

即用即查

PHP

函数参考手册

陈向辉 编著

排版精准、实例丰富、图文并茂；光盘中提供了索引，查找代码方便

■ 65类901个PHP函数

■ 703个实例源文件



人民邮电出版社
POSTS & TELECOM PRESS



CD-ROM

即用即查

PHP

函数参考手册

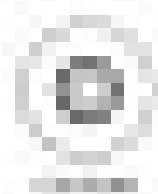
阮一峰 编

第1版 2018年1月第1次印刷 16开 256页 32.00元

ISBN 978-7-113-23456-1

9 787113 234561 >

人民邮电出版社
POST & TELECOM PRESS



即用即查

PHP

函数参考手册

陈向辉 编著

人 民 邮 电 出 版 社

北 京

图书在版编目 (CIP) 数据

即用即查 PHP 函数参考手册 / 陈向辉编著. —北京: 人民邮电出版社, 2008.2

ISBN 978-7-115-17185-6

I. 即… II. 陈… III. PHP 语言—程序设计—技术手册 IV. TP312-62

中国版本图书馆 CIP 数据核字 (2007) 第 176204 号

内 容 提 要

本书采用简洁明了的形式进行编写, 目录非常清晰, 读者可以通过目录迅速找到所需要的函数。另外, 书中实例丰富且简单易懂, 通过实例的演示, 读者能够迅速掌握各个函数的使用方法。

本书按照函数的分类, 详细介绍了与 PHP 有关的各种函数, 包括: PHP 简介、服务器数据库、数组字符、文件目录操作、网络通信、图像处理、Win32 函数、XML 函数、综合函数等, 共 9 篇 35 章。这些章节全面涵盖了 PHP 编程中从字符、数据处理到数据库、网络通信, 从图像处理到 XML 应用等各方面的内容。本书的内容完全可以满足一般编写 PHP 程序的要求。

本书适用于 PHP 初学者、PHP 程序员以及其他 Web 应用程序开发和管理技术人员作为工具手册使用。

即用即查 PHP 函数参考手册

- ◆ 编 著 陈向辉
责任编辑 屈艳莲
- ◆ 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@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
北京铭成印刷有限公司印刷
新华书店总店北京发行所经销
- ◆ 开本: 787×1092 1/16
印张: 41.75
字数: 967 千字
印数: 1—4 000 册
- 2008 年 2 月第 1 版
2008 年 2 月北京第 1 次印刷

ISBN 978-7-115-17185-6/TP

定价: 69.00 元 (附光盘)

读者服务热线: (010) 67132692 印装质量热线: (010) 67129223

反盗版热线: (010) 67171154

前言

您在使用 PHP 的过程中，是不是遇到过下面介绍的问题。

1. 为了实现某个特定功能，苦苦设计了很长的代码，结果，执行效果却并不理想。而别人却能通过短短的一段代码，甚至一个语句来实现，为什么？

解惑：PHP 中有大量的函数，或许您对常用的操作函数比较了解，但您却对一些不太常用的函数不甚了解。其实有些操作完全可以通过 PHP 的内置库函数来解决。如果您了解了这些函数，或者知道有这样的函数，那么，在具体使用时，就可以通过查找手册，直接利用库函数，达到事半功倍的效果。本书详细地介绍了 PHP 中的 8 大类函数，并在相应的函数中结合了大量的实例，一定会给您的学习和工作提供很多帮助。

2. 在编写代码时，对某个方法或者函数有印象，但不记得具体用法了，查找一些参考资料时却不容易找到，怎么办？

解惑：本书在编写过程中，尽可能地涵盖 PHP 的相关内容，书中详细介绍了 PHP 的基本语法和各类函数，对每一个函数或语法都从功能说明、使用范围、函数语法、示例程序等 4 个方面进行介绍，方便用户查阅。

3. PHP 虽然语言结构简单，但其内容太多，学起来很费力，而且看完一本书之后，很难应用于实际中。

解惑：本书内容全面详实，适合于不同阶段的 PHP 用户，初学者可以以此书为依据，逐步学习，逐渐成熟；PHP 程序员则可以把本书作为函数字典，以快速查找不经常使用的函数。另外，本书所举实例简洁明了、贴近实际，通过这些实例的学习，读者可以在学习的过程中培养实际的应用能力，将所学内容立即应用到工作实践中。

为什么要写这本书？突出速查

笔者从事网站开发和网页设计多年，曾经参加过几个网站的设计团队，参与或独立开发过很多大型网站。在开发过程中笔者发现，很多开发人员对 PHP 的基础知识掌握不扎实、不全面，在编写代码的过程中经常要查阅一些资料，而且要翻阅多本参考书才能解决。为了帮助广大读者提高开发和管理 Web 站点的水平，笔者精心编著了本书。本书依照读者的学习规律，首先介绍 PHP 的基本概念，在掌握了这些基本概念和基本操作的基础上，再对内容进行深入的讲解，严格遵循由浅入深、循序渐进的原则，既注重基础知识，又注重实际应用，同时也将方便用户查阅作为排版的基本要求，因此本书是 PHP 用户案头必备的参考手册。

