

Symbols

- <?, 113
 - . (concatenation operator), 32
 - “ ” (double quotes), strings, 19–20
 - == (equality operators), 42
 - #! (hash-bangs), CLI PHP shell scripts environments, 511–512
 - ? (question mark), 39
 - & (reference) sign, 8
 - @ (silence operator), 39
 - ‘ ’ (single quotes), strings, 20
 - ./configure, 486
 - \$_COOKIE, 115
 - \$_GET, 115
 - \$_POST, 115
 - \$key, 26
 - \$this
 - accessing
 - methods, 59–62
 - properties, 59–61
 - static methods, 64–65
 - static properties, 62–64
 - PHP 5, 437–440
 - \$type, 130
 - \$value, 26
 - __autoload(), 7, 80, 82, 197
 - __call(), 87, 109
 - __construct(), 3, 57
 - __destructor(), 3
 - __toString(), 76–77
 - 118N, 567
 - 118Nv2, 567
- ## A
- a, 366
 - ab (Apache Benchmarking tool), 457–458
 - abstract, 615–616
 - abstract classes, 5, 72–73
 - abstract methods, 6, 72–73
 - abstracted errors, PEAR DB, 186
 - error codes, 186–187
 - error handling, 187
 - Acceleration Mode (ZPS), 472
 - accessing, 616–617
 - array elements, 24
 - files, 261
 - functions, 262–264
 - methods with \$this, 59, 61–62
 - nested arrays, 26
 - properties with \$this, 59–61
 - resources, 497–498
 - static methods with \$this, 64–65
 - static properties with \$this, 62–64
 - zval values, 500–501
 - adding INI directives to extensions, 503–504
 - Addive System of Photographic Exposure (APEX), 327
 - addslashes(), 128
 - Advanced PHP Cache. *See* APC
 - Advanced PHP Debugger. *See* APD
 - aligning text, 325
 - alldeps, 366
 - allow_url_fopen, 198
 - alpha blending, 321
 - analyzing trace data (APD), 462–465
 - anti-aliasing, 318
 - ApacheBench, 457–458
 - APC (Advanced PHP Cache), 470, 530
 - APD (Advanced PHP Debugger), 461, 588
 - analyzing trace data, 462–465
 - installing, 461–462
 - APEX (Additive System of Photographic Exposure), 327
 - API, 422
 - extension API, 490

- reflection API,
 - 103–105
 - wrapping third-party extensions, 493
 - architecture
 - one script per
 - function, 144
 - one script serves
 - all, 143
 - separating logic from
 - layout, 144–146
 - Archive_Tar, 548
 - Archive_Zip, 548
 - arg_separator.input, 198
 - arguments, 487
 - array elements,
 - modifying/creating, 25
 - array indexes, 23, 196
 - array_walk(), 170
 - array(), 23
 - array_merge(), 445–446
 - ArrayAccess interface, 88
 - arrays, 23
 - accessing
 - array elements, 24
 - nested arrays, 26
 - array(), 23
 - associative arrays, 88
 - modifying, creating
 - array elements, 25
 - overloading array
 - access syntax, 88–89
 - reading array
 - values, 25
 - traversing, 30
 - each(), 28
 - foreach, 26–27
 - list(), 28–30
 - rest(), 28
 - assessing string offsets, 21–22
 - assigning \$this in PHP 5, 437, 439–440
 - assignment operators, 32–33
 - associative arrays, 88
 - atomic, 278
 - attributes, 221
 - debug, 422
 - default, 428
 - format, 422
 - install-as, 422
 - md5sum, 421
 - name, 421–423, 428
 - optional, 424
 - phpapi, 422
 - platform, 421
 - prompt, 428
 - rel, 424
 - role, 421
 - type, 422–423
 - zendapi, 422
 - zts, 422
 - Auth, 392, 398, 527
 - DB and user data, 394–396
 - password files, 393–394
 - scalability, 397–398
 - security, 396
 - Auth_HTTP, 397
 - disabling
 - session.trans_sid, 396
 - HTTPS, 397
 - sessions, 395
 - Auth_Enterprise, 528
 - Auth_HTTP, 397, 528
 - Auth_PrefManager, 528
 - Auth_RADIUS, 528
 - Auth_SASL, 529
 - authentication
 - Auth, 527
 - Auth_Enterprise, 528
 - Auth_HTTP, 528
 - Auth_PrefManager, 528
 - Auth_RADIUS, 528
 - Auth_SASL, 529
 - LiveUser, 529
 - radius, 530
 - sasl, 530
 - authentication ticket, 392
 - author, 617
 - Automatic Optimization, 470–472
 - avoiding dependencies (PEAR), 425–426
- ## B
- B, 365
 - back references, 295
 - backward compatibility, 408
 - bar charts, 320–325
 - basic IT template (HTML_Template_IT), 384–385
 - Basic Multilingual Plane (BMP), 330
 - bcompiler, 589
 - Benchmark, 530
 - benchmarking, 457
 - ApacheBench, 457–458
 - Benchmark, 530
 - micro-benchmarks, 477–479
 - Seige, 458–459
 - testing versus real traffic, 459
 - binaries directory, 353
 - Binary Large Object. *See* BLOB
 - binary operators, 31–32
 - binary strings, 488

- binding variables
 - (MySQL), prepared statements, 156–158
 - bitwise operators, 35
 - BLOB (Binary Large Object), 158
 - inserting BLOB data (MySQL), 159
 - retrieving BLOB data (MySQL), 159–160
 - blocks, 384–385
 - HTML_Template_Flexy, 388–389
 - HTML_Template_IT, 386–387
 - BMP (Basic Multilingual Plane), 330
 - body tags, 221
 - boolean stream_eof(void), 270
 - boolean stream_flush(void), 271
 - boolean stream_open(), 270
 - boolean stream_seek(int offset, int whence), 271
 - Booleans, 22
 - bot-proof submission forms, 315–320
 - bottlenecks, 459–460
 - break, 43
 - browsing SimpleXML objects, 233–234
 - buffered queries (MySQL), 153
 - building
 - packages
 - PEAR Example (HelloWorld), 411–414
 - regression tests (PEAR), 416
 - tarballs (PEAR), 414
 - verification (PEAR), 414–415
 - built-in functions, OO wrappers, 456
 - by-reference, 51
 - assignment operators, 33
 - parameters, 8, 52
 - by-value, 50
 - parameters, 52
 - bz2, 545
 - bz2_filter, 595
- ## C
- C, 356
 - C
 - including in PEAR packages, 428
 - rewriting code in C, 479
 - writing comments, 14
 - c, 356
 - C++
 - inheritance, 70
 - writing comments, 15
 - Cache, 531
 - Cache TimeToLive (cache_ttl), 359
 - Cache_Lite, 399–401, 531
 - caching, 451–453
 - APC, 530
 - Cache, 531
 - Cache_Lite, 531
 - call caching, 456
 - compiled templates, 456
 - database query/result caching, 453–455
 - exclusive caching, 475
 - output caching, 456
 - partial caching, 475
 - PEAR, 399–401
 - Calendar, 542
 - call caching, 456
 - call(), 253
 - call_user_func_array(), 87
 - case sensitivity, 30
 - cast operators, 38
 - casting objects (compatibility mode), 435–436
 - catching
 - errors (PEAR), 207
 - exceptions, 216–218
 - category, 618
 - CET (Central European Time), 305
 - CGI
 - shell scripts, 508
 - versus CLI, 508, 510
 - default parameters, 509
 - options, 509
 - php.ini name and location, 510
 - changelog element, 419
 - changes in functions
 - array_merge(), 445–446
 - stripos(), 446–447
 - strrpos(), 446–447
 - changing
 - configuration
 - parameters (PEAR Installer), 361–364
 - from PHP 4 to PHP 5
 - compatibility mode. *See* compatibility mode

- character encoding
 - (SQLite), 174
- character set conversions, 330–335
- character sets
 - character set conversions, 330–335
 - iconv_strlen, 335–337
 - iconv_strpos(), 338–339
 - languages, 329
 - mb_strlen, 335–337
 - UTF-8 characters, 337
- charts, bar charts, 320–325
- check_auth(), 132
- check_login(), 137
- checkIfNotRectangle(), 71
- class constants, 5, 65
- class inheritance, 73
- class properties, 62
- class type hints, 6
- classes, 56, 422
 - abstract classes, 5, 72–73
 - child, 70
 - class constants, 5
 - class type hints, 6
 - class type hints in function parameters, 82–83
 - declaring, 57
 - defining before usage, 443
 - error class in PHP document, 635
 - final classes, 4, 76
 - parent, 70
 - PEAR_Error class
 - addUserInfo(), 211
 - getCallback(), 211
 - getcode(), 211
 - getMessage(), 211
 - getMode(), 211
 - getType(), 211
 - getUserInfo, 211
 - PEAR_Error
 - constructor, 210–211
 - QueryCacheStrategyWrapper, 455
 - static members, 5
 - Sum class in PHP document, 635, 637
 - SumNumberElements class in PHP document, 637
 - SumNumbers class in PHP document, 637–638
 - symbol naming (PCS), 404
 - undefined symbols, 196–197
- CLI (Command Line), 507
 - parsing command-line options, 512–515
- PEAR, 408–410
 - shell scripts, 508
 - versus CGI, 508, 510
 - default parameters, 509
 - options, 509
 - php.ini name and location, 510
- CLI Installer, 378
- CLI PHP shell scripts, environments, 510
 - #! (hash-bangs), 511–512
 - execution lifetime, 511
 - user input, 510
- clients (XML RPC), 247–249
- cloning objects, 4, 66–67
- closing tags, 221
- code, optimizing. *See* optimizing, code
- code inclusion control structures
 - eval() statements, 47
 - include statements, 46–47
- coding extensions, 484–489
- Command Line. *See* CLI
- command-line
 - interfaces, 443
- command-line options, parsing, 512–515
- command-line shortcuts, 377
- command-line tools (written in PHP), PHP filter utility, 520–525
- commands
 - pear command, 354–357
 - PEAR Installer
 - commands, 364
 - pear config-get, 376
 - pear config-set, 376
 - pear config-show, 358, 376–377
 - pear download, 375
 - pear info, 369–370
 - pear install, 364–367
 - pear list, 368–369
 - pear list-all, 370
 - pear list-upgrades, 370
 - pear remote-info, 375
 - pear remote-list, 374–375

