

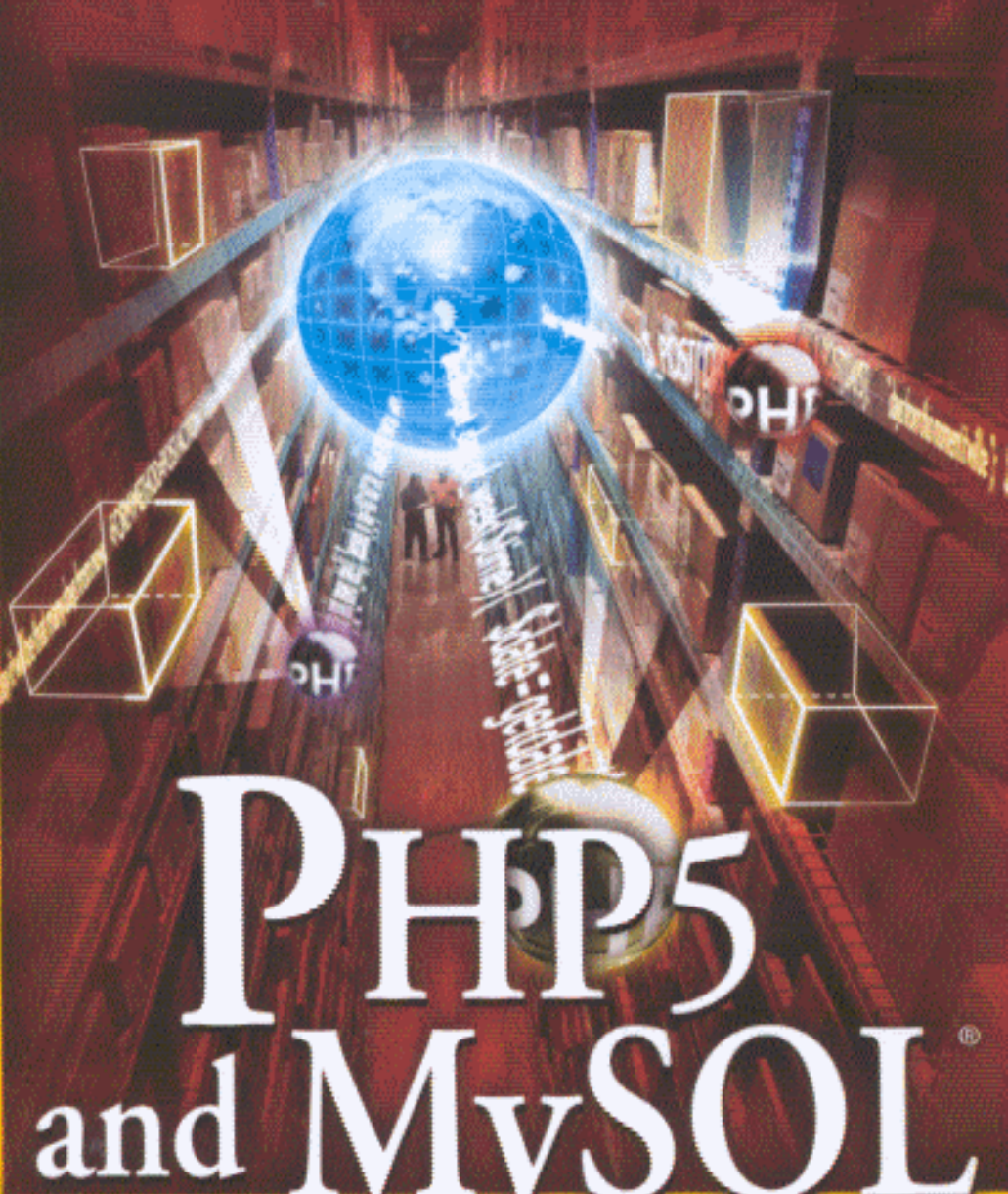
# 100%

ONE HUNDRED PERCENT  
COMPREHENSIVE  
AUTHORITATIVE  
WHAT YOU NEED  
ONE HUNDRED PERCENT

**Activate** this powerhouse pair of tools and start building dynamic, database-backed Web pages

**Banish** installation and development problems with expert troubleshooting advice

**Explore** exceptions and error handling, debugging, PEAR, security, PostgreSQL, and more



# PHP5 and MySQL<sup>®</sup> Bible

**COMPANION  
WEB SITE**

Includes all code listings for  
downloading

Tim Converse and Joyce Park with Clark Morgan

# Contents

Preface . . . . . ix

Acknowledgments . . . . . xv

## Part I: PHP: The Basics 1

### Chapter 1: Why PHP and MySQL? 3

What Is PHP? . . . . . 3

What Is MySQL? . . . . . 4

The History of PHP . . . . . 4

The History of MySQL . . . . . 5

Reasons to Love PHP and MySQL . . . . . 6

- Cost . . . . . 6
- Ease of Use . . . . . 8
- HTML-embeddedness . . . . . 9
- Cross-platform compatibility . . . . . 11
- Not tag-based . . . . . 11
- Stability . . . . . 12
- Speed . . . . . 12
- Open source licensing . . . . . 13
- Many extensions . . . . . 14
- Fast feature development . . . . . 14
- Popularity . . . . . 15
- Not proprietary . . . . . 16
- Strong user communities . . . . . 17