这本书有什么与众不同之处？如何使用本书

目前市场上关于 PHP 的书很多，但是，这些书往往侧重于某一方面，内容不全，或者是实例不足，不便于读者理解和应用。本书力求内容全面、讲解详细，所举实例具体到某一个函数和方法，总结起来有如下特点。

1. 内容全面，实例丰富：本书全面介绍了用 PHP 进行网络编程时所涉及的 8 大类 34 小类函数。在介绍这些内容的同时结合了大量的实例，通过这些实例能够帮助读者更好地理解本书所讲述的内容。

2. 贴近实际，注重实用：本书所用实例除了和所讲内容密切相关以外，在编写思路上力求贴近工程实际。

3. 简洁明了，通俗易懂：在本书的三级目录中，都以简洁明了的语言概括了所讲函数的功能，能够帮助读者快速地把握所讲函数的主旨，有助于提高学习效率。

本书包括哪些内容？

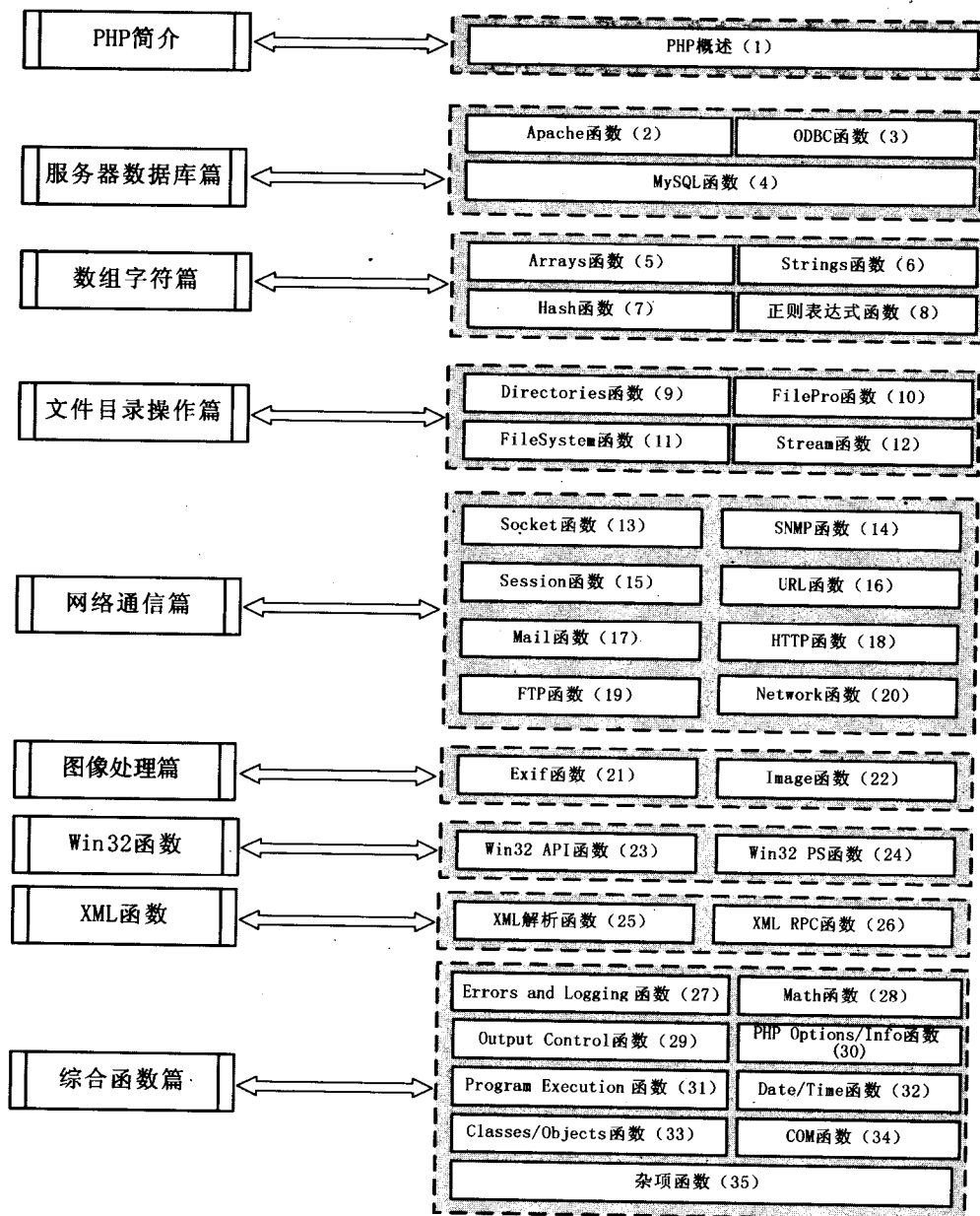
本书分 9 篇共 35 章，分层次地介绍了 PHP 的相关知识，各篇内容如下图所示。