- pear search,
 - 373–374
- pear uninstall, 373
- pear upgrade,
 - 371–372
- pear upgrade-all, 372
- shortcuts, 377
- run-tests, 416
- sub-commands, 355
- comment tokens, 443–444
- comments
 - documentation comments, 613–615
 - writing, 14
- communicating with XML, 244
- comparing
 - JPEG and PNG, 320
 - object (scompatibility mode), 436–437
 - version numbers, 348–349
- comparison operators, 33–34
- compatibility
 - command-line interfaces, 443
 - comment tokens, 443–444
 - E_STRICT, 441
 - automagically creating objects, 441
 - constructors, 442
 - defining classes
 - before usage, 443
 - inherited methods, 442–443
 - public, 441
 - var, 441
 - get_class(), 440
 - MySQL, 445
 - object models, 433
- compatibility mode, 435
- casting objects, 435–436
- comparing objects, 436–437
- compile, 384
- compile directory, 388
- Compiled Code Caching (ZPS), 472–473
- compiled templates, 456
- Compiled-code Caching, 470
- compression, content
 - compression (ZPS), 476
- compression streams, 268–269
- concatenation operators, 32
- conditional control structures, 39
 - if statements, 39–41
 - switch statements, 41–42
- Config, 531
- configuration layers (PEAR Installer), 360–361
- configuration options,
 - connections (PEAR DB), 179
- configuration parameters (PEAR Installer), 358, 360
 - changing, 361–364
- configureoption
 - element, 428
- configuring
 - fetch modes (PEAR DB), 183
- Studio Server for
 - remote debugging and profiling, 650–651
- connect(), 62
- connections
 - MySQL, 151–153
 - PEAR DB
 - configuration options, 179
 - DSNs, 178
 - establishing connections, 178–179
- console
 - Console_Color, 531
 - Console_Getopt, 513, 532
 - Console_Progressbar, 532
 - Console_Table, 532
 - ecasound, 532
 - System_Command, 532
 - win32std, 533
 - xmms, 533
- Console_Color, 531
- Console_Getopt, 513, 532
- Console_Progressbar, 532
- Console_Table, 532
- constants, 30–31
 - symbol naming (PCS), 403
 - undefined symbols, 195
- constructors, 57–58
 - E_STRICT, 442
 - PEAR_Error constructor, 210–211
- Contact_Vcard_Build, 545
- Contact_Vcard_Parse, 545
- content caching, 473
- Content Compression, 471
- content compression (ZPS), 476

- content syndication, 236
- Content-Caching module, 471
- continue, 43
- contributor, 418
- control structures, 39
 - code inclusion control structures
 - eval() statements, 47
 - include statements, 46–47
 - conditional control structures, 39
 - if statements, 39–41
 - switch statements, 41–42
 - loop control structures
 - do.. while loops, 43–44
 - for loops, 44–45
 - while loops, 42
 - controlling loops, 43
 - convenience methods (PEAR DB), 188
 - getAll(), 189
 - getAssoc(), 188–189
 - getCol(), 188
 - getOne(), 188
 - getRow(), 188
 - converting XML (XSLT), 239–243
 - cookies, 131–133
 - copyright, 618
 - count deleted rows (PEAR DB), 185
 - count number of rows (PEAR DB), 186
 - crack, 600
 - createAggregate(), 174
 - createDbConnection(), 62
 - createSequence(), 185
 - cross-site scripting, protecting user input, 118–119
 - Crypt_CBC, 543
 - Crypt_CHAP, 543
 - Crypt_Crypt, 544
 - Crypt_HMAC, 544
 - Crypt_RC4, 544
 - Crypt_Xtea, 544
 - custom error handlers, PHP errors, 204
 - cvsclient, 574
 - cybercash, 585
 - cybermut, 585
 - cyrus, 574

D

 - D, 357
 - d, 357
 - daemonizing, 518
 - data, fetching with SQLite, 168–170
 - Database directory, 353
 - data files, 353
 - data handling, 301
 - formatting date and time, 305, 309–312
 - parsing date formats, 313–314
 - retrieving date and time information, 301–305
 - data source name (DSN), 178
 - data types, 18
 - arrays, 23
 - accessing array elements, 24
 - accessing nested arrays, 26
 - array(), 23
 - modifying/creating array elements, 25
 - reading array values, 25
 - traversing, 26–30
 - Booleans, 22
 - constants, 30–31
 - floating-point numbers, 19
 - integers, 19
 - null, 23
 - strings, 19
 - accessing string offsets, 21–22
 - double quotes, 19–20
 - here-docs, 21
 - single quotes, 20
 - database query/result caching, 453–455
 - databases
 - Auth, 394–396
 - DB, 533–534
 - DB_ado, 535
 - DB_DataObject, 535
 - DB_DataObject_Form Builder, 535
 - DB_ldap, 536
 - DB_ldap2, 536
 - DB_NestedSet, 536
 - DB_Pager, 537
 - DB_QueryTool, 537
 - DB_Table, 537
 - DBA, 534
 - DBA_Relational, 534
 - dbplus, 534
 - Gtk_MDB_Designer, 537
 - isis, 538
 - MDB, 538–539
 - MDB_QueryTool, 541
 - MDB2, 539–540

- mdbtools, 541
 - MySQL. *See* MySQL
 - oci8, 541
 - odbtcp, 541
 - Paradox, 542
 - PEAR DB. *See* PEAR DB
 - PEAR DB
 - SQL_Parser, 542
 - SQLite. *See* SQLite
- Date, 543
- date element, 420
- date_time, 543
- dates
 - Calendar, 542
 - Date, 543
 - date_time, 543
 - formatting, 305, 309–312
 - parsing date formats, 313–314
 - retrieving date and time information, 301–305
- daylight savings time, 310
- DB (database abstraction layer), 176, 533–534
- ldap2, 536
- PEAR DB. *See* PEAR DB
- PEAR DB
- DB ERROR ACCESS VIOLATION, 187
- DB ERROR ALREADY EXISTS, 187
- DB ERROR CANNOT CREATE, 187
- DB ERROR CANNOT DROP, 187
- DB error codes, 186–187
- DB ERROR CONNECT FAILED, 187
- DB ERROR
 - CONSTRAINT, 187
- DB ERROR
 - CONSTRAINT NOT NULL, 187
- DB ERROR DIVZERO, 187
- DB ERROR INVALID, 187
- DB ERROR INVALID DATE, 187
- DB ERROR INVALID NUMBER, 187
- DB ERROR MISMATCH, 187
- DB ERROR NODBSELECTED, 187
- DB ERROR NOSUCHDB, 187
- DB ERROR NOSUCHFIELD, 187
- DB ERROR NOSUCHTABLE, 187
- DB ERROR NOT CAPABLE, 187
- DB ERROR NOT FOUND, 187
- DB ERROR NOT LOCKED, 187
- DB ERROR SYNTAX, 187
- DB ERROR TRUNCATED, 187
- DB ERROR UNSUPPORTED, 187
- DB ERROR VALUE COUNT ON ROW, 187
- DB PORTABILITY
 - DELETE COUNT, 185
- DB PORTABILITY ERRORS, 186
- DB PORTABILITY LOWERCASE, 186
- DB PORTABILITY NULL TO EMPTY, 186
- DB PORTABILITY NUMROWS, 186
- DB PORTABILITY RTRIM, 186
- DB_ado, 535
- DB_DataObject, 535
- DB_DataObject_FormBuilder, 535
- DB_ldap, 536
- DB_NestedSet, 536
- DB_Pager, 537
- DB_QueryTool, 537
- DB_Table, 537
- DBA, 534
- DBA_Relational, 534
- dbplus, 534
- debug attributes, 422
- Debug Log Level (verbose), 360
- Debug URL, 649–650
- debuggers, Zend Studio, 648
 - Debug URL, 650
 - Internal Debugger, 649
 - Remote Debugger, 649–650
- declarations, XML, 220
- declaring
 - classes, 57
 - function parameters, 52
 - by-reference parameters, 52