Summary . . . . . 17

### Chapter 2: Server-Side Web Scripting 19

Static HTML . . . . . 19

Client-Side Technologies . . . . . 22

Server-Side Scripting . . . . . 26

What Is Server-Side Scripting Good for? . . . . . 32

Summary . . . . . 33

### Chapter 3: Getting Started with PHP 35

Hosting versus DIY . . . . . 35

- The ISP option . . . . . 35
- The self-hosting option: Pros and cons . . . . . 38
- Compromise solutions . . . . . 39

Installing PHP	39
Before you can begin	40
Installation procedures	41
Development tools	47
Summary	50

## **Chapter 4: Adding PHP to HTML** . . . . . **53**

Your HTML Is Already PHP-Compliant!	53
Escaping from HTML	53
Canonical PHP tags	54
Short-open (SGML-style) tags	54
Hello World	55
Jumping in and out of PHP mode	56
Including files	57
Summary	59

## **Chapter 5: Syntax and Variables** . . . . . **61**

PHP Is Forgiving	61
HTML Is Not PHP	61
PHP's Syntax Is C-Like	62
PHP is whitespace insensitive	62
PHP is sometimes case sensitive	62
Statements are expressions terminated by semicolons	63
Braces make blocks	65
Comments	66
C-style multiline comments	66
Single-line comments: # and //	66
Variables	67
PHP variables are Perl-like	67
Declaring variables (or not)	67
Assigning variables	67
Reassigning variables	68
Unassigned variables	68
Variable scope	69
You can switch modes if you want	70
Constants	70
Types in PHP: Don't Worry, Be Happy	71
No variable type declarations	71
Automatic type conversion	71
Types assigned by context	71
Type Summary	72
The Simple Types	72
Integers	73
Doubles	73
Booleans	75
NULL	76
Strings	77
Output	80
Echo and print	80
Variables and strings	81
Summary	82

**Chapter 6: Control and Functions . . . . . 83**

- Boolean Expressions . . . . . 84
  - Boolean constants . . . . . 84
  - Logical operators . . . . . 84
  - Comparison operators . . . . . 86
  - The ternary operator . . . . . 87
- Branching . . . . . 88
  - If-else . . . . . 89
  - Switch . . . . . 92
- Looping . . . . . 94
  - Bounded loops versus unbounded loops . . . . . 94
  - While . . . . . 94
  - Do-while . . . . . 95
  - For . . . . . 95
  - Looping examples . . . . . 96
  - Break and continue . . . . . 99
  - A note on infinite loops . . . . . 101
- Alternate Control Syntaxes . . . . . 101
- Terminating Execution . . . . . 102
- Using Functions . . . . . 104
  - Return values versus side effects . . . . . 105
- Function Documentation . . . . . 105
  - Headers in documentation . . . . . 106
  - Finding function documentation . . . . . 106
- Defining Your Own Functions . . . . . 107
  - What is a function? . . . . . 107
  - Function definition syntax . . . . . 107
  - Function definition example . . . . . 108
  - Formal parameters versus actual parameters . . . . . 109
  - Argument number mismatches . . . . . 109
- Functions and Variable Scope . . . . . 110
  - Global versus local . . . . . 111
  - Static variables . . . . . 112
  - Exceptions . . . . . 113
- Function Scope . . . . . 115
  - Include and require . . . . . 115
  - Recursion . . . . . 116
- Summary . . . . . 118

**Chapter 7: Passing Information between Pages . . . . . 119**

- HTTP Is Stateless . . . . . 119
- GET Arguments . . . . . 120
- A Better Use for GET-Style URLs . . . . . 122
- POST Arguments . . . . . 124
- Formatting Form Variables . . . . . 125
  - Consolidating forms and form handlers . . . . . 128
  - Using array variables with forms . . . . . 129
- PHP Superglobal Arrays . . . . . 132
- Extended Example: An Exercise Calculator . . . . . 134
- Summary . . . . . 136