本书约定：

为了便于读者区分函数在不同 PHP 版本中的使用，本书使用版本号标识函数的适用范围。以下是几个常见形式的说明。

- PHP 3、PHP 4、PHP 5：表示该函数适合于 PHP 3、PHP 4、PHP 5 的所有版本。
- PHP 3>=3.0.6、PHP 4、PHP 5：表示该函数适合于高于 PHP 3.0.6 的所有版本。
- PHP 4、PHP 5：表示该函数适合于 PHP 4 及 PHP 5 的所有版本。

- PHP 4>4.3.0、PHP 5: 表示该函数适合于高于 PHP 4.3.0 的所有版本。
- PHP 5: 表示该函数仅适合于 PHP 5 及以后的版本。



本书适合的读者:

- 从事 Web 应用开发的人员;
- PHP 程序开发人员;
- 网页设计与制作人员;
- 网页制作爱好者;
- 大中专院校学生。

本书由陈向辉编写，同时参与编写的还有王石、熊英、陈杰、陈冠军、付京周、袁福庆、张杰、赵显琮、卜庆龄、常利、冯曼菲、匡妍娜、雷成建、李小波、刘浩然、刘会神、王晓悦、马震、齐志华、舒军、孙大林、孙佳楠、王辉、王沛等人，在此一并表示感谢。由于编著时间仓促，书中不足之处在所难免，恳请广大读者批评指正。

编 者

2007 年 12 月

目录

第1篇 PHP 简介

第1章 PHP 基础知识	3
1.1 PHP 概述	3
1.2 PHP 开发环境的搭建	4
1.2.1 Web 服务器 Apache 的安装	4
1.2.2 PHP 解释器的安装	6
1.2.3 MySQL 数据库的安装	7
1.3 第一个 PHP 程序	10
1.3.1 HELLO WORLD	10
1.3.2 测试 MySQL 数据库是否成功安装	10
1.4 小结	11

第2篇 服务器数据库篇

第2章 Apache 函数	15
2.1 Apache 信息获取类函数	15
2.1.1 apache_child_terminate 函数：在本次请求结束后终止 Apache 进程	15
2.1.2 apache_get_modules 函数：获取 Apache 的模块列表	16
2.1.3 apache_get_version 函数：获取 Apache 的版本	16
2.1.4 apache_getenv 函数：获取 Apache 的环境变量	17
2.1.5 apache_lookup_uri 函数：获取特定的 URL 有关信息	17
2.1.6 apache_note 函数：获取或设置 Apache 请求记录	18
2.1.7 apache_request_headers 函数：获取 HTTP 请求的标头	19
2.1.8 apache_response_headers 函数：获取所有 HTTP 回应标头	19
2.1.9 getallheaders 函数：获取所有 HTTP 变数值	20
2.2 Apache 配置类函数	20
2.2.1 apache_reset_timeout 函数：重设 Apache 写操作时间	21
2.2.2 apache_setenv 函数：设置 Apache 的环境变量	21
2.2.3 virtual 函数：执行 Apache 的子请求	22
2.3 Apache 编码类函数	22