- by-value
 - parameters, 52
 - default
 - parameters, 52
 - variables, 15
- default attribute, 428
- default layers, 360
- default parameters, 52
 - CLI versus CGI, 509
- delegation design patterns, implementing, 107–109
- deleting
 - cookies, 133
 - resources, 498–499
- dep element, 423
- dependencies (PEAR), 423–424
 - dep, 423
 - dependency types, 424–425
 - deps, 423
 - examples, 426
 - optional dependencies, 426
 - reasons for avoiding dependencies, 425–426
- dependency types, 424–425
- deprecated, 618
- deps element, 423
- dereference objects, 6
- description element, 417
- design patterns
 - delegation design patterns, implementing, 107–109
 - OO, 94
 - factory patterns, 97–101
 - observer patterns, 101–103
 - singleton patterns, 97–98
 - strategy patterns, 95–96
- designing for performance, 456
- caching, 451–453
 - call caching, 456
 - compiled templates, 456
 - database query/result caching, 453–455
 - output caching, 456
- generalization, 456
- OO wrappers for built-in functions, 456
- state, 450
 - isolating, 451
 - sessions, 450
- destructors, 58–59
- developer, 418
- DG libraries, exif, 326–329
- dir element, 420
- directories
 - binaries directory, 353
 - compile directory, 388
 - Data base
 - directory, 353
 - documentation base directory, 353
 - installation prefix, 353
 - PEAR data
 - directory, 359
 - PEAR documentation
 - directory, 359
 - PEAR executables
 - directory, 359
 - PEAR installer cache
 - directory, 359
- PEAR main
 - directory, 359
- PHP code
 - directory, 353
- PHP extension
 - directory, 359
- Signature Key
 - Directory, 360
 - Test base directory, 353
 - upload directory, 139
- disabling session.trans_sid, 396
- display_errors, 130, 203
- display_startup_errors (Boolean), 203
- do.. while loops, 43–44
- docblock, 614
- Document Object Model. *See* DOM
- documentation,
 - phpDocumentor tool, 638, 640
- Documentation base
 - directory, 353
- documentation
 - comments, 613–615
- documents, XHTML, 220
- DOM (Document Object Model), 9, 222
 - creating DOM trees, 229–231
 - parsing XML, 226–229
 - XPath, 229
- double quotes (“ ”), strings, 19–20
- draw(), 72–73
- dropSequence(), 185
- DSN (data source name), 178
- DSO (dynamically shared object), 485
- DST (daylight savings time), 310

- DTD files, 220
- dynamic content caching (ZPS), 473–475
- E**
- E_COMPILE_ERROR, 202
- E_COMPILE_WARNING, 202
- E_CORE_ERROR, 202
- E_ERROR, 201
- E_NOTICE, 202
- E_PARSE, 202
- E_STRICT, 202, 441
 - automagically creating objects, 441
 - constructors, 442
 - defining classes before usage, 443
 - inherited methods, 442–443
 - public, 441
 - var, 441
- E_USER_ERROR, 202
- E_USER_NOTICE, 202
- E_USER_WARNING, 202
- E_WARNING, 202
- each(), 28
- ecasound, 532
- editing files (Zend Studio), 647
- elements
 - changelog, 419
 - configureoptions, 428
 - date, 420
 - dep, 423
 - deps, 423
 - description, 417
 - dir, 420
 - email, 418
 - file, 421
 - filelist, 420
 - license, 418–419
 - maintainer, 418
 - name, 417
 - notes, 420
 - package, 417
 - provides, 422
 - release, 419
 - role, 418
 - root, 417
 - state, 419
 - summary, 417
 - user, 418
- Emacs, 406
- email element, 418
- embedding
 - HTML, 14, 112–114
 - PHP code in HTML, 112–114
- empty string handling (PEAR DB), 186
- empty(), 17
- encapsulating includes (PEAR), 411
- enchant, 599
- encryption
 - Crypt_CBC, 543
 - Crypt_CHAP, 543
 - Crypt_Crypt, 544
 - Crypt_HMAC, 544
 - Crypt_RC4, 544
 - Crypt_Xtea, 544
 - Message, 544
- entities, 221
- environments, CLI PHP
 - shell scripts, 510
 - #! (hash-bangs, 511–512)
 - execution lifetime, 511
 - user input, 510
- equality operators (==), 42
- error class in PHP
 - document, 635
- error levels, PHP errors, 201–202
- error messages, 192, 516
- error modes (PEAR)
 - PEAR_ERROR_CALL_BACK, 213
 - PEAR_ERROR_DIE, 213
 - PEAR_ERROR_PRINT, 213
 - PEAR_ERROR_TRIGGER, 213
- error reporting, PHP
 - errors, 202, 204
- error_append_string (String), 203
- error_log (String), 203
- error_prepend_string (String), 203
- error_reporting (Integer), 202
- errors, 191
 - abstracted errors. *See* abstracted errors
 - catching PEAR errors, 207
 - exceptions. *See* exceptions
 - PEAR DB, 186
 - PEAR errors, 206–207, 212, 411
 - catching errors, 207
 - delExpect(), 215
 - expectError(), 214–215
 - isError(), 207
 - popErrorHandling(), 214
 - popExpect(), 215

- pushErrorHandler(), 213–214
 - raiseError(), 209
 - raising errors, 207–208
 - setErrorHandler(), 212
 - throwError(), 208
- PHP errors, 201
 - custom error handlers, 204
 - error levels, 201–202
 - error reporting, 202, 204
 - silencing errors, 205–206
- portability errors
 - dealing with portability, 200
 - operating system differences, 197
 - PHP configuration differences, 197
 - portability tools, 200–201
 - SAPI differences, 199
- programming
 - errors, 192
 - eval(), 193
 - parse errors, 192–194
 - syntax errors, 192
- protecting scripts, 129–130
- runtime errors, 201
- SQLite, 163
- undefined symbols, 194
 - array indexes, 196
 - functions and classes, 196–197
- logical errors, 197
 - variables and constants, 195
- escape sequences, 285, 288
- establishing connections (PEAR DB), 178–179
- eval() statements, 47, 193
- example, 619
- exception handling, 7, 77–80
- exceptions, 216
 - generating, 217
 - try, catch, and throw, 216–218
- exclusive caching, 475
- exclusive locks, 276
- exec (shell scripts), 517–518
- execute(), 181–182
- executing queries (PEAR DB), 180
- execution lifetime, CLI
 - PHP shell scripts environments, 511
- exif, 326–329
- exif phpinfo() output, 326
- exif_read_data(), 327
- exit code (shell scripts), 516
- explode(), 299
- ExposureTime, 327
- expressions. *See* regular expressions
- eXtensible Stylesheet Language Transformations. *See* XLST
- extension API, 483, 490
- extension dependencies (PEAR), 424
- extensions
 - extension API, 483, 490
 - global variables, 501–502
 - iconv, 334
 - INI directives, adding, 503–504
 - mbstring, 330
 - MySQLi, 10
 - Perl, 11
 - reasons for writing your own, 483
 - Sablotron, 9
 - SimpleXML. *See* SimpleXML extension
 - SOAP extension, 257–258
 - SOAP clients, 259
 - SOAP servers, 258
 - testing, 501
 - third-party extensions. *See* third-party extensions
 - Tidy, 10
 - type specifiers, 487
 - writing, 484–489
 - memory management, 489
 - returning values from PHP functions, 490
 - self_concat(), 490–491
 - XSL, 9
- external program dependencies (PEAR), 425

F

- factory pattern, 97–101
- fann, 601
- faults (XML RPC), 246–247
- fclose(), 262
- feature, 422
- feof(), 262
- fetching data (SQLite), 168–170
- fetching modes
 - MySQL, 156
 - PEAR DB, 182–183
- fetching results (PEAR DB), 182
 - configuring fetch modes, 183
 - fetch modes, 182
 - fetchInto(), 183
 - fetchRow(), 183
 - using your own result class, 183–184
- fetchInto(), 180, 183
- fetchRow(), 183
- ffi, 589
- fgets(), 262
- File, 548
- file element, 421
- file formats
 - bz2, 545
 - Contact_Vcard_Build, 545
 - Contact_Vcard_Parse, 545
 - File_DICOM, 545
 - File_Fstab, 546
 - File_Gettext, 546
 - File_IMC, 546
 - File_Ogg, 547
 - Fileinfo, 545
 - Geneology_Gedcom, 547
 - MP3_ID, 547
 - Spreadsheet_Excel_Writer, 547
 - zip, 548
- file functions, wrapping
 - third-party extensions, 496
- file systems
 - Archive_Tar, 548
 - Archive_Zip, 548
 - File, 548
 - File_Find, 548
 - File_HtAccess, 549
 - File_Passwd, 549
 - File_SearchReplace, 549
 - File_SMBPasswd, 549
 - VFS, 549
- File_DICOM, 545
- File_Find, 548
- File_Fstab, 546
- File_Gettext, 546
- File_HtAccess, 549
- File_IMC, 546
- File_Ogg, 547
- file_open(), 496
- File_Passwd, 549
- File_SearchReplace, 549
- File_SMBPasswd, 549
- filedescriptors, 265
- Fileinfo, 545
- filelist element, 420
- files
 - accessing, 261–264
 - DTD files, 220
 - editing in Zend Studio, 647
 - including in PEAR, 410–411
 - locking, 276–277
 - program input/output, 266–267
 - removing, 277–278
 - renaming, 278
 - RSS files, 235
 - streams, 261, 264
 - filedescriptors, 265
 - files, 266–267
 - input/output streams, 267–268
 - pipes, 266
 - popen(), 264
 - proc_open(), 265
 - temporary files, 278–279
 - unlinking from directories, 277
 - uploading, 137–142
- filesource, 620
- filters, protecting
 - scripts, 127
- final, 620–621
- final classes, 4, 76
- final keyword, 4
- final methods, 75–76
- finding bottlenecks, 459
- Flexy, 387
 - basic template, 387–388
 - blocks, 388–389
 - HTML attribute handling, 390–391
 - HTML element handling, 391–392
 - markup format, 390
- floating-point numbers, 19
- flock(), 276
- FNumber, 327
- fopen(), 262
- for loops, 44–45
- force options, 365
- foreach()