**Chapter 8: Strings . . . . . 137**

- Strings in PHP . . . . . 137
  - Interpolation with curly braces . . . . . 138
  - Characters and string indexes . . . . . 138
  - String operators . . . . . 139
  - Concatenation and assignment . . . . . 139
  - The heredoc syntax . . . . . 140
- String Functions . . . . . 140
  - Inspecting strings . . . . . 141
  - Finding characters and substrings . . . . . 141
  - Comparison and searching . . . . . 143
  - Searching . . . . . 143
  - Substring selection . . . . . 144
  - String cleanup functions . . . . . 145
  - String replacement . . . . . 146
  - Case functions . . . . . 148
  - Escaping functions . . . . . 149
  - Printing and output . . . . . 150
- Extended Example: An Exercise Calculator . . . . . 151
- Summary . . . . . 156

**Chapter 9: Arrays and Array Functions . . . . . 157**

- The Uses of Arrays . . . . . 157
- What Are PHP Arrays? . . . . . 158
- Creating Arrays . . . . . 160
  - Direct assignment . . . . . 160
  - The array() construct . . . . . 160
  - Specifying indices using array() . . . . . 161
  - Functions returning arrays . . . . . 161
- Retrieving Values . . . . . 162
  - Retrieving by index . . . . . 162
  - The list() construct . . . . . 162
- Multidimensional Arrays . . . . . 163
- Inspecting Arrays . . . . . 164
- Deleting from Arrays . . . . . 165
- Iteration . . . . . 165
  - Support for iteration . . . . . 165
  - Using iteration functions . . . . . 167
  - Our favorite iteration method: foreach . . . . . 167
  - Iterating with current() and next() . . . . . 168
  - Starting over with reset() . . . . . 170
  - Reverse order with end() and prev() . . . . . 171
  - Extracting keys with key() . . . . . 171
  - Empty values and the each() function . . . . . 172
  - Walking with array\_walk() . . . . . 173
- Extended Example: An Exercise Calculator . . . . . 175
- Summary . . . . . 189

**Chapter 10: Numbers . . . . . 191**

- Numerical Types . . . . . 191
- Mathematical Operators . . . . . 192
  - Arithmetic operators . . . . . 192

Arithmetic operators and types	192
Incrementing operators	193
Assignment operators	194
Comparison operators	194
Precedence and parentheses	195
Simple Mathematical Functions	196
Randomness	196
Seeding the generator	197
Example: Making a random selection	199
Extended Example: An Exercise Calculator	200
Summary	206

## **Chapter 11: Basic PHP Gotchas . . . . . 209**

Installation-Related Problems	209
Symptom: Text of file displayed in browser window	209
Symptom: PHP blocks showing up as text under HTTP or browser prompts you to save file	210
Symptom: Server or host not found/Page cannot be displayed	210
Rendering Problems	210
Symptom: Totally blank page	210
Symptom: Document contains no data	211
Symptom: Incomplete or unintended page	212
Symptom: PHP code showing up in Web browser	214
Failures to Load Page	215
Symptom: Page cannot be found	215
Symptom: Failed opening [file] for inclusion	216
Parse Errors	216
Symptom: Parse error message	216
The missing semicolon	217
No dollar signs	217
Mode issues	218
Unescaped quotes	219
Unterminated strings	219
Other parse error causes	219
File Permissions	219
Symptom: HTTP error 403	220
Missing Includes	220
Symptom: Include warning	220
Unbound Variables	221
Symptom: Variable not showing up in print string	221
Symptom: Numerical variable unexpectedly zero	221
Causes of unbound variables	221
Overwritten Variables	223
Symptom: The variable has a valid value, just not the one you expected	223
Function Problems	224
Symptom: Call to undefined function my_function()	224
Symptom: Call to undefined function ()	224
Symptom: Call to undefined function array()	224
Symptom: Cannot redeclare my_function()	225
Symptom: Wrong parameter count	225

Math Problems . . . . .	225
Symptom: Division-by-zero warning . . . . .	225
Symptom: Unexpected arithmetic result . . . . .	226
Symptom: NaN (or NAN) . . . . .	226
Time-outs . . . . .	227
Summary . . . . .	227

## Part II: PHP and MySQL

231

### Chapter 12: Choosing a Database for PHP . . . . . 233

What Is a Database? . . . . .	233
Why a Database? . . . . .	233
Maintainability and scalability . . . . .	234
Portability . . . . .	234
Avoiding awkward programming . . . . .	234
Searching . . . . .	234
Security . . . . .	235
N-tier architecture . . . . .	235
Potential downside: Performance . . . . .	235
Choosing a Database . . . . .	236
You may not have a choice . . . . .	236
Flat-file, relational, object-relational . . . . .	236
ODBC/JDBC versus native API . . . . .	237
Swappable databases . . . . .	238
Advanced Features to Look For . . . . .	238
A GUI . . . . .	238
Subquery . . . . .	238
SELECT INTO . . . . .	239
Complex joins . . . . .	239
Threading and locking . . . . .	239
Transactional databases . . . . .	239
Procedures and triggers . . . . .	240
Indexes . . . . .	240
Foreign keys and integrity constraints . . . . .	240
Database replication . . . . .	241
PHP-Supported Databases . . . . .	241
Database Abstraction (or Not) . . . . .	242
Our Focus: MySQL . . . . .	243
Summary . . . . .	244

