录下，以免引发访问请求混乱的问题。特殊情况下，网络管理人员可能因为某种需要而使用主目录以外的其他目录，或者使用其他计算机上的目录，让 Internet 用户作为站点访问。对于 Web 服务器来说，将虚拟目录作为主目录的一个子目录来对待，和主目录拥有相同的域名，但实际上这个子目录是不存在的；而对于用户来说，访问时并未感觉到虚拟目录与站点中其他目录之间的区别。将希望使用的目录设为虚拟目录，从而让用户访问。设置虚拟目录时必须指定位置，虚拟目录的实际位置可以在本地服务器上，也可以在远程服务器上。当用户访问的虚拟目录在远程服务器上时，Web 服务器将充当代理的角色，将通过与远程计算机联系并检索用户所请求的文件来实现信息服务支持。

在 Apache 中添加虚拟目录的方式如下：

使用鼠标左键单击状态栏中的软件图标，在弹出的菜单中选择“Apache”，在弹出的子菜单中选择“`httpd.conf`”，在打开的 `httpd.conf` 文件中搜索“`IfModule dir_module`”，找到如下部分代码：

```
<IfModule dir_module>
DirectoryIndex index.php index.php3 index.html index.htm
</IfModule>
```

在这部分代码的下方添加下面的代码：

```
<IfModule dir_module>
DirectoryIndex index.html index.htm index.php
Alias /raid "D:/test/1"
<Directory D:/test/1>
Options All
AllowOverride None
Require all granted
</Directory>
</IfModule>
```

其中“DirectoryIndex”用于设置虚拟目录中首页的显示优先级；“Alias”表示虚拟目录；“/raid”中的 raid 表示虚拟目录的名字；“D:/test/1”表示的是虚拟目录的路径；“<Directory>…</Directory>”部分用于设置虚拟目录的访问权限；“Options All”表示使用所有目录的访问特性；Options 选项用于定义目录使用哪些特性，包括 Indexes、MultiViews 和 ExecCGI 等；“AllowOverride None”表示禁止使用.htaccess 文件，基于安全和效率的原因，虽然可以通过.htaccess 文件来设置目录的访问权限，但应尽可能地避免使用.htaccess 文件；“Require all granted”表示允许所有用户访问，Require 只用于控制访问权限。

WampServer 提供了一种非常方便的添加虚拟目录的方式，操作步骤如下：

(1) 使用鼠标左键单击状态栏中的软件图标，在弹出的菜单中选择“Apache”，在弹出的子菜单中选择“Alias 目录”，再在弹出的子菜单中单击“添加一个 Alias”，出现录入虚拟目录名界面，如图 1-27 所示。