- iterators, 89–94
 - with references, 8
 - traversing arrays, 26–27
- forking shell scripts, 517
- format attribute, 422
- formatting date and time, 305, 309–312
- forms, 116
 - bot-proof submission forms, 315–320
 - PEAR, 398
 - HTML_QuickForm, 398
 - login forms, 399
 - receiving data, 399
- fputs(), 262
- fread(), 262
- FreeImage, 563
- fribidi, 566
- front-ends, 354
 - PEAR Installer, 378–379, 381
 - CLI Installer, 378
 - Gtk Installer, 378–381
- FSM, 594
- ftell(), 271
- FTP, SSL support, 272
- function declaration
 - macros, 495
- function parameters, declaring, 52
- function scope, 49–50
- functions, 48, 293, 422
 - __autoload(), 80, 82, 197
 - array_walk(), 170
 - array_merge(), 445–446
 - call_user_func_array(), 87
 - check_auth(), 132
 - check_login(), 137
 - checkIfNotRectangle(), 71
 - class type hints in function parameters, 82–83
 - declaring function parameters, 52
 - exif_read_data(), 327
 - explode(), 299
 - fclose(), 262
 - feof(), 262
 - fetchInto(), 180
 - fgets(), 262
 - file-accessing functions, 262–264
 - flock(), 276
 - fopen(), 262
 - fputs(), 262
 - fread(), 262
 - function scope, 49–50
 - fwrite(), 262
 - get_class (PHP 5), 440–441
 - gettimeofday(), 302
 - gmmktime(), 303
 - iconv_substr(), 336–337
 - imagecolorallocate(), 316
 - imagecolorallocatealpha(), 321
 - imagecreatefrompng(), 321
 - imagepng(), 319
 - imagesize(), 142
 - imagettftext(), 317–318
 - matching functions, 293, 295
 - mb_convert_encoding(), 333
 - mb_strpos(), 340
 - mb_substitute_character(), 332
 - mb_substr(), 340
 - mbstring(), 333
 - microtime(), 302
 - mktime(), 303
 - move_uploaded_file(), 142
 - passing objects to, 433–435
 - php index(), 165
 - PHP_MINIT_FUNCTION(), 495
 - popen(), 264
 - preg_grep(), 295
 - preg_match, 279
 - preg_match(), 293
 - preg_match_all(), 294–295
 - preg_replace(), 296, 298
 - preg_replace_callback(), 296
 - preg_split, 299
 - proc_open(), 265
 - process_children(), 228
 - rename(), 278
 - replacement functions, 295–298
 - resource-enabled PHP function, 497
 - result set-related functions (SQLite), 172–173
 - returning values by reference, 51
 - returning values by value, 50

self_concat(), 485
 session_destroy(), 136
 session_start(), 135
 session_write_close(),
 136
 set_error_handler(),
 130
 setcookie(), 131
 splitting strings,
 299–301
 sqlite escape string(),
 170
 static variables, 53
 strlen, 48
 strpos(), 338
 stripslashes(), 446–447
 strrpos(), 446–447
 strtotime(), 314
 symbol naming
 (PCS), 404
 time(), 303
 tmpfile(), 278
 UDFs, 165
 undefined symbols,
 196–197
 unlink(), 277
 user-defined functions,
 48–49
 version_compare(),
 408
 XML_RPC_decode(),
 249
 fwrite(), 262

G

-G, 357
 Games_Chess, 596
 GD libraries, 314
 bar charts, 320–325
 bot-proof submission
 forms, 315–320
 GD phpinfo() output, 315

Geneology_Gedcom, 547
 generalization, 456
 generating exceptions,
 217
 get_class(), 440–441
 getAll(), 189
 getAssoc(), 188–189
 getAttribute(), 228
 getCol(), 188
 getName(), 56
 getOne(), 188
 getopt(), 513
 getRow(), 188
 gettimeofday(), 302
 global, 621
 global defaults, 208
 global variables
 extensions, 501–502
 protecting user input,
 117–118
 symbol naming
 (PCS), 404
 gmmktime(), 303
 Google (SOAP), 252–254
 go-pear.org, installing
 PEAR, 351–354
 graphics, manipulating
 bar charts, 320–325
 bot-proof submission
 forms, 315–320
 exif, 326–329
 Gtk, 378
 Gtk Installer, 378–381
 Gtk_MDB_Designer, 537
 Gtk_VarDump, 550
 guidelines for SQLite, 176
 Gutmans, Andi, 643

H

-h, 357
 hash tables, 23
 hash(), 125

hash-bangs (#!), 511–512
 head tags, 221
 helper, 418
 here-docs (strings), 21
 HMAC (Keyed-Hashing
 for Message
 Authentication), 123
 Crypt_HMAC, 124–127
 verification, 122–124
 HTML
 embedding, 14,
 112–114
 HTML_BBCodeParser,
 550
 HTML_Common, 550
 HTML_Crypt, 551
 HTML_CSS, 551
 HTML_Form, 551
 HTML_Javascript, 552
 HTML_Menu, 552
 HTML_Page, 552
 HTML_Parse, 553
 HTML_Progress, 553
 HTML_QuickForm, 553
 HTML_QuickForm_
 Controller, 554
 HTML_Select, 554
 HTML_Select_
 Common, 554
 HTML_Table, 555
 HTML_Table_Matrix,
 555
 HTML_Table_Sortable,
 556
 HTML_Template_
 Flexy, 556–557
 HTML_Template_IT,
 557
 HTML_Template_
 PHPLIB, 558
 HTML_Template_
 Sigma, 558

- HTML_Template_Xipe, 559
- HTML_TreeMenu, 560
- Pager, 560
- Pager_Sliding, 560
- tidy, 560
- HTML attribute
 - handling, 390–391
- HTML element handling, 391–392
- HTML_BBCodeParser, 550
- HTML_Common, 550
- HTML_Crypt, 551
- HTML_CSS, 551
- html_errors (Boolean), 203
- HTML_Form, 551
- HTML_Javascript, 552
- HTML_Menu, 552
- HTML_Page, 552
- HTML_Parse, 553
- HTML_Progress, 553
- HTML_QuickForm, 398, 553
- HTML_QuickForm_Controller, 554
- HTML_Select, 554
- HTML_Select_Common, 554
- HTML_Table, 555
- HTML_Table_Matrix, 555
- HTML_Table_Sortable, 556
- HTML_Template_Flexy, 387, 556–557
 - basic Flexy Template, 387–388
 - blocks, 388–389
 - HTML attribute handling, 390–391
 - HTML element handling, 391–392
 - markup format, 390
- HTML_Template_IT, 384, 557
 - basic IT template, 384–385
 - IT with blocks, 386–387
 - placeholder syntax, 384
- HTML_Template_PHPLIB, 558
- HTML_Template_Sigma, 558
- HTML_Template_Xipe, 559
- HTML_TreeMenu, 560
- HTTP, 561
 - HTTP, 561
 - HTTP_Client, 561
 - HTTP_Download, 561
 - HTTP_Header, 561
 - HTTP_Request, 562
 - HTTP_Server, 562
 - HTTP_Session, 562
 - HTTP_Upload, 562
 - HTTP_WebDAV_Client, 563
 - HTTP_WebDAV_Server, 563
 - SSL support, 272
- HTTP Proxy Server (http_proxy), 360
- HTTP_Client, 561
- HTTP_Download, 561
- HTTP_Header, 561
- HTTP_Request, 562
- HTTP_Server, 562
- HTTP_Session, 562
- HTTP_Upload, 562
- HTTP_WebDAV_Client, 563
- HTTP_WebDAV_Server, 563
- iconv extension, 334
- iconv_strlen, 335–337
- iconv_strpos(), 337–339
- iconv_substr(), 336–337
- IDE (Integrated Development Environment), 644
- idn, 567
- if statements, 39–41
- ignore, 622
- ignore_repeated_errors (Boolean), 203
- ignore_repeated_source (Boolean), 203
- ignore-errors, 366
- Image_Barcode, 563
- Image_Color, 564
- Image_GIS, 564
- Image_Graph, 564
- Image_GraphViz, 565
- Image_IPTC, 565
- Image_Remote, 565
- Image_Text, 565
- Image_Tools, 565
- Image_Transform, 566
- imagecolorallocate(), 316
- imagecolorallocatealpha(), 321
- imagecreatefrompng(), 321
- imagecreatetruecolor(), 316
- imagepng(), 319
- images
 - FreeImage, 563

- Image_Barcode, 563
- Image_Color, 564
- Image_GIS, 564
- Image_Graph, 564
- Image_GraphViz, 565
- Image_IPTC, 565
- Image_Remote, 565
- Image_Text, 565
- Image_Tools, 565
- Image_Transform, 566
- imagemagick, 566
- imagesize(), 142
- imagestring(), 328
- imagettftext(), 317–318
- imagemagick, 566
- implementing delegation
 - design patterns, 107–109
- include statements, 46–47
 - incapsulating (PEAR), 411
- include/require, 47
- include_once, 410
- including
 - C code in PEAR packages, 428
 - files in PEAR, 410–411
- increment/decrement operators, 37–38
- incrementing strings, 38
- indentation (PCS), 406–407
- indirect references to variables, 16
- inheritance, 68
 - C++, 70
 - class inheritance, 73
 - interfaces, 75
- inheritdoc (inline), 622
- INI directives, 501
 - adding to extensions, 503–504
- Inline_C, 589
- input filters, protecting scripts, 127
- input validation, 120–122
- input/output streams, 267–268
- inserting BLOB data (MySQL), 159
- install-as attribute, 422
- installation prefix, 353
- installing
 - APD, 461–462
 - packages, 354
 - with optional dependencies, 366
 - with pear command, 354–357
- PEAR
 - go-pear.org, 351–354
 - with PHP Windows Installer, 351
 - with Unix/Linux PHP distribution, 350
- Zdebug, 466
- Zend Studio, 645–647
- installroot=DIR, 366
- instanceof operator, 4, 71
- instances, creating with new keyword, 57–58
- int stream_tell (void), 271
- int stream_write (string data), 270
- integers, 19
- Integrated Development Environment (IDE), 644
- Iterator interface, 91
- interfaces, 4, 55, 74–75
 - ArrayAccess interface, 88
 - class inheritance, 73
 - command-line interfaces, 443
 - inheritance, 75
 - Iterator interface, 91
 - Iterator Aggregate interface, 92
 - Traversable interface, 91
- intermediate code, 472
- internal (inline), 622–623
- Internal Debugger, 649
- internationalization
 - 118N, 567
 - 118Nv2, 567
 - fribidi, 566
 - idn, 567
 - Translation, 567
 - Translation2, 568
- introspection. *See* reflection
- IO_Exception, 218
- isis, 538
- ISO 8601 year format, 309
- isolating state, 451
- isset(), 16–17
- IT with blocks, HTML_Template_IT, 386–387
- IteratorAggregate interface, 92
- iterators
 - foreach() loop, 89–94
 - SQLite, 170–172