### Chapter 13: SQL Tutorial . . . . . 245

Relational Databases and SQL . . . . .	245
SQL Standards . . . . .	246
The Workhorses of SQL . . . . .	246
SELECT . . . . .	247
INSERT . . . . .	251
UPDATE . . . . .	251
DELETE . . . . .	252

Database Design . . . . .	252
Privileges and Security . . . . .	255
Setting database permissions . . . . .	255
Keep database passwords outside the Web tree . . . . .	256
Use two layers of password protection . . . . .	257
Learn to make backups . . . . .	258
Summary . . . . .	258
<b>Chapter 14: MySQL Database Administration . . . . .</b>	<b>259</b>
MySQL Licensing . . . . .	259
Installing MySQL: Moving to Version 4 . . . . .	260
Preinstall considerations . . . . .	260
Downloading MySQL . . . . .	262
Installing MySQL on Windows . . . . .	262
Installing MySQL on Unix . . . . .	263
Installing MySQL on Mac OS X . . . . .	264
Post-installation housekeeping . . . . .	264
Basic MySQL client commands . . . . .	265
MySQL User Administration . . . . .	265
Local development . . . . .	268
Standalone Web site . . . . .	268
Shared-hosting Web site . . . . .	269
PHPMyAdmin . . . . .	269
Backups . . . . .	272
Replication . . . . .	274
Recovery . . . . .	276
mysamchk . . . . .	277
mysqlcheck . . . . .	278
Summary . . . . .	278
<b>Chapter 15: PHP/MySQL Functions . . . . .</b>	<b>279</b>
Connecting to MySQL . . . . .	279
Making MySQL Queries . . . . .	281
Fetching Data Sets . . . . .	282
Getting Data about Data . . . . .	284
Multiple Connections . . . . .	285
Building in Error Checking . . . . .	287
Creating MySQL Databases with PHP . . . . .	288
MySQL data types . . . . .	289
MySQL Functions . . . . .	291
Summary . . . . .	293
<b>Chapter 16: Displaying Queries in Tables . . . . .</b>	<b>295</b>
HTML Tables and Database Tables . . . . .	295
One-to-one mapping . . . . .	296
Example: A single-table displayer . . . . .	296
The sample tables . . . . .	298
Improving the displayer . . . . .	299



Complex Mappings . . . . .	302
Multiple queries versus complex printing . . . . .	302
A multiple-query example . . . . .	303
A complex printing example . . . . .	305
Creating the Sample Tables . . . . .	307
Summary . . . . .	309
<b>Chapter 17: Building Forms from Queries . . . . .</b>	<b>311</b>
HTML Forms . . . . .	311
Basic Form Submission to a Database . . . . .	312
Self-Submission . . . . .	314
Editing Data with an HTML Form . . . . .	322
TEXT and TEXTAREA . . . . .	322
CHECKBOX . . . . .	324
RADIO . . . . .	327
SELECT . . . . .	332
Summary . . . . .	335
<b>Chapter 18: PHP/MySQL Efficiency . . . . .</b>	<b>337</b>
Connections — Reduce, Reuse, Recycle . . . . .	337
A bad example: One connection per statement . . . . .	338
Multiple results don't need multiple connections . . . . .	338
Persistent connections . . . . .	339
Indexing and Table Design . . . . .	340
Indexing . . . . .	340
Everything including the kitchen sink . . . . .	342
Other types of indexes . . . . .	343
Table design . . . . .	344
Making the Database Work for You . . . . .	345
It's probably faster than you are . . . . .	345
A bad example: Looping, not restricting . . . . .	345
Creating date and time fields . . . . .	347
Finding the last inserted row . . . . .	348
Summary . . . . .	350
<b>Chapter 19: PHP/MySQL Gotchas . . . . .</b>	<b>351</b>
No Connection . . . . .	351
Problems with Privileges . . . . .	353
Unescaped Quotes . . . . .	354
Broken SQL Statements . . . . .	356
Misspelled names . . . . .	358
Comma faults . . . . .	358
Unquoted string arguments . . . . .	358
Unbound variables . . . . .	359
Too Little Data, Too Much Data . . . . .	359
Specific SQL Functions . . . . .	360
mysql_affected_rows() versus mysql_num_rows() . . . . .	360
mysql_result() . . . . .	361
OCIFetch() . . . . .	361
Debugging and Sanity Checking . . . . .	361
Summary . . . . .	362