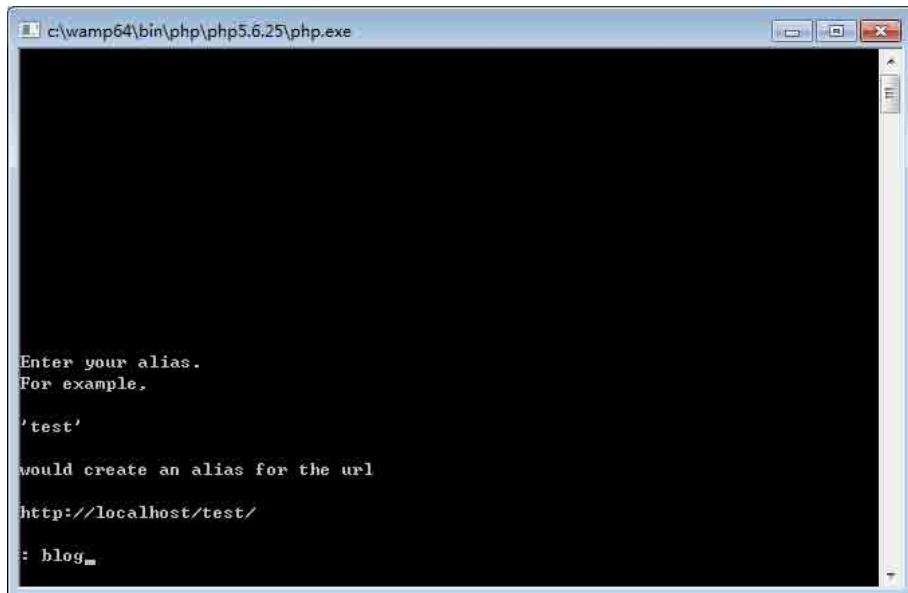


图 1-27 设置虚拟目录名

(2) 在出现的命令行界面中，输入虚拟目录名“blog”后，按回车键进入虚拟目录的路径录入界面，如图 1-28 所示。

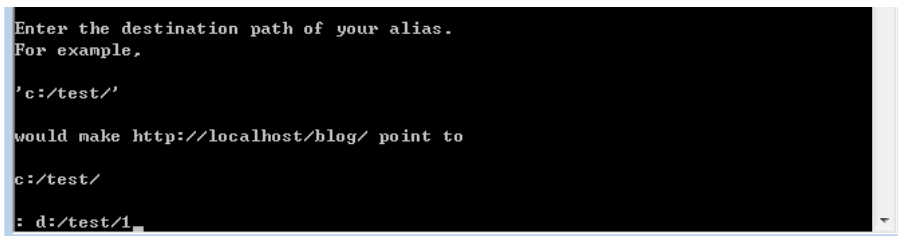


图 1-28 设置虚拟目录的路径

(3) 输入“d:/test/1”后按回车键，则会提示“Alias created. Press Enter to exit ...”。在 Alias 目录中会出现刚才添加的虚拟目录，如图 1-29 所示。



图 1-29 虚拟目录列表

4) 配置虚拟主机

虚拟主机(Virtual Host)是在同一台机器上搭建属于不同域名或基于不同 IP 的多个网站服务的技术，可以为运行在同一台物理机器上的各个网站配置不同的 IP 和端口，也可让多个网站拥有不同的域名。用 WampServer 配置虚拟主机的方式为：单击鼠标左键，在弹出的菜单中选择“Your VirtualHosts”，在弹出的子菜单中会显示已安装的虚拟主机，初始状态下只有 localhost，如图 1-30 所示。



图 1-30 虚拟主机列表

如果不希望使用 localhost 作为域名进行访问，可以通过配置虚拟主机，使用自定义域名的形式访问。Apache 2.4.23 版本默认启用 Apache 的虚拟主机功能，配置虚拟主机时，只需要使用鼠标左键单击状态栏中的软件图标，然后在“Apache”菜单的子菜单中选择“httpd-vhosts.conf”，找到如下代码：

```
<VirtualHost *:80>
  ServerName localhost
  DocumentRoot d:/test
  <Directory "d:/test/">
    Options +Indexes +Includes +FollowSymLinks +MultiViews
    AllowOverride All
    Require local
  </Directory>
</VirtualHost>
```

将第 2 行的 `ServerName localhost` 修改为 `ServerName test.com`，修改后需要保存并重新启动所有服务。这样就可以在“Your VirtualHosts”的子菜单中找到刚才新建的虚拟主机“test.com”。

5) 多域名访问

如果一台主机上存在多个网站，每个网站对应不同的域名，那么需要增加新的虚拟主机，每个虚拟主机对应一个网站，并使用一个独立的域名。可以先在 `httpd-vhosts.conf` 文件中将 `<VirtualHost *:80>...</VirtualHost>` 及其内部所有代码复制，然后在文档结束处粘贴一份。然后分别对其中的 `ServerName`、`DocumentRoot` 和 `Directory` 进行修改，其中 `ServerName` 修改为新的域名，`DocumentRoot` 和 `Directory` 修改为新网站的根目录。最后找到 `c:\windows\system32\drivers\etc` 目录下的 `hosts` 文件，使用记事本打开，在末尾添加下面的代码：