J-K

JPEG, comparing to
PNG, 320

kadm5, 574

KCachegrind (Zdebug),
468–470

Keyed-Hashing for
Message Authentication. *See* HMAC

keywords
abstract, 615–616
access, 616–617
author, 617
category, 618
copyright, 618
deprecated, 618
example, 619
filesource, 620
final, 4, 620–621
global, 621
ignore, 622
inheritdoc (inline), 622
internal (inline),
622–623
licence, 623
link, 623
link (inline), 623
name, 624
new, 57–58
package, 624, 626
param, 626–627
return, 627
see, 627
since, 628
static, 628
staticvar, 629
subpackage, 629
todo, 630
uses, 630–631
var, 631
version, 631

Knowles, Alan, 613

L

language features
character sets, 329
object oriented model,
1–5

layers, 360

layout, separating from
logic, 144–146

lazy matching, 288–289

lead, 418

libraries
GD libraries. *See* GD
libraries

libxslt, 239

libxslt library, 239

licence, 623

license element, 418–419

limitations of PHP 3 and 4,
2

limitQuery (PEAR DB),
180–181

link, 623

link (inline), 623

Linux PHP distribution,
installing PEAR,
350

list(), 28–30

LiveUser, 529

load-balancing by
session id, 398

locking files, 276–277

Log, 568

log_errors, 130

log_errors (Boolean), 203

log_errors_max_len
(Integer), 203

Log_Parser, 568

logging, 568

logic, separating from
layout, 144–146

logical errors, 197

logical operators, 34–35

login forms, 399

loop control structures
do.. while loops, 43–44
for loops, 44–45
while loops, 42

loops, controlling, 43

lowercasing (PEAR DB),
186

lzf, 599

M

macros
function declaration
macros, 495

STD_PHP_INI_
ENTRY macro
parameters, 503

TSRM, 504–505

VCWD, 496

ZEND_FETCH_
RESOURCE
macro argu-
ments, 498

ZEND_INIT_
MODULE_
GLOBALS macro
parameters, 502

zval accessor macros,
499–501

magic quotes, 198

magic_quotes_gpc, 198

magic_quotes_runtime,
198

Mail, 569
Mail_IMAP, 569
Mail_Mbox, 569
Mail_Mime, 570
Mail_Queue, 570
mailparse, 569
POP3, 570
vpopmail, 571

Mail_IMAP, 569

Mail_Mbox, 569

- Mail_Mime, 570
- Mail_Queue, 570
- mailparse, 569
- maintainers element, 418
- major versus minor
 - version versus patch level, 349
- managing variables
 - empty(), 17
 - isset(), 16–17
 - unset(), 17
- manipulating graphics
 - bar charts, 320–325
 - bot-proof submission forms, 315–320
 - exif, 326–329
- markup format,
 - HTML_Template_Flexy, 390
- matching functions, 293, 295
- Math
 - Math_Baseex, 571
 - Math_Complex, 571
 - Math_Fibonacci, 571
 - Math_Histogram, 571
 - Math_Integer, 572
 - Math_Matrix, 572
 - Math_Numerical_RootFinding, 572
 - Math_Quaternion, 573
 - Math_RPN, 573
 - Math_Stats, 573
 - Math_TrigOp, 573
 - Math_Vector, 574
- Math_Baseex, 571
- Math_Complex, 571
- Math_Fibonacci, 571
- Math_Histogram, 571
- Math_Integer, 572
- Math_Matrix, 572
- Math_Numerical_RootFinding, 572
- Math_Quaternion, 573
- Math_RPN, 573
- Math_Stats, 573
- Math_TrigOp, 573
- Math_Vector, 574
- Math_Numerical_RootFinding, 572
- Math_Quaternion, 573
- Math_RPN, 573
- Math_Stats, 573
- Math_TrigOp, 573
- Math_Vector, 574
- mb_convert_encoding(), 333
- mb_strlen, 335–337
- mb_strpos(), 340
- mb_substitute_character(), 332
- mb_substr(), 340
- mbstring extension, 330
- mbstring phpinfo() output, 331
- mbstring(), 333
- MD5 checksum, PEAR packager, 430
- md5sum attribute, 421
- MDB, 538–539
- MDB_QueryTool, 541
- MDB2, 539–540
- mbdtools, 541
- member variables,
 - symbol naming (PCS), 406
- memcache, 589
- memory
 - new memory manager, 11
 - writing extensions, 489
- Message, 544
- messages (XML RPC), 244–245
- metacharacters, 280, 283, 285
- methods
 - __toString(), 76–77
 - abstract methods, 6, 72–73
 - accessing with \$this, 59, 61–62
 - call(), 253
 - connect, 62
 - createAggregate(), 174
 - createDbConnection(), 62
 - draw(), 72
 - final methods, 75–76
 - getAll(), 189
 - getAssoc(), 188–189
 - getAttribute(), 228
 - getCol(), 188
 - getName(), 56
 - getOne(), 188
 - getRow(), 188
 - hash(), 125
 - inherited methods, E_STRICT, 442–443
 - outputObject(), 388
 - overloading, 85–87
 - POST, 115
 - raiseError, 207
 - service(), 255
 - setDbConnection(), 62
 - setName(), 56
 - singleQuery(), 168
 - static methods, 5
 - accessing with \$this, 64–65
 - symbol naming (PCS), 405
- micro-benchmarks, 477–479
- microtime(), 302
- mistakes when protecting user input, 117

- cross-site scripting, 118–119
 - global variables, 117–118
 - SQL Injection, 119–120
 - mktime(), 303
 - modifiers, 289, 293
 - modifying array elements, 25
 - module_number, 494
 - mono, 590
 - move_uploaded_file(), 142
 - moving files with rename()
 - function, 278
 - MP3_ID, 547
 - mqseries, 574
 - multi statements
 - (MySQL), 155–156
 - multi-byte strings, 330
 - multi-dimensional arrays.
 - See nested arrays
 - MySQL, 149
 - BLOB
 - inserting BLOB data, 159
 - retrieving BLOB data, 159–160
 - buffered queries, 153
 - compatibility, 445
 - connections, 151–153
 - example data, 151
 - fetching modes, 156
 - multi statements, 155–156
 - PHP interface, 150–151
 - prepared statements, 156–158
 - queries, 154–155
 - scalability, 150
 - speed, 150
 - strengths and weaknesses of, 150
 - unbuffered queries, 154
 - MySQLi, 10
 - mysqli connection functions, 152
 - mysqli fetch functions, 156
 - mysqli query functions, 154
 - mysqli_options constants, 153
- ## N
- name, 624
 - name attribute, 421–423, 428
 - name element, 417
 - naming conventions, symbols in PCS, 403
 - classes, 404
 - constants, 403
 - functions, 404
 - global variables, 404
 - member variables, 406
 - methods, 405
 - negation operators, 36
 - nested arrays,
 - accessing, 26
 - Net_CheckIP, 575
 - Net_Curl, 575
 - Net_Cyrus, 575
 - Net_Dict, 575
 - Net_Dig, 575
 - Net_DIME, 576
 - Net_DNS, 576
 - Net_Finger, 576
 - Net_FTP, 576
 - Net_GameServerQuery, 576
 - Net_Geo, 577
 - Net_Gopher, 577
 - Net_Ident, 577
 - Net_IMAP, 577
 - Net_IPv4, 577
 - Net_IPv6, 578
 - Net_IRC, 578
 - Net_LDAP, 578
 - Net_LMTP, 579
 - Net_NNTP, 579
 - Net_Ping, 579
 - Net_POP3, 579
 - Net_Portscan, 579
 - Net_Server, 580
 - Net_Sieve, 580
 - Net_SmartIRC, 580–581
 - Net_SMTP, 582
 - Net_Socket, 582
 - Net_Traceroute, 582
 - Net_URL, 582
 - Net_UserAgent_Detect, 582
 - Net_UserAgent_Mobile, 583
 - Net_Whois, 583
 - netools, 574
 - networking
 - cvsclient, 574
 - cyrus, 574
 - kadm5, 574
 - mqseries, 574
 - Net_CheckIP, 575
 - Net_Curl, 575
 - Net_Cyrus, 575
 - Net_Dict, 575
 - Net_Dig, 575
 - Net_DIME, 576
 - Net_DNS, 576
 - Net_Finger, 576
 - Net_FTP, 576
 - Net_GameServerQuery, 576