**Chapter 20: Object-Oriented Programming with PHP . . . . . 365**

What Is Object-Oriented Programming? . . . . .	365
The simple idea . . . . .	366
Elaboration: Objects as datatypes . . . . .	367
Elaboration: Inheritance . . . . .	367
Elaboration: Encapsulation . . . . .	369
Elaboration: Constructors and destructors . . . . .	369
Terminology . . . . .	369
Basic PHP Constructs for OOP . . . . .	371
Defining classes . . . . .	371
Accessing member variables . . . . .	372
Creating instances . . . . .	372
Constructor functions . . . . .	372
Inheritance . . . . .	373
Overriding functions . . . . .	375
Chained subclassing . . . . .	375
Modifying and assigning objects . . . . .	377
Scoping issues . . . . .	377
Advanced OOP Features . . . . .	378
Public, Private, and Protected Members . . . . .	378
Interfaces . . . . .	380
Constants . . . . .	380
Abstract Classes . . . . .	381
Simulating class functions . . . . .	381
Calling parent functions . . . . .	382
Automatic calls to parent constructors . . . . .	384
Simulating method overloading . . . . .	384
Serialization . . . . .	385
Introspection Functions . . . . .	387
Function overview . . . . .	387
Example: Class genealogy . . . . .	390
Example: Matching variables and DB columns . . . . .	392
Example: Generalized test methods . . . . .	395
Extended Example: HTML Forms . . . . .	398
Gotchas and Troubleshooting . . . . .	404
Symptom: Member variable has no value in member function . . . . .	404
Symptom: Parse error, expecting T_VARIABLE ... . . . .	405
OOP Style in PHP . . . . .	405
Naming conventions . . . . .	405
Accessor functions . . . . .	405
Designing for inheritance . . . . .	406
Summary . . . . .	407

**Chapter 21: Advanced Array Functions . . . . . 409**

Transformations of Arrays . . . . .	409
Retrieving keys and values . . . . .	410
Flipping, reversing, and shuffling . . . . .	410
Merging, padding, slicing, and splicing . . . . .	412
Stacks and Queues . . . . .	415

Translating between Variables and Arrays . . . . .	416
Sorting . . . . .	417
Printing Functions for Visualizing Arrays . . . . .	418
Summary . . . . .	419

## **Chapter 22: String and Regular Expression Functions . . . . . 421**

Tokenizing and Parsing Functions . . . . .	421
Why Regular Expressions? . . . . .	424
Regex in PHP . . . . .	424
An example of POSIX-style regex . . . . .	425
Regular expression functions . . . . .	426
Perl-Compatible Regular Expressions . . . . .	427
Example: A Simple Link-Scraper . . . . .	430
The regular expression . . . . .	430
Using the expression in a function . . . . .	432
Advanced String Functions . . . . .	434
HTML functions . . . . .	434
Hashing using MD5 . . . . .	435
Strings as character collections . . . . .	436
String similarity functions . . . . .	438
Summary . . . . .	438

## **Chapter 23: Filesystem and System Functions . . . . . 439**

Understanding PHP File Permissions . . . . .	439
File Reading and Writing Functions . . . . .	440
File open . . . . .	441
File read . . . . .	443
Constructing file downloads by using fpassthru() . . . . .	444
File write . . . . .	445
File close . . . . .	446
Filesystem and Directory Functions . . . . .	447
feof . . . . .	447
file_exists . . . . .	447
filesize . . . . .	447
Network Functions . . . . .	450
Syslog functions . . . . .	450
DNS functions . . . . .	450
Socket functions . . . . .	450
Date and Time Functions . . . . .	451
If you don't know either date or time . . . . .	451
If you've already determined the date/time/timestamp . . . . .	452
Calendar Conversion Functions . . . . .	453
Summary . . . . .	454

## **Chapter 24: Sessions, Cookies, and HTTP . . . . . 455**

What's a Session? . . . . .	455
So what's the problem? . . . . .	455
Why should you care? . . . . .	456
Home-Grown Alternatives . . . . .	456
IP address . . . . .	456
Hidden variables . . . . .	457
Cookie-based homegrown sessions . . . . .	457

How Sessions Work in PHP	458
Making PHP aware of your session	459
Propagating session variables	459
Where is the data really stored?	461
Sample Session Code	462
Session Functions	465
Configuration Issues	468
Cookies	469
The setcookie() function	470
Examples	471
Deleting cookies	472
Reading cookies	472
register_globals and variable overwriting	473
Cookie pitfalls	474
Sending HTTP Headers	475
Example: Redirection	476
Example: HTTP authentication	476
Header gotchas	477
Gotchas and Troubleshooting	478
Summary	478

## **Chapter 25: Types and Type Conversions** . . . . . **479**

Type Round-up	479
Resources	480
What are resources?	480
How to handle resources	480
Type Testing	481
Assignment and Coercion	481
Integer overflow	486
Finding the largest integer	486
Summary	487

## **Chapter 26: Advanced Use of Functions** . . . . . **489**

Variable Numbers of Arguments	489
Default arguments	489
Arrays as multiple-argument substitutes	490
Multiple arguments in PHP4 and above	491
Call-by-Value	493
Call-by-Reference	493
Variable Function Names	495
An Extended Example	495
Summary	499