127.0.0.1 新域名

虚拟主机名(也就是域名)要和 `ServerName` 的值保持一致。每增加一个虚拟主机，就需要增加一行这样的代码，并对域名部分进行相应的更新。

为什么要添加这行代码呢？需要先搞清楚浏览器在接受域名访问请求后的工作流程，浏览器在接收到一个域名的访问请求时，会先在本地的 DNS 缓存文件中查找是否存在与该域名对应的 IP(如果用户以前成功访问过该域名，就会在本地的 DNS 缓存文件中存放与该域名对应的服务器 IP 地址)，如果没有与该域名对应的 IP 地址，就访问 DNS 服务器，获得该域名所指向的服务器 IP 地址，然后通过 IP 地址和服务器建立连接。然而这里设置的域名并不一定是真实存在的，而有可能是模拟域名。为了避免上述操作，需要在本地 DNS 的缓存文件中增加这行代码，这样当访问这个模拟域名时，实际访问的是 127.0.0.1 这个本地 IP。使用这种方式就可以为当前目录下的多个站点指定不同的域名。

1.4 第一个 PHP 程序

1.4.1 开发第一个 PHP 网页

在学习了前几节的内容后，现在尝试开发属于自己的简单的 PHP 网站，主要需要以下步骤：

(1) 首先新建一个用于存放网站文件的目录。需要注意的是：在命名新建的网站目录及网页文件时，尽量不要使用中文。例如，在 D 盘根目录新建文件夹 `example`，然后在 `example` 文件夹中新建文件夹 `chap1`。

(2) 使用记事本新建一个 PHP 文件。PHP 文件和 HTML 文件一样都是纯文本文件，因此可以用记事本编辑，只需将后缀名由原来的 `.txt` 修改为 `.php` 即可。例如，可以新建一个名为 `1-1.txt` 的记事本文件，然后将后缀名由 `.txt` 修改为 `.php`，使用鼠标右键单击 `1-1.php`，在弹出的菜单中选择“用记事本打开该文件”命令，在文件中输入以下代码：

```
<?php
    echo 'Hello world!';
?>
```

单击“文件”菜单中的“保存”菜单项，保存输入的代码并关闭该文件，这样只有一个网页的 PHP 网站就建设好了，然后对 Apache 服务器进行设置，以便能通过浏览器访问这个网站。

1.4.2 设置 PHP 网站

可以参考 1.3.2 节中的“修改 WWW 目录为指定目录”部分，将当前网站修改为服务器默认网站；也可以使用“添加虚拟目录”的形式，将当前网站设置为默认网站的二级网站；还可以使用“配置虚拟主机”或“多域名访问”的形式为当前网站设置新的域名。但是由于唯一的网页文件名为 `1-1.PHP`，因此需要修改当前网站的默认首页。

使用“修改 WWW 目录为指定目录”的方式，按照以下步骤设置：

(1) 打开 `httpd.conf` 文件，找到第 261 行和第 262 行，将其中的 `DocumentRoot` 和 `Directory` 由原来的默认网站目录修改为以下代码：

```
DocumentRoot "D:/example/chap1"
<Directory "D:/example/chap1/">
```

(2) 打开 `httpd-vhosts.conf` 文件，找到 `ServerName localhost`，将其中的“`DocumentRoot`”和“`Directory`”的值由原来的默认网站目录修改为以下代码：

```
DocumentRoot D:/example/chap1
<Directory "D:/example/chap1/">
```

(3) 修改网站的默认首页，打开 `httpd.conf` 文件，在其中搜索“`DirectoryIndex`”，找到以下代码：

```
<IfModule dir_module>
    DirectoryIndex index.php index.php3 index.html index.htm
</IfModule>
```