- Net_Geo, 577
 - Net_Gopher, 577
 - Net_Ident, 577
 - Net_IMAP, 577
 - Net_IPv4, 577
 - Net_IPv6, 578
 - Net_IRC, 578
 - Net_LDAP, 578
 - Net_LMTP, 579
 - Net_NNTP, 579
 - Net_Ping, 579
 - Net_POP3, 579
 - Net_Portscan, 579
 - Net_Server, 580
 - Net_Sieve, 580
 - Net_SmartIRC, 580–581
 - Net_SMTP, 582
 - Net_Socket, 582
 - Net_spread, 583
 - Net_Traceroute, 582
 - Net_URL, 582
 - Net_UserAgent_Detect, 582
 - Net_UserAgent_Mobile, 583
 - Net_Whois, 583
 - netools, 574
 - opendirectory, 583
 - tcpwrap, 584
 - uuid, 584
 - yaz, 584
 - new keyword, 57–58
 - nextId(), 185
 - nobuild, 365
 - nocompress, 365
 - notes element, 420
 - null, 23
 - numbers
 - floating point numbers, 19
 - Number_Roman, 584
 - Number_Words, 585
 - Numbers_Roman, 584
 - Numbers_Words, 585
 - numeric operators, 32
- O**
- o, 366
 - Object Cloning, 4
 - object model (PHP 5), 433
 - object-oriented features, 3–7
 - object-oriented model. *See* OO model
 - object-oriented programming. *See* OOP
 - objects, 55
 - casting (compatibility mode), 435–436
 - classes, 56
 - cloning, 4, 66–67
 - comparing (compatibility mode), 436–437
 - creating automatically, 441
 - dereference objects not returned from methods, 6
 - iterators, 89–94
 - passing to functions, 433–435
 - SimpleXML objects
 - browsing, 233–234
 - creating, 232–233
 - storing, 234
 - observer pattern, 101–103
 - obtaining PEAR DB, 176
 - oci8, 541
 - odbtcp, 541
 - oggvorbis, 595
 - OLE, 596
 - one script per function, 144
 - one script serves all, 143
 - onlyreqdeps, 366
 - OO (object oriented)
 - model, 1–2
 - design patterns, 94
 - factory patterns, 97–101
 - observer patterns, 101–103
 - singleton patterns, 97–98
 - strategy patterns, 95–96
 - versus procedural code, 480–481
 - OO applications, 70
 - OO wrappers for built-in functions, 456
 - OOP (object-oriented programming), 55
 - polymorphism, 67–69
 - opcode cache, 470
 - openal, 595
 - opendirectory, 583
 - opening tags, 221
 - OpenSSL, 272
 - operating system dependencies (PEAR), 425
 - operating system
 - differences, portability errors, 197
 - operators, 31
 - assignment operators, 32–33
 - binary operators, 31–32
 - bitwise operators, 35

- cast operators, 38
- comparison operators, 33–34
- increment/decrement operators, 37–38
- instanceof, 4, 71
- logical operators, 34–35
- negation operators, 36
- silence operators, 39
- ternary operators, 39
- unary operators, 36
- optimizing, 459
 - code, 477
 - micro-benchmarks, 477–479
 - OO versus procedural code, 480–481
 - rewriting in C, 479
 - optional attribute, 424
 - optional dependencies, 366
 - installing packages with, 366
 - PEAR, 426
 - options
 - CLI versus CGI, 509
 - short_open_tags INI, 14
 - output caching, 456
 - Cache_Lite, 400–401
 - outputObject() method, 388
 - overloading, 85
 - array access syntax, 88–89
 - iteration, 90
 - methods, 85–87
 - property overloading, 85–87

P

- pipes, 266
- package, 624, 626
- package element, 417
- package information (package.xml), 417–419
- package.xml, 416, 431
 - package information, 417–419
 - release information, 419–422
- packages
 - building
 - PEAR Example (HelloWorld), 411–414
 - regression tests (PEAR), 416
 - tarballs (PEAR), 414
 - verification (PEAR), 414–415
 - Cache_Lite, 399–401
 - installing, 354
 - with optional dependencies, 366
 - with pear command, 354–357
 - PEAR, 346
 - releasing packages, 428
 - PEAR Release Process, 430
- packaging, PEAR
 - packager, 430
 - MD5 checksum, 430
 - package.xml, 431
 - source analysis, 430
 - tarballs, 431
- Pager, 560
- Pager_Sliding, 560
- panda, 599
- Paradox, 542
- parallel PEAR
 - installations, 362
- param, 626–627
- parameters
 - class type hints in
 - function parameters, 82–83
 - declaring function parameters, 52
- parent, 70
- parse errors, 192–194
- parsing
 - command-line options, 512–515
 - date formats, 313–314
 - XML, 222
 - DOM, 226–229
 - PEAR. *See* PEAR SAX, 222–226
- partial caching, 475
- passing
 - objects to functions, 433–435
 - by reference, 52
 - by value, 52
- password files, Auth, 393–394
- passwords, protecting scripts, 127–129
- pattern syntax, 280
- patterns, 279. *See also* design patterns
- payment
 - cybercash, 585
 - cybermut, 585
 - Payment_Clieop, 586
 - Payment_DTA, 586
 - Payment_Process, 586
 - spplus, 586
 - TCLink, 586

- Payment_Clieop, 586
- Payment_DTA, 586
- Payment_Process, 586
- Payment_splusplus, 586
- PCRE `phpinfo()`
 - output, 279
- PCS (PEAR's Coding Standard), 403
 - indentation, 406–407
 - symbol naming, 403
 - classes, 404
 - constants, 403
 - functions, 404
 - global variables, 404
 - member variables, 406
 - methods, 405
- PEAR (PHP Extension and Application Repository), 200–201, 345, 587
 - Auth, 392, 398
 - DB and user data, 394–396
 - password files, 393–394
 - scalability
 - considerations, 397–398
 - security
 - considerations, 396–397
 - building packages
 - PEAR Example, HelloWorld, 411–414
 - regression tests, 416
 - tarballs, 414
 - verification, 414–415
 - caching, 399–401
 - CLI environment, 408–410
 - Crypt_HMAC, 124–127
 - dependencies, 423–424
 - dep, 423
 - dependency types, 424–425
 - deps, 423
 - examples, 426
 - optional dependencies, 426
 - reasons for avoiding, 425–426
 - embedded
 - in UNIX shell scripts, 409
 - in Windows .BAT files, 410
 - encapsulating includes, 411
 - error handling, 206–207, 212, 411
 - catching errors, 207
 - `delExpect()`, 215
 - `expectError()`, 214–215
 - `isError()`, 207
 - `popErrorHandler()`, 214
 - `popExpect()`, 215
 - `pushErrorHandler()`, 213–214
 - `raiseError()`, 209
 - raising errors, 207–208
 - `setErrorHandling()`, 212
 - `throwError()`, 208
 - error modes
 - PEAR_ERROR_CALLBACK, 213
 - PEAR_ERROR_DIE, 213
 - PEAR_ERROR_PRINT, 213
 - PEAR_ERROR_TRIGGER, 213
 - files, including, 410–411
 - forms, 398
 - HTML_QuickForm, 398
 - login forms, 399
 - receiving data, 399
 - including C code in packages, 428
 - installing
 - go-pear.org, 351–354
 - packages, 354–357
 - with PHP Windows Installer, 351
 - with Unix/Linux PHP distribution, 350
 - package.xml, 416
 - package information, 417–419
 - release information, 419–422
 - packages, 346
 - parallel PEAR
 - installations, 362
 - parsing XML, 234
 - XML_RSS, 236–239
 - XML_Tree, 235–236
- PEAR_Error class
 - `addUserInfo()`, 211
 - `getCallback()`, 211
 - `getcode()`, 211
 - `getMessage()`, 211
 - `getMode()`, 211
 - `getType()`, 211
 - `getUserInfo()`, 211

- PEAR_Error
 - constructor, 210–211
- PEAR_Frontend_Gtk, 587
- PEAR_Frontend_Web, 587
- PEAR_Info, 587
- PEAR_PackageFileManager, 588
- release versioning, 408
- releases, 346–347
- releasing packages, 428
- SOAP, 252
 - Google, 252–254
 - SOAP clients, 255–256
 - SOAP servers, 254–255
- string substitutions, 427
- template systems. *See*
 - template systems
- uploading, 432
- version numbers, 347
 - comparing, 348–349
 - version number format, 347
- pear command, 354–357, 363
- pear config-get, 376
- pear config-set, 376
- pear config-show, 358, 376–377
- PEAR data directory (data_dir), 359
- PEAR DB, 176
 - abstracted errors, 186
 - error codes, 186–187
 - error handling, 187
- connections
 - configuration options, 179
 - DSNs, 178
 - establishing, 178–179
- convenience methods, 188
 - getAll(), 189
 - getAssoc(), 188–189
 - getCol(), 188
 - getOne(), 188
 - getRow(), 188
- executing queries, 180
 - execute(), 181–182
 - limitQuery, 180–181
 - prepare(\$query), 181–182
 - simpleQuery(), 182
- features that are abstracted, 177–178
- fetching results, 182
 - configuring fetch modes, 183
 - fetch modes, 182
 - fetchInto(), 183
 - fetchRow(), 183
 - using your own result class, 183–184
- obtaining, 176
- portability, 185
 - count deleted rows, 185
 - count number of rows, 186
 - empty string handling, 186
 - errors, 186
 - lowercasing, 186
 - trimming data, 186
- sequences, 184–185
 - createSequence(), 185
 - dropSequence(), 185
 - nextId(), 185
- strengths and weaknesses of, 177
- PEAR documentation
 - directory (doc_dir), 359
- pear download, 375
- PEAR executables
 - directory (bin_dir), 359
- pear help, 355
- pear help config-set, 362
- pear help options, 355
- pear info, 369–370
- pear install, 364–367
- PEAR Installer, 354
 - commands, 364
 - pear config-get, 376
 - pear config-set, 376
 - pear config-show, 376–377
 - pear download, 375
 - pear info, 369–370
 - pear install, 364–367
 - pear list, 368–369
 - pear list-all, 370
 - pear list-upgrades, 370
 - pear remote-info, 375
 - pear remote-list, 374–375
 - pear search, 373–374