2.3.1	ascii2ebcdic 函数：将 ASCII 码转换为 EBCDIC 码	22
2.3.2	ebcdic2ascii 函数：将 EBCDIC 码转换为 ASCII 码	23
2.4	小结	23
第 3 章	ODBC 函数	24
3.1	ODBC 连接类函数	25
3.1.1	odbc_connect 函数：打开一个 ODBC 连接	25
3.1.2	odbc_close 函数：关闭一个已经打开的 ODBC 连接	26
3.1.3	odbc_close_all 函数：关闭所有已经打开的 ODBC 连接	27
3.1.4	odbc_pconnect 函数：打开一个持续有效的 ODBC 连接	27
3.2	ODBC 操作类函数	28
3.2.1	odbc_commit 函数：更新所有处于未决状态的操作	28
3.2.2	odbc_do 函数：在打开的 ODBC 连接上执行 SQL 语句	29
3.2.3	odbc_exec 函数：执行 SQL 语句	30
3.2.4	odbc_execute 函数：执行一个预置的 SQL 语句	30
3.2.5	odbc_free_result 函数：释放传回资料所占用的内存	31
3.2.6	odbc_prepare 函数：预置 SQL 语句的执行	32
3.2.7	odbc_rollback 函数：撤销所有处于未决状态的操作	32
3.3	ODBC 信息获取类函数	33
3.3.1	odbc_columnprivileges 函数：列出给定表的列和相关的权限	33
3.3.2	odbc_columns 函数：列出指定表的列的名称	34
3.3.3	odbc_cursor 函数：获取光标的名称	35
3.3.4	odbc_data_source 函数：返回连接数据库的信息	35
3.3.5	odbc_error 函数：获取最后的错误代码	36
3.3.6	odbc_errormsg 函数：获取最后的错误信息	36
3.3.7	odbc_fetch_array 函数：获取结果集数组	37
3.3.8	odbc_fetch_into 函数：获取传回的指定列	38
3.3.9	odbc_fetch_object 函数：返回结果集到对象	38
3.3.10	odbc_fetch_row 函数：获取传回的一列	39
3.3.11	odbc_field_len 函数：获取字段的长度	40
3.3.12	odbc_field_name 函数：获取字段的名称	41
3.3.13	odbc_field_num 函数：获取字段的序号	41
3.3.14	odbc_field_precision 函数：获取字段的长度	42
3.3.15	odbc_field_scale 函数：获取字段的浮点数	42
3.3.16	odbc_field_type 函数：获取字段的资料类型	43
3.3.17	odbc_foreignkeys 函数：返回特定表的外来键	44

3.3.18	odbc_gettypeinfo 函数: 返回数据库的类型信息	44
3.3.19	odbc_longreadlen 函数: 设定传回栏的最大值	45
3.3.20	odbc_num_fields 函数: 获取字段数目	45
3.3.21	odbc_num_rows 函数: 获取传回的列数目	45
3.3.22	odbc_primarykeys 函数: 返回列的名字作为表的主键	46
3.3.23	odbc_procedurecolumns 函数: 返回检索过程的参数信息	46
3.3.24	odbc_procedures 函数: 获取存在于特定数据源中的进程信息	47
3.3.25	odbc_result_all 函数: 传回 HTML 表格信息	47
3.3.26	odbc_result 函数: 获取结果数据	47
3.3.27	odbc_specialcolumns 函数: 返回一个表中在传送更新时可以自动更新的列	48
3.3.28	odbc_statistics 函数: 获取表的状态及其索引	49
3.3.29	odbc_tableprivileges 函数: 列出表格和每个表格关联的权限	49
3.3.30	odbc_tables 函数: 获取特定数据库上的表的名称	50
3.4	ODBC 设置类函数	50
3.4.1	odbc_autocommit 函数: 开启或关闭自动更新	50
3.4.2	odbc_binmode 函数: 设定二进制的数据处理方式	51
3.4.3	odbc_next_result 函数: 检查下一个结果集是否可用	52
3.4.4	odbc_setoption 函数: 调整 ODBC 设定	52
3.5	小结	53

第 4 章 MySQL 函数 54

4.1	MySQL 连接类函数	54
4.1.1	mysql_connect 函数: 打开一个到 MySQL 服务器的连接	54
4.1.2	mysql_close 函数: 关闭 MySQL 连接	55
4.1.3	mysql_pconnect 函数: 打开一个到 MySQL 服务器的持久链接	56
4.2	MySQL 操作类函数	57
4.2.1	mysql_create_db 函数: 新建一个 MySQL 数据库	57
4.2.2	mysql_data_seek 函数: 移动内部结果的指针	57
4.2.3	mysql_db_query 函数: 发送一条 MySQL 查询	58
4.2.4	mysql_drop_db 函数: 删除一个 MySQL 数据库	59
4.2.5	mysql_escape_string 函数: 转义一个用来查询的字符串	59
4.2.6	mysql_free_result 函数: 释放结果内存	60
4.2.7	mysql_list_dbs 函数: 列出 MySQL 服务器中所有的数据库	60
4.2.8	mysql_list_fields 函数: 列出 MySQL 结果中的字段	61
4.2.9	mysql_list_processes 函数: 列出 MySQL 进程	61
4.2.10	mysql_list_tables 函数: 列出 MySQL 数据库中的表	62
4.2.11	mysql_ping 函数: Ping 一个服务器连接, 如果没有连接则重新连接	63