在上述代码的第二行中添加 `1-1.php`，如下所示：

```
DirectoryIndex 1-1.php index.php index.php3 index.html index.htm
```

这样就可以将当前 Apache 服务器管理的所有网站的默认首页修改为 1-1.php，如果只需要修改某个网站的默认首页，那么需要到虚拟目录或虚拟主机部分修改 DirectoryIndex 的默认首页及其顺序。

1.4.3 运行 PHP 网站

打开浏览器，在地址栏中输入“http://localhost/”，按回车键会出现如图 1-31 所示的页面。当看到这个网页中出现的“Hello world!”时，就说明第一个 PHP 网站已经搭建成功了。

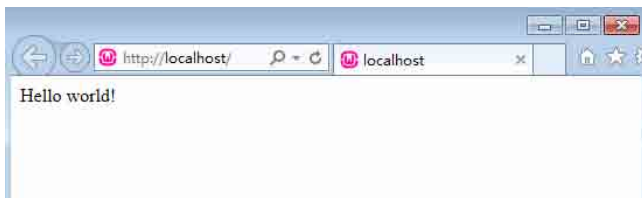


图 1-31 第一个 PHP 页面的运行结果

1-1.php 文件中的代码如下：

```
<?php
    echo 'Hello world!';
?>
```

第一行代码“<?php”和第三行代码“>”两部分联合起来表示 PHP 脚本代码。第二行代码“echo 'Hello world!';”表示将一对单引号中的信息嵌入到网页中。

在运行的网页中使用鼠标右键单击，在弹出的菜单中选择“查看源文件”命令，会出现如图 1-32 所示的内容。输入的 PHP 脚本代码无法在浏览器中查看。

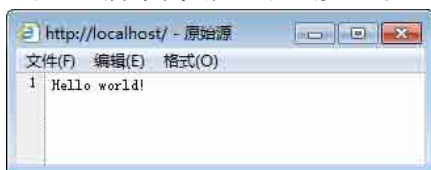


图 1-32 PHP 代码被转换为 HTML 代码

1.5 本章小结

本章首先介绍了什么是 PHP，PHP 的工作流程，PHP 的版本和发展历程、优势及常用工具；其次对 PHP 集成运行环境的安装和相关配置进行了详细介绍；最后演示了第一个 PHP 程序的运行。通过这些内容，使读者对 PHP 程序设计有一个全面的认识。

1.6 思考与练习

一、选择题

- 关于将 PHP 语言嵌入到 HTML 中，以下说法中错误的是()。
 - 可以在两个 HTML 标记对的开始和结束标记中嵌入 PHP
 - 可以在 HTML 标记的属性位置嵌入 PHP
 - HTML 文档中可以嵌入任意多个 PHP 标记
 - PHP 嵌入 HTML 中的标记必须是<?php 和?>
- 下面关于 PHP 服务器的说法中错误的是()。
 - PHP 的运行需要 Web 服务器的参与
 - SQL Server 可以用作 PHP 开发的数据库服务器
 - PHP 预处理器的功能是解释并执行 PHP 代码
 - MySQL 数据库是完全免费的
- 在安装 PHP 之前，首先需要一种()。
 - 文件服务器
 - 信息服务器
 - 数据库服务器
 - Web 服务器
- Apache 的配置文件是()。
 - php.ini
 - apache.ini
 - server.xml
 - httpd.conf
- 如果 Apache 的网站主目录是 D:\shop，并且没有建立任何虚拟目录，那么在浏览器的地址栏中输入 `http://localhost/admin/admin.php`，回车后打开的文件将是()。
 - D:\shop\admin\admin.php
 - D:\localhost\admin\admin.php
 - D:\shop\admin.php
 - D:\shop\localhost\admin\admin.php

二、编程题

- 自己动手练习 PHP 运行环境的搭建。
- 编写程序，在 PHP 页面中显示欢迎信息“欢迎来到 PHP 学习网站!”。