- pear uninstall, 373
- pear upgrade, 371–372
- pear upgrade-all, 372
- shortcuts, 377
- configuration layers, 360–361
- configuration parameters, 358, 360–364
- environment variables, 409
- front-ends, 378–3381
 - CLI Installer, 378
 - Gtk Installer, 378–381
- PEAR installer cache directory (cache_dir), 359
- pear list, 368–369
- pear list-all, 370
- pear list-upgrades, 370
- PEAR main directory (php_dir), 359
- PEAR package dependencies, 424
- PEAR packager, 430
 - MD5 checksum, 430
 - package.xml, 431
 - source analysis, 430
 - tarballs, 431
- PEAR project, 345
- PEAR Release Process, 429
 - creating packages, 430
 - proposals, 429
 - tarballs, 430
 - tests, 430
 - uploading release, 430
 - voting process, 429
- pear remote-info, 375
- pear remote-list, 374–375
- pear search, 373–374
- PEAR Server (master_server), 360
- pear uninstall, 373
- pear upgrade, 371–372
- pear upgrade-all, 372
- PEAR username/PEAR password (username/password), 360
- PEAR_Error constructor, 210–211
- PEAR_Error class, 210–211
- PEAR_ERROR_CALLBACK, 213
- PEAR_ERROR_DIE, 213
- PEAR_ERROR_PRINT, 213
- PEAR_ERROR_TRIGGER, 213
- PEAR_Frontend_Gtk, 587
- PEAR_Frontend_Web, 587
- PEAR_Info, 587
- PEAR_PackageFileManager, 588
- PEAR's Coding Standard. *See* PCS
- pear-config, 427
- PECL (PHP Extension Community Library), 345
- PECL_Gen, 601
- performance, 449, 456
 - APC, 470
 - benchmarking, 457
 - ApacheBench, 457–458
 - Seige, 458–459
- testing versus real traffic, 459
- caching, 451–453
 - call caching, 456
 - compiled templates, 456
 - database query/result caching, 453–455
 - output caching, 456
- generalization, 456
- OO wrappers for built-in functions, 456
- profiling with APD, 461
 - analyzing trace data, 462–465
 - installing, 461–462
- profiling with
 - Zdebug, 465
 - installing, 466
 - KCachegrind, 468–470
 - tracing script execution, 466–467
- profiling with Zend Studio's Profiler, 459–461
- state, 450
 - isolating, 451
 - sessions, 450
- ZPS, 470
 - Automatic Optimization, 471–472
 - Compiled Code Caching, 472–473
 - content compression, 476
 - dynamic content caching, 473–475

- Performance Profiler
(Zend Studio),
651–652
- perl, 590
- Perl extension, 11
- PHP
 - apd, 588
 - bcompiler, 589
 - embedding code in
HTML, 112–114
 - ffi, 589
 - Inline_C, 589
 - memcache, 589
 - mono, 590
 - perl, 590
 - PHP_CompatInfo, 591
 - PHP_Fork, 591
 - PHP_Parser, 592
 - PHPDoc, 590
 - PHPUnit, 590
 - PHPUnit2, 590
 - python, 592
 - tools
 - crack, 600
 - fann, 601
 - PECL_Gen, 601
 - PhpDocumentor, 601
 - SPL, 602
 - Valkyrie, 602
 - Validate, 593
 - Var_Dump, 593
 - vld, 593
 - Xdebug, 594
 - Zend. *See* Zend, 644
- PHP 3, limitations of, 2
- PHP 4
 - changing to PHP 5
compatibility
mode. *See* com-
patibility mode
 - limitations of, 2
 - object model, 433
 - passing objects to
functions, 434
- PHP 5
 - \$this, 437, 439–440
 - get_class(), 440–441
 - object model, 433
 - passing objects to
functions, 434
- PHP code directory, 353
- PHP configuration differ-
ences, portability
errors, 197
- PHP dependencies
(PEAR), 424
- PHP document
 - with error class, 635
 - with Sum class,
635, 637
 - with SumNumberEle-
ments class, 637
 - with SumNumbers
class, 637–638
- PHP errors, 201
 - custom error
handlers, 204
 - error levels, 201–202
 - error reporting,
202, 204
 - silencing errors,
205–206
- PHP Extension and
Application Reposi-
tory. *See* PEAR
- PHP Extension Commu-
nity Library. *See*
PECL
- PHP extension directory
(ext_dir), 359
- PHP filter utility,
520–525
- php index() function, 165
- PHP interfaces
 - MySQL, 150–151
 - SQLite, 162
 - error handling, 163
 - setting up data-
bases, 162
 - simple queries,
162–165
 - transactions,
164–165
 - triggers, 165
- PHP sessions,
134–137, 396
- PHP Windows Installer,
installing PEAR,
351
- php.ini, CLI versus
CGI, 510
- PHP_CompatInfo, 591
- PHP_Fork, 591
- PHP_MINIT_FUNCTION(),
495
- PHP_Parser, 592
- phpapi attribute, 422
- PHPDoc, 590
- phpDocumentor tool,
601, 613, 633–634,
638, 640
- comments, 614
- parameters, 633
- php document with
error class, 635
- php document with
Sum class,
635, 637
- php document with
SumNumberEle-
ments class, 637

- php document with
 - SumNumbers
 - class, 637–638
- tag references, 615
 - abstract, 615–616
 - access, 616–617
 - author, 617
 - category, 618
 - copyright, 618
 - deprecated, 618
 - example, 619
 - filesource, 620
 - final, 620–621
 - global, 621
 - ignore, 622
 - inheritdoc (inline), 622
 - internal (inline), 622–623
 - licence, 623
 - link, 623
 - link (inline), 623
 - name, 624
 - package, 624, 626
 - param, 626–627
 - return, 627
 - see, 627
 - since, 628
 - static, 628
 - staticvar, 629
 - subpackage, 629
 - todo, 630
 - uses, 630–631
 - var, 631
 - version, 631
- phpfilter, 521
- phpinfo() output, 279
- .phpt, 416
- PHPUnit, 590
- PHPUnit2, 590
- placeholders, 384
- platform attribute, 421
- PNG, comparing to
 - JPEG, 320
- polymorphism, 67–69
- POP3, 570
- popen(), 264, 517
- portability, 200
 - PEAR DB, 185
 - count deleted
 - rows, 185
 - count number of
 - rows, 186
 - empty string
 - handling, 186
 - errors, 186
 - lowercasing, 186
 - trimming data, 186
 - portability errors
 - dealing with
 - portability, 200
 - operating system
 - differences, 197
 - PHP configuration
 - differences, 197
 - portability tools, 200–201
 - SAPI differences, 199
 - portability tools, 200–201
 - POST method, 115, 137
 - pprofp options, 465
 - Preferred Package Stage
 - (preferred_state), 359
 - preferred_state, 374
 - preg_grep(), 295
 - preg_match(), 279, 293
 - preg_match_all(), 294–295
 - preg_replace(), 296, 298, 478
 - preg_replace_callback(), 296
 - preg_split(), 299
- PREG_SPLIT_DELIM_
 - CAPTURE, 300
- PREG_SPLIT_NO_
 - EMPTY, 300
- PREG_SPLIT_OFFSET_
 - CAPTURE, 301
- prepare(\$query), 181–182
- prepared statements
 - (MySQL), 156
 - binding variables, 156–158
- proc_open(), 265
- procedural code versus
 - OO, 480–481
- process control, shell
 - scripts, 516
 - exec, 517–518
 - forking, 517
 - processes, 516
 - signals, 519–520
- process_children(), 228
- processes, shell
 - scripts, 516
- processing FSM, 594
- product support for Zend
 - Studio, 652–653
- Profiler (Zend Studio), 459–461
- profiling, 459
 - with APD, 461
 - analyzing trace
 - data, 462–465
 - installing, 461–462
 - Studio Server, config-uring, 650–651
 - with Zdebug, 465
 - installing, 466
 - KCachegrind, 468–470
 - tracing script execu-tion, 466–467

- with Zend Studio's Profiler, 459–461
- prog, 422
- program input/output, 264
- filedescriptors, 265
- files, 266–267
- pipes, 266
- popen(), 264
- proc_open(), 265
- programming errors, 192
 - eval(), 193
 - parse errors, 192–194
 - syntax errors, 192
- projects, creating (Zend Studio Client), 648
- prompt attribute, 428
- properties
 - accessing with \$this, 59–61
 - overloading, 85–87
 - static properties, accessing with \$this, 62–64
- proposals, PEAR Release Process, 429
- protecting
 - scripts, 120
 - Crypt_HMAC, 124–127
 - error handling, 129–130
 - HMAC verification, 122–124
 - input filters, 127
 - input validation, 120–122
 - passwords, 127–129
 - user input, 117–120
- provides element, 422
- ps, 599
- public, E_STRICT, 441
- python, 592