## **Chapter 27: Mathematics** . . . . . **501**

Mathematical Constants	501
Tests on Numbers	502
Base Conversion	503
Exponents and Logarithms	506
Trigonometry	507
Arbitrary Precision (BC)	511
An arbitrary-precision example	512
Converting code to arbitrary-precision	513
Summary	515

<b>Chapter 28: PEAR</b> . . . . .	<b>517</b>
What Is PEAR? . . . . .	517
The PEAR Package System . . . . .	518
A sampling of PEAR packages . . . . .	518
How the PEAR database works . . . . .	519
The Package Manager . . . . .	519
Using the Manager . . . . .	523
PHP Foundation Classes (PFC) . . . . .	525
PHP Extension Code Library (PECL) . . . . .	525
The PEAR Coding Style . . . . .	525
Indenting, whitespace, and line length . . . . .	526
Formatting control structures . . . . .	526
Formatting functions and function calls . . . . .	528
Summary . . . . .	528
<b>Chapter 29: Security</b> . . . . .	<b>531</b>
Possible Attacks . . . . .	532
Site defacement . . . . .	532
Accessing source code . . . . .	533
Reading arbitrary files . . . . .	535
Running arbitrary programs . . . . .	537
Viruses and other e-critters . . . . .	538
E-mail safety . . . . .	539
Register Globals . . . . .	540
File Uploads . . . . .	542
Encryption . . . . .	545
Public-key encryption . . . . .	545
Single-key encryption . . . . .	546
Encrypting cookies . . . . .	548
Hashing . . . . .	549
Digitally signing files . . . . .	550
Secure Sockets Layer . . . . .	551
FYI: Security Web Sites . . . . .	552
Summary . . . . .	552
<b>Chapter 30: Configuration</b> . . . . .	<b>555</b>
Viewing Environment Variables . . . . .	555
Understanding PHP Configuration . . . . .	555
Compile-time options . . . . .	556
CGI compile-time options . . . . .	559
Apache configuration files . . . . .	561
The php.ini file . . . . .	563
Improving PHP Performance . . . . .	566
Summary . . . . .	568
<b>Chapter 31: Exceptions and Error Handling</b> . . . . .	<b>569</b>
Error Handling in PHP5 . . . . .	569
Errors and exceptions . . . . .	569
The Exception class . . . . .	571
The try/catch block . . . . .	572
Throwing an exception . . . . .	572

Defining your own Exception subclasses . . . . .	573
Limitations of Exceptions in PHP . . . . .	575
Other Methods of Error Handling . . . . .	576
Native PHP errors . . . . .	576
Defining an error handler . . . . .	578
Triggering a user error . . . . .	579
Logging and Debugging . . . . .	580
Summary . . . . .	581

## **Chapter 32: Debugging . . . . . 583**

General Troubleshooting Strategies . . . . .	583
Change one thing at a time . . . . .	583
Try to isolate the problem . . . . .	584
Simplify, then build up . . . . .	584
Check the obvious . . . . .	584
Document your solution . . . . .	584
After fixing, re-test . . . . .	584
A Menagerie of Bugs . . . . .	584
Compile-time bugs . . . . .	585
Run-time bugs . . . . .	585
Logical bugs . . . . .	585
Using Web Server Logs . . . . .	585
Apache . . . . .	585
IIS . . . . .	587
PHP Error Reporting and Logging . . . . .	587
Error reporting . . . . .	587
Error logging . . . . .	588
Choosing which errors to report or log . . . . .	588
Error-Reporting Functions . . . . .	589
Diagnostic print statements . . . . .	589
Using print_r() . . . . .	590
Using syslog() . . . . .	590
Logging to a custom location . . . . .	592
Using error_log() . . . . .	592
Visual Debugging Tools . . . . .	593
Avoiding errors in the first place . . . . .	594
Finding errors when they occur . . . . .	595
Summary . . . . .	596

## **Chapter 33: Style . . . . . 599**

The Uses of Style . . . . .	599
Readability . . . . .	600
Comments . . . . .	602
PHPDoc . . . . .	602
File and variable names . . . . .	603
Uniformity of style . . . . .	605
Maintainability . . . . .	605
Avoid magic numbers . . . . .	605
Functions . . . . .	606
Include files . . . . .	606
Object wrappers . . . . .	607
Consider using version control . . . . .	607

Robustness	607
Unavailability of service	608
Unexpected variable types	608
Efficiency and Conciseness	608
Efficiency: Only the algorithm matters	609
Efficiency optimization tips	609
Conciseness: The downside	610
Conciseness tips	611
HTML Mode or PHP Mode?	613
Minimal PHP	613
Maximal PHP	614
Medium PHP	615
The heredoc style	616
Separating Code from Design	618
Functions	618
Cascading style sheets in PHP	618
Templates and page consistency	618
Summary	620