- 4.2.12 `mysql_query` 函数: 发送一条 MySQL 查询 ———— 63
- 4.2.13 `mysql_real_escape_string` 函数: 转义 SQL 语句中的特殊字符 ———— 64
- 4.2.14 `mysql_result` 函数: 获取结果数据 ———— 65
- 4.2.15 `mysql_select_db` 函数: 选择 MySQL 数据库 ———— 66
- 4.2.16 `mysql_unbuffered_query` 函数: 向 MySQL 发送一条 SQL 查询, 并不获取和缓存结果的行 ———— 67
- 4.3 **MySQL 信息获取类函数** ———— 67
 - 4.3.1 `mysql_affected_rows` 函数: 获取前一次 MySQL 操作所影响的记录行数 ———— 68
 - 4.3.2 `mysql_client_encoding` 函数: 获取字符集的名称 ———— 68
 - 4.3.3 `mysql_db_name` 函数: 获取数据库名 ———— 69
 - 4.3.4 `mysql_errno` 函数: 返回上一个 MySQL 操作中的错误信息的数字编码 ———— 70
 - 4.3.5 `mysql_error` 函数: 返回上一个 MySQL 操作产生的文本错误信息 ———— 70
 - 4.3.6 `mysql_fetch_array` 函数: 从结果集中获取一行作为关联数组, 或数字数组, 或二者都有 ———— 71
 - 4.3.7 `mysql_fetch_assoc` 函数: 从结果集中获取一行作为关联数组 ———— 72
 - 4.3.8 `mysql_fetch_field` 函数: 从结果集中获取列信息并作为对象返回 ———— 73
 - 4.3.9 `mysql_fetch_lengths` 函数: 获取结果集中每个输出的长度 ———— 75
 - 4.3.10 `mysql_fetch_object` 函数: 从结果集中获取一行作为对象 ———— 76
 - 4.3.11 `mysql_fetch_row` 函数: 从结果集中获取一行作为枚举数组 ———— 77
 - 4.3.12 `mysql_field_flags` 函数: 从结果集中获取字段关联的标志 ———— 77
 - 4.3.13 `mysql_field_len` 函数: 返回指定字段的长度 ———— 78
 - 4.3.14 `mysql_field_name` 函数: 获取结果集中指定字段的字段名 ———— 79
 - 4.3.15 `mysql_field_seek` 函数: 将结果集中的指针设定为指定的字段偏移量 ———— 80
 - 4.3.16 `mysql_field_table` 函数: 获取指定字段所在的表名 ———— 81
 - 4.3.17 `mysql_field_type` 函数: 获取结果集中指定字段的类型 ———— 81
 - 4.3.18 `mysql_get_client_info` 函数: 获取 MySQL 客户端信息 ———— 82
 - 4.3.19 `mysql_get_host_info` 函数: 获取 MySQL 主机信息 ———— 83
 - 4.3.20 `mysql_get_proto_info` 函数: 获取 MySQL 协议信息 ———— 83
 - 4.3.21 `mysql_get_server_info` 函数: 获取 MySQL 服务器信息 ———— 84
 - 4.3.22 `mysql_info` 函数: 获取最近一条查询的信息 ———— 84
 - 4.3.23 `mysql_insert_id` 函数: 获取上一个 INSERT 操作产生的 ID ———— 85
 - 4.3.24 `mysql_num_fields` 函数: 获取结果集中字段的数目 ———— 86
 - 4.3.25 `mysql_num_rows` 函数: 获取结果集中行的数目 ———— 86
 - 4.3.26 `mysql_stat` 函数: 获取当前系统状态 ———— 87
 - 4.3.27 `mysql_tablename` 函数: 获取表名 ———— 88
 - 4.3.28 `mysql_thread_id` 函数: 获取当前线程的 ID ———— 88
- 4.4 **小结** ———— 89

第3篇 数组字符篇

第5章 Arrays 函数 ———— 93

5.1 数组建立类函数 ———— 93

- 5.1.1 array 函数：新建一个数组 ———— 93
- 5.1.2 array_combine 函数：用两个数组的值作为新数组的键与值来新建一个数组 ———— 94
- 5.1.3 range 函数：建立一个包含指定范围单元的数组 ———— 94