## Part IV: Connections

621

### Chapter 34: PostgreSQL 623

Why Choose PostgreSQL?	623
Why Object-Relational Anyway?	624
Installing PostgreSQL	624
Linux installation	625
But is it a database yet?	626
Down to Real Work	627
PHP and PostgreSQL	629
The Cartoons Database	630
Summary	637

### Chapter 35: Oracle 639

When Do You Need Oracle?	639
Money	640
Other rivalrous resources	640
Huge data sets	640
Lots of big formulaic writes or data munging	640
Triggers	641
Legal liability	641
Bottom line: Two-year outlook	641
Oracle and Web Architecture	641
Specialized team members	642
Shared development databases	642
Limited schema changes	642
Tools (or lack thereof)	642
Replication and failover	642
Data caching	643
Using OCI8 Functions	643
Escaping strings	644
Parsing and executing	644

Error reporting . . . . .	644
Memory management . . . . .	644
Ask for nulls . . . . .	644
Fetching entire data sets . . . . .	645
All caps . . . . .	645
Transactionality . . . . .	645
Stored procedures and cursors . . . . .	646
Project: Point Editor . . . . .	647
Project: Batch Editor . . . . .	657
Summary . . . . .	667

## **Chapter 36: PEAR Database Functions . . . . . 669**

The Debatable Virtue of Database Independence . . . . .	669
Native database connectivity . . . . .	672
Database abstraction . . . . .	673
Pear DB Concepts . . . . .	673
Data Source Names (DSNs) . . . . .	674
Connection . . . . .	675
Query . . . . .	676
Row retrieval . . . . .	676
Disconnection . . . . .	676
A complete example . . . . .	677
PEAR DB Functions . . . . .	678
Members of the DB class . . . . .	678
Members of the DB_Common class . . . . .	678
Members of the DB_Result class . . . . .	679
Summary . . . . .	679

## **Chapter 37: E-mail . . . . . 681**

Understanding E-mail . . . . .	681
TCP/IP server . . . . .	682
Mail Transfer Agent, aka SMTP server . . . . .	682
Mail spool . . . . .	683
Mail User Agent, aka local mail client . . . . .	684
Mail-retrieval program, aka POP/IMAP server . . . . .	684
Mailing list manager . . . . .	685
Receiving E-mail with PHP . . . . .	686
Implementing from scratch . . . . .	686
Modifying other people's PHP . . . . .	686
Cosmetic changes . . . . .	687
Sending E-mail with PHP . . . . .	687
Windows configuration . . . . .	688
Unix configuration . . . . .	688
The mail function . . . . .	688
More Fun with PHP E-mail . . . . .	690
Sending mail from a form . . . . .	690
Sending mail from a database . . . . .	693
Sending attachments with MIME mail . . . . .	694
A custom PHP mail application . . . . .	696
Sending mail from a cronjob . . . . .	699
E-mail Gotchas . . . . .	701
Summary . . . . .	701



**Chapter 38: PHP and JavaScript . . . . . 703**

Outputting JavaScript with PHP . . . . .	703
Dueling objects . . . . .	704
PHP doesn't care what it outputs . . . . .	704
Where to use JavaScript . . . . .	705
PHP as a Backup for JavaScript . . . . .	705
Static Versus Dynamic JavaScript . . . . .	707
Dynamically generated forms . . . . .	708
Passing data back to PHP from JavaScript . . . . .	714
Summary . . . . .	717

**Chapter 39: PHP and Java . . . . . 719**

PHP for Java programmers . . . . .	719
Similarities . . . . .	719
Differences . . . . .	720
Java Server Pages and PHP . . . . .	721
Guide to this book . . . . .	722
Integrating PHP and Java . . . . .	723
The Java SAPI . . . . .	723
The Java extension . . . . .	724
The Java object . . . . .	726
Errors and exceptions . . . . .	727
Potential gotchas . . . . .	728
The sky's the limit . . . . .	729
Summary . . . . .	729

**Chapter 40: PHP and XML . . . . . 731**

What Is XML? . . . . .	731
Working with XML . . . . .	734
Documents and DTDs . . . . .	735
The structure of a DTD . . . . .	736
Validating and nonvalidating parsers . . . . .	739
SAX versus DOM . . . . .	739
DOM . . . . .	740
Using DOM XML . . . . .	740
DOM functions . . . . .	741
SAX . . . . .	743
Using SAX . . . . .	743
SAX options . . . . .	745
SAX functions . . . . .	746
SimpleXML API . . . . .	747
Using SimpleXML . . . . .	747
SimpleXML functions . . . . .	748
A Sample XML Application . . . . .	748
Gotchas and Troubleshooting . . . . .	755
Summary . . . . .	756