5.2 数组操作类函数 ———— 95

- 5.2.1 array_chunk 函数：将一个数组分割成多个数组 ———— 95
- 5.2.2 array_count_values 函数：统计数组中所有的值出现的次数 ———— 97
- 5.2.3 array_diff_assoc 函数：带索引检查计算数组的差集 ———— 97
- 5.2.4 array_diff_key 函数：使用键名比较计算数组的差集 ———— 98
- 5.2.5 array_diff_uassoc 函数：用回调函数做索引检查来计算数组的差集 ———— 98
- 5.2.6 array_diff_ukey 函数：用回调函数对键名比较计算数组的差集 ———— 99
- 5.2.7 array_diff 函数：计算数组的差集 ———— 100
- 5.2.8 array_fill 函数：用给定的值填充数组 ———— 100
- 5.2.9 array_filter 函数：用回调函数过滤数组中的单元 ———— 101
- 5.2.10 array_flip 函数：交换数组中的键和值 ———— 102
- 5.2.11 array_intersect_assoc 函数：带索引检查计算数组的交集 ———— 103
- 5.2.12 array_intersect_key 函数：使用键名比较计算数组的交集 ———— 103
- 5.2.13 array_intersect_uassoc 函数：带索引检查计算数组的交集，并用回调函数比较索引 ———— 104
- 5.2.14 array_intersect_ukey 函数：用回调函数比较键名来计算数组的交集 ———— 104
- 5.2.15 array_intersect 函数：计算数组的交集 ———— 105
- 5.2.16 array_key_exists 函数：检查给定的键名或索引是否存在于数组中 ———— 106
- 5.2.17 array_map 函数：将回调函数作用到给定数组的单元上 ———— 106
- 5.2.18 array_merge_recursive 函数：递归地合并一个或多个数组 ———— 107
- 5.2.19 array_merge 函数：合并一个或多个数组 ———— 108
- 5.2.20 array_multisort 函数：对多个数组或多维数组进行排序 ———— 108
- 5.2.21 array_pad 函数：用值将数组填补到指定长度 ———— 109
- 5.2.22 array_pop 函数：将数组的最后一个单元弹出 ———— 110
- 5.2.23 array_push 函数：将一个或多个单元压入数组的末尾 ———— 111
- 5.2.24 array_rand 函数：从数组中随机取出一个或多个单元 ———— 111
- 5.2.25 array_reduce 函数：用回调函数迭代地将数组简化为单一的值 ———— 112
- 5.2.26 array_search 函数：在数组中查找给定的值并返回相应的键名 ———— 113
- 5.2.27 array_shift 函数：移除数组开头的单元 ———— 114
- 5.2.28 array_unique 函数：移除数组中重复的值 ———— 114

- 5.2.29 array_unshift 函数: 在数组开头插入一个或多个单元 ———— 115
- 5.2.30 array_slice 函数: 从数组中取出一段 ———— 115
- 5.2.31 array_splice 函数: 把数组中的一部分去掉并用其他值取代 ———— 116
- 5.2.32 array_sum 函数: 计算数组中所有值的和 ———— 118
- 5.2.33 array_uintersect_assoc 函数: 带索引检查计算数组的交集并用回调函数比较数据 ———— 118
- 5.2.34 array_uintersect_uassoc 函数: 带索引检查计算数组的交集并用回调函数比较数据和索引 ———— 119
- 5.2.35 array_uintersect 函数: 计算数组的交集并用回调函数比较数据 ———— 119
- 5.2.36 array_walk_recursive 函数: 对数组中的每个成员递归地应用用户函数 ———— 120
- 5.2.37 array_walk 函数: 对数组中的每个成员应用用户函数 ———— 121
- 5.2.38 compact 函数: 用变量名和值建立一个数组 ———— 121
- 5.2.39 count 函数: 计算数组中的单元数目或对象中的属性个数 ———— 122
- 5.2.40 current 函数: 返回数组中的当前单元 ———— 123
- 5.2.41 extract 函数: 从数组中将变量导入当前的符号表中 ———— 124
- 5.2.42 in_array 函数: 查找数组中是否存在某个值 ———— 125
- 5.2.43 key 函数: 从关联数组中获取键名 ———— 125
- 5.2.44 list 函数: 把数组中的值赋给一些变量 ———— 126
- 5.2.45 next 函数: 将数组中的内部指针向前移动一位 ———— 127
- 5.2.46 prev 函数: 将数组中的内部指针倒回一位 ———— 127
- 5.2.47 reset 函数: 将数组中的内部指针指向第一个单元 ———— 128
- 5.2.48 shuffle 函数: 打乱一个数组 ———— 128
- 5.3 数组信息获取类函数 ———— 129**
- 5.3.1 array_change_key_case 函数: 返回字符串键名全为小写或大写的数组 ———— 129
- 5.3.2 array_keys 函数: 返回数组中所有的键名 ———— 130
- 5.3.3 array_reverse 函数: 返回一个单元顺序相反的数组 ———— 130
- 5.3.4 array_values 函数: 返回数组中所有的值 ———— 131
- 5.3.5 each 函数: 返回数组中当前的键/值对并将数组指针向前移动一步 ———— 132
- 5.4 数组比较类函数 ———— 132**
- 5.4.1 arsort 函数: 对数组进行逆向排序并保持索引关系 ———— 133
- 5.4.2 asort 函数: 对数组进行排序并保持索引关系 ———— 133
- 5.4.3 krsort 函数: 对数组按照键名逆向排序 ———— 134
- 5.4.4 ksort 函数: 对数组按照键名排序 ———— 135
- 5.4.5 natcasesort 函数: 用自然排序算法对数组进行不区分大小写字母的排序 ———— 135
- 5.4.6 natsort 函数: 用自然排序算法对数组排序 ———— 136
- 5.4.7 rsort 函数: 对数组逆向排序 ———— 137
- 5.4.8 sort 函数: 对数组排序 ———— 137
- 5.4.9 uasort 函数: 使用用户自定义的比较函数对数组中的值进行排序并保持索引关联 ———— 138
- 5.4.10 uksort 函数: 使用用户自定义的比较函数对数组中的键名进行排序 ———— 139