**Chapter 41: Web Services . . . . . 757**

The End of Programming as We Know It . . . . .	757
The ugly truth about data munging . . . . .	757
Brutal simplicity . . . . .	758

REST, XML-RPC, SOAP, .NET . . . . .	760
REST . . . . .	760
XML-RPC . . . . .	761
SOAP . . . . .	762
.NET services . . . . .	763
Current Issues with Web Services . . . . .	763
Fat and slow . . . . .	763
Potentially heavy load . . . . .	763
Standards . . . . .	764
Hide and seek . . . . .	764
Who pays and how? . . . . .	764
Project: A REST Client . . . . .	765
Project: A SOAP Server and Client . . . . .	770
Summary . . . . .	774

## **Chapter 42: Graphics . . . . . 775**

Your Options . . . . .	775
HTML Graphics . . . . .	775
Creating images using gd . . . . .	780
What is gd? . . . . .	780
Image formats and browsers . . . . .	780
Choosing versions . . . . .	781
Installation . . . . .	782
gd Concepts . . . . .	782
Functions . . . . .	784
Images and HTTP . . . . .	786
Example: Fractal images . . . . .	788
Gotchas and Troubleshooting . . . . .	795
Symptom: Completely blank image . . . . .	796
Symptom: Headers already sent . . . . .	796
Symptom: Broken image . . . . .	796
Summary . . . . .	797

## **Part V: Case Studies**

**799**

### **Chapter 43: Weblogs . . . . . 801**

Why Weblogs? . . . . .	801
The Simplest Weblog . . . . .	801
Adding an HTML Editing Tool . . . . .	808
Adding Database Connectivity . . . . .	809
Changes and Additions . . . . .	817
Summary . . . . .	817

### **Chapter 44: User Authentication . . . . . 819**

Designing a User-Authentication System . . . . .	819
Avoiding Common Security Issues . . . . .	820
Turn off register_globals . . . . .	821
Check for string length and safety . . . . .	821
One-way encrypt passwords . . . . .	822

Registration . . . . .	823
Login/Logout . . . . .	831
User Tools . . . . .	836
Forgotten password . . . . .	836
Changing sensitive user data . . . . .	839
Edit non-sensitive user data . . . . .	846
Administrator Tools . . . . .	851
Authorization: Basic auth, cookie, database, and IP . . . . .	851
Login as user . . . . .	852
Summary . . . . .	855
<b>Chapter 45: A User-Rating System . . . . .</b>	<b>857</b>
Initial Design . . . . .	857
Domain: A quotation site . . . . .	858
Possible ratings . . . . .	858
Linking ratings with content . . . . .	859
Collecting Votes . . . . .	859
Aggregating Results . . . . .	867
Extensions and Alternatives . . . . .	869
Summary . . . . .	870
<b>Chapter 46: A Trivia Game . . . . .</b>	<b>871</b>
Concepts Used in This Chapter . . . . .	871
The Game . . . . .	872
Our version . . . . .	872
Sample screens . . . . .	872
The rules . . . . .	874
Playing the game yourself . . . . .	875
The Code . . . . .	875
Code files . . . . .	875
Creating the database . . . . .	906
General Design Considerations . . . . .	910
Separation of code and display . . . . .	910
Persistence of data . . . . .	910
Exception handling . . . . .	911
Summary . . . . .	911
<b>Chapter 47: Converting Static HTML Sites . . . . .</b>	<b>913</b>
Planning the Big Upgrade . . . . .	913
The baby and the bathwater . . . . .	914
Technical assessment . . . . .	915
Redesigning the User Interface . . . . .	916
Planning a New Database Schema . . . . .	918
Dumping Data into a Database . . . . .	922
Data-messaging . . . . .	922
Data dumping . . . . .	923
Harvesting data . . . . .	928

Templating . . . . .	932
Performance and Caching . . . . .	941
Caching . . . . .	942
Summary . . . . .	943
<b>Chapter 48: Data Visualization with Venn Diagrams . . . . .</b>	<b>945</b>
Scaled Venn Diagrams . . . . .	945
The task . . . . .	945
Outline of the Code . . . . .	946
Necessary Trigonometry . . . . .	947
Planning the Display . . . . .	950
Simplifying assumptions . . . . .	950
Determining size and scale . . . . .	951
Display . . . . .	957
Visualizing a Database . . . . .	958
Trying it out . . . . .	963
Extensions . . . . .	965
Summary . . . . .	965
<b>Appendix A: PHP for C Programmers . . . . .</b>	<b>967</b>
<b>Appendix B: PHP for Perl Hackers . . . . .</b>	<b>973</b>
<b>Appendix C: PHP for HTML Coders . . . . .</b>	<b>979</b>
<b>Appendix D: PHP Resources . . . . .</b>	<b>987</b>
<b>Index . . . . .</b>	<b>997</b>