5.4.11 usort 函数: 使用用户自定义的比较函数对数组中的值进行排序	140
5.5 小结	140

第 6 章 Strings 函数 142

6.1 字符串转换类函数 142

6.1.1 addslashes 函数: 以 C 语言风格使用反斜线转义字符串中的字符	142
6.1.2 addslashes 函数: 使用反斜线引用字符串	143
6.1.3 chop 函数: 清除字符串中的连续空格	143
6.1.4 get_html_translation_table 函数: 返回 htmlspecialchars()函数和 htmlentities()函数的转换表	144
6.1.5 chunk_split 函数: 将字符串分割成小块	146
6.1.6 hebrew 函数: 转换希伯来逻辑字符为可见字符	146
6.1.7 hebrevc 函数: 转换希伯来语文本为可见文本, 包括换行符	147
6.1.8 html_entity_decode 函数: 转换 HTML 字符编码为字符	147
6.1.9 htmlentities 函数: 转换字符为 HTML 字符编码	148
6.1.10 htmlspecialchars_decode 函数: 转换特殊 HTML 字符编码为字符	149
6.1.11 htmlspecialchars 函数: 转换特殊字符为 HTML 字符编码	149
6.1.12 nl2br 函数: 将换行字符转换成 " "	150
6.1.13 quotemeta 函数: 加入引用符	150
6.1.14 rtrim 函数: 清除字符串末尾的空白 (或其他字符)	151
6.1.15 strip_tags 函数: 清除 HTML 及 PHP 的标记	151
6.1.16 stripslashes 函数: 将用 addslashes()函数处理后的字符串原样返回	152
6.1.17 stripslashes 函数: 清除字符串中的反斜线	152
6.1.18 strtolower 函数: 将字符串中的字母全部转换为小写	153
6.1.19 strtoupper 函数: 将字符串中的字母全部转换为大写	153
6.1.20 trim 函数: 截取字符串首尾的空格 (或其他字符)	154
6.2 字符串操作类函数	154
6.2.1 explode 函数: 将字符串按指定字符切开	154
6.2.2 str_pad 函数: 用一个字符串填充另一个字符串到指定长度	155
6.2.3 str_split 函数: 把字符串转化为数组	156
6.2.4 str_shuffle 函数: 随机打乱字符串中的字符顺序	157
6.2.5 str_ireplace 函数: 将某个子字符串替换为另一个字符串	157
6.2.6 localeconv 函数: 获取当地的数字和货币信息	158
6.2.7 ltrim 函数: 清除字符串左端的连续空白 (或者其他字符)	159
6.2.8 money_format 函数: 把数字字符串转换为货币表示形式	160
6.2.9 nl_langinfo 函数: 获取语言和地区信息	160
6.2.10 number_format 函数: 用千位分隔符格式化数字字符串	162
6.2.11 parse_str 函数: 将字符串解析为变量	163

- 6.2.12 setlocale 函数: 设置地区信息 ———— 163
- 6.2.13 similar_text 函数: 计算两个字符串中相匹配的字符串数量 ———— 164
- 6.2.14 str_repeat 函数: 重复使用指定字符串 ———— 165
- 6.2.15 str_replace 函数: 区分大小写的字符串替换 ———— 165
- 6.2.16 str_word_count 函数: 返回字符串中的单词数 ———— 166
- 6.2.17 strcmp 函数: 对两个字符串进行比较 ———— 166
- 6.2.18 strlen 函数: 统计字符串的长度 ———— 167
- 6.2.19 strnatcmp 函数: 使用自然顺序演算法来对比字符串 ———— 167
- 6.2.20 strrev 函数: 颠倒字符串 ———— 168
- 6.2.21 strtok 函数: 切开字符串 ———— 168
- 6.2.22 strstr 函数: 转换字符串的某些字符 ———— 169
- 6.2.23 substr_count 函数: 计算一个字符串在另一个字符串中出现的次数 ———— 170
- 6.2.24 substr_replace 函数: 替换字符串中的一部分为另一个字符串 ———— 170
- 6.2.25 substr 函数: 截取字符串 ———— 171
- 6.2.26 ucfirst 函数: 将字符串中第一个字符改为大写 ———— 171
- 6.2.27 ucwords 函数: 将字符串中每个单词的第一个字母改为大写 ———— 172
- 6.2.28 wordwrap 函数: 将字符串按字符个数换行 ———— 172
- 6.3 字符串查找类函数 ———— 173
 - 6.3.1 chr 函数: 将指定的序数转化为相应的 ASCII 码字符 ———— 173
 - 6.3.2 implode 函数: 将数组合并为字符串 ———— 173
 - 6.3.3 join 函数: 将数组转化为字符串 ———— 174
 - 6.3.4 crc32 函数: 计算一个字符串的 crc32 多项式 ———— 174
 - 6.3.5 strchr 函数: 返回字符串在另一个字符串中首次出现的位置到末尾的子字符串 ———— 175
 - 6.3.6 strcspn 函数: 返回字符串中起首不符合 mask 的子字符串长度 ———— 175
 - 6.3.7 strpos 函数: 查找一个字符串在另一个字符串中首次出现的位置(不区分大小写) ———— 176
 - 6.3.8 stristr 函数: 返回一个字符串在另一个字符串中首次出现的位置到后者末尾的子字符串(不区分大小写) ———— 176
 - 6.3.9 stripos 函数: 不区分大小写查找字符在字符串中首次出现的位置 ———— 177
 - 6.3.10 strrpos 函数: 查找字符在字符串中最后出现的位置 ———— 177
 - 6.3.11 strspn 函数: 返回字符串中起首符合 mask 的子字符串长度 ———— 178
 - 6.3.12 strstr 函数: 返回字符串中某字符串开始处至结束的字符串 ———— 178
- 6.4 字符串编码类函数 ———— 178
 - 6.4.1 bin2hex 函数: 将二进制字符转化为十六进制 ———— 179
 - 6.4.2 convert_cyr_string 函数: 将字符由一种 Cyrillic 字符转换成另一种 ———— 179
 - 6.4.3 convert_uudecode 函数: 对用 uuencode 算法编码的字符串解码 ———— 180
 - 6.4.4 convert_uuencode 函数: 对字符串进行 uuencode 算法编码 ———— 180
 - 6.4.5 count_chars 函数: 返回字符串中所有字符的信息 ———— 180
 - 6.4.6 ord 函数: 返回一个字符的 ASCII 码 ———— 181

6.4.7	quoted_printable_decode 函数: 将 qp 编码字符串转换成 8 位元字符串	182
6.4.8	str_rot13 函数: 对一个字符串进行 rot13 编码	182
6.5	字符串加密类函数	183
6.5.1	crypt 函数: 将字符串用 DES 编码加密	183
6.5.2	md5_file 函数: 计算给定文件的 MD5hash	183
6.5.3	md5 函数: 对一个字符串进行 MD5 加密	184
6.5.4	sha1_file 函数: 计算给定文件的 sha1hash	184
6.5.5	sha1 函数: 计算字符串的 sha1hash	185
6.6	字符串输出类函数	185
6.6.1	echo 函数: 输出一个或多个字符	186
6.6.2	fprintf 函数: 输出格式化字符串到流	186
6.6.3	print 函数: 输出格式化字符串	187
6.6.4	printf 函数: 输出一个格式化的字符串	187
6.6.5	sprintf 函数: 返回一个格式化的字符串到变量	188
6.6.6	sscanf 函数: 按照一定格式解析输入的字符串	188
6.6.7	vfprintf 函数: 输出格式化字符串到流	189
6.6.8	vprintf 函数: 输出一个格式化的字符串	189
6.6.9	vsprintf 函数: 输出格式化的字符串到变量	190
6.7	字符串比较类函数	190
6.7.1	levenshtein 函数: 计算两个字符串的 Levenshtein 距离	191
6.7.2	metaphone 函数: 计算字符串的 metaphone 键	191
6.7.3	soundex 函数: 计算字符串的 soundex 键	192
6.7.4	strcoll 函数: 对两个字符串进行比较	193
6.7.5	strcmp 函数: 比较两个字符串	193
6.7.6	strnatcasecmp 函数: 用自然运算法则比较字符串 (不区分大小写)	194
6.7.7	strncasecmp 函数: 比较字符串的前 n 个字符 (不区分大小写)	194
6.7.8	strncmp 函数: 比较两个字符串的前 n 个字符	195
6.7.9	strpbrk 函数: 查找字符串中的一系列字符	195
6.7.10	strpos 函数: 寻找字符串中某字符最先出现的位置	196
6.7.11	strrchr 函数: 查找一个字符串在另一个字符串中末次出现的位置, 并返回从字符串中的这个位置起, 一直到字符串结尾的子字符串	196
6.7.12	substr_compare 函数: 从指定起始位置起在一定长度内比较两个字符串 (二进制精确, 区分大小写可选)	197
6.8	小结	197
第 7 章 Hash 函数 198		
7.1	PHP 中的常用 Hash 函数	198
7.1.1	hash_algos 函数: 返回已经注册的 Hash 规则列表	198