Q

- q, 356
- queries
 - buffered queries (MySQL), 153
 - MySQL, 154–155
 - PEAR DB
 - execute(), 181–182
 - executing, 180
 - limitQuery, 180–181
 - prepare(\$query), 181–182
 - simpleQuery(), 182
 - simple queries (SQLite), 162–165
 - unbuffered queries (MySQL), 154
- QueryCacheStrategy-Wrapper class, 455
- querying
 - database structure (SQLite), 175–176
 - functions (SQLite), 168
- question mark (?), 39
- quotes
 - double quotes (“ ”), strings, 19–20
 - single quotes (‘ ’), strings, 20

R

- R DIR, 366
- radius, 530
- raiseError method, 207, 209
- raising PEAR errors, 207–208
- RDF Site Summary, Really Simple Syndication. *See* RSS
- reading array values, 25
- Real numbers. *See* floating-point numbers

- receiving data (forms), 399
- references, foreach(), 8
- reflection, 103
 - examples of reflection, 106–107
 - implementing
 - delegation design patterns, 107–109
 - reflection API, 103–105
 - reflection API, 103–105
- register key, 116
- register_argc_argv, 198
- register_globals, 117, 198
- registering
 - resources types, 494–495
 - Zend Studio, 645–647
- register-only, 365
- registration (user input), 114–117
- regression tests, building packages in PEAR, 416
- regular equality
 - operators (==), 42
- regular expressions, 279
 - functions, 293
 - matching functions, 293, 295
 - replacement
 - functions, 295–298
 - splitting strings, 299–301
- syntax, 279–280
 - escape sequences, 285, 288
 - lazy matching, 288–289
 - metacharacters, 280, 283, 285

- modifiers, 289, 293
- pattern syntax, 280
- rel attribute, 424
- release element, 419
- release information,
 - package.xml, 419–422
- release versioning (PEAR), 408
- releases (PEAR), 346–347
- releasing packages (PEAR), 428
- Remote Debugger, 649–650
 - Studio Server, configuring, 650–651
- Remote Procedure Calls. *See* RPC
- removing
 - files, 277–278
 - resources, 498–499
- rename(), 278
- renaming files, 278
- replacement functions, 295–298
- reporting PHP errors, 202, 204
- requests, XML RPC, 245–246
- require_once, 410
- resource-enabled PHP functions, 497
- resources
 - accessing, 497–498
 - removing, 498–499
 - wrapping third-party extensions, 493–495
- responses, XML RPC, 246
- rest(), 28
- result set-related functions, 172–173

- retrieving
 - BLOB data (MySQL), 159–160
 - date and time information, 301–305
- retrospection, XML RPC, 249–250
- return, 627
- returning values
 - from PHP functions, writing extensions, 490
 - by reference, 51
 - by value, 50
- rewriting code in C, 479
- role attribute, 421
- role element, 418
- root element, 220, 417
- root nodes, 220
- RPC (Remote Procedure Calls), 244
- RSS (RDF Site Summary, Really Simple Syndication), 236
- RSS files, 235
- run-tests command, 416
- runtime errors, 201

S

- S, 357
- s, 357, 365
- Sablotron extension, 9
- SAPI (Server API), 507
- SAPI differences, portability errors, 199
- sasl, 530
- SAX (Simple API for XML), 9, 222
 - parsing XML, 222–226

- scalability
 - Auth, 397–398
 - load-balancing by
 - session id, 398
 - session storage, 398
 - MySQL, 150
 - Science_Chemistry, 594
 - script execution, tracing with Zdebug, 466–467
 - scripts
 - cross-site scripting, 118–119
 - one script per function, 144
 - one script serves all, 143
 - protecting, 120
 - Crypt_HMAC, 124–127
 - error handling, 129–130
 - HMAC verification, 122–124
 - input filters, 127
 - input validation, 120–122
 - passwords, 127–129
 - sections, 384
 - security
 - Auth, 396
 - Auth_HTTP, 397
 - disabling
 - session.trans_sid, 396
 - HTTPS, 397
 - protecting user input, 117–120
 - see, 627
 - Seige, 458–459
 - self, 70

- self_concat(), 485, 490–491
- separating logic from layout, 144–146
- sequences
 - escape sequences, 285, 288
 - PEAR DB, 184–185
 - createSequence(), 185
 - dropSequence(), 185
 - nextId(), 185
- servers, XML RPC, 250–252
- service(), 255
- Services_ExchangeRates, 603
- Services_Weather, 603
- session id, load-balancing, 398
- session storage, Auth, 398
- session.trans_sid, disabling, 396
- session_destroy(), 136
- session_start(), 135
- session_write_close(), 136
- sessions, 134–137, 395, 450
- set_error_handler(), 130
- set_exception_handler(), 7
- setcookie(), 131
- setCurrentBlock(), 386
- setDbConnection(), 62
- setName(), 56
- shared lock, 277
- shell scripts
 - CLI, 508. *See also* CLI PHP shell scripts guidelines for writing, 508
 - error messages, 516
 - exit code, 516
 - usage messages, 515–516
- parsing command-line options, 512–515
- process control, 516
 - exec, 517–518
 - forking, 517
 - processes, 516
 - signals, 519–520
 - writing comments, 15
- short tags, 113
- short_open_tags INI option, 14
- short-circuit evaluation, 35
- shortcuts, command-line, 377
- SIGINT, 519
- signals, shell scripts, 519–520
- Signature Handling Program (sig_bin), 360
- Signature Key Directory (sig_keydir), 360
- Signature Key Id (sig_keyid), 360
- Signature Type (sig_type), 360
- silence operators, 39
- silencing PHP errors, 205–206
- Simple API for XML. *See* SAX
- simple queries (SQLite), 162–165
- simpleQuery(), 182
- SimpleXML extension, 9–10, 222, 231
 - browsing SimpleXML objects, 233–234
 - creating SimpleXML objects, 232–233
 - storing SimpleXML objects, 234
- since, 628
- single quotes ('), strings, 20
- singleQuery(), 168
- singleton pattern, 97–98
- SML, parsing, 222
- SOAP, 10, 252, 603
 - clients, 255–256
 - Google, 252–254
 - servers, 254–255
- SOAP extension, 257–259
- SOAP_Interop, 603
- soft, 365
- source analysis, PEAR packager, 430
- spaces, converting to UCS-2BE, 339
- speed (MySQL), 150
- SPL, 602
- splitting strings, 299–301
- spread, 583
- Spreadsheet_Excel_Writer, 547
- SQL Injection, protecting user input, 119–120
- SQL_Parser, 542
- SQLite, 160, 542
 - aggregate UDFs, 173–174
 - best areas of use, 161
 - character encoding, 174
 - fetching data, 168–170
 - guidelines for using, 176
 - iterators, 170–172
 - PHP interfaces, 162
 - error handling, 163

- setting up
 - databases, 162
- simple queries, 162–165
- transactions, 164–165
- triggers, 165
- querying
 - database structure, 175–176
 - functions, 168
 - result set-related functions, 172–173
 - strengths and weaknesses of, 160–161
 - tuning, 174–175
 - UDFs, 165–168
- SQLite Database
 - constructor, 162
- sqlite escape string(), 170
- SSL, support for HTTP and FTP, 272
- stgrab, 598
- state, 450
 - isolating, 451
 - sessions, 450
- state element, 419
- static, 628
- static members, 5, 62
- static methods, 5
 - accessing with \$this, 64–65
- static properties,
 - accessing with \$this, 62–64
- static variables, 53
- staticvar, 629
- STD_PHP_INI_ENTRY
 - macro parameters, 503
- storing SimpleXML
 - objects, 234
- str_replace(), 478
- strategy pattern, 95–96
- Strategy wrapper, 455
- STREAM_NOTIFY_
 - AUTH_REQUIRED, 275
 - STREAM_NOTIFY_AUTH_RESULT, 275
 - STREAM_NOTIFY_CONNECT, 275
 - STREAM_NOTIFY_FAILURE, 276
 - STREAM_NOTIFY_FILE_SIZE_IS, 275
 - STREAM_NOTIFY_MIME_TYPE_IS, 275
 - STREAM_NOTIFY_PROGRESS, 276
 - STREAM_NOTIFY_REDIRECTED, 276
- Stream_SHM, 595
- Stream_Var, 596
- URL streams, 271–276
- user streams, 270
 - boolean stream_eof (void), 270
 - boolean
 - stream_flush (void), 271
 - boolean
 - stream_open, 270
 - boolean stream_seek (int offset, int whence), 271
 - int stream_tell (void), 271
 - int stream_write (string data), 270
 - string stream_read (int count), 270
 - void stream_close (void), 270
 - zlib_filter, 596
- streams layer, 261
- strengths
 - of MySQL, 150
 - of SQLite, 160–161
- string offsets, accessing, 21–22
- string stream_read (int count), 270
- string substitutions (PEAR), 427
- strings, 19
 - accessing string offsets, 21–22
 - binary strings, 488
 - double quotes (“ ”), 19–20
 - here-docs, 21

incrementing
 strings, 38
 single quotes ('), 20
 splitting strings,
 299–301
 substrings,
 replacement
 functions, 296
 strlen, 48
 strpos(), 338
 strrpos(), 446–447
 strstr(), 446–447
 strtotime(), 314
 structures
 Games_Chess, 596
 OLE, 596
 Structures_DataGrid,
 597
 Structures_Graph, 597
 Text_Statistics, 597
 Tree, 597
 Structures_DataGrid, 597
 Structures_Graph, 597
 Studio Client Package
 (Zend Studio), 645
 stylesheets, 239
 sub-commands, help, 355
 subjects, 279
 subpackage, 629
 substrings, 296
 Sum class in PHP
 document, 635, 637
 summary element, 417
 SumNumberElements
 class in PHP
 document, 637
 SumNumbers class in
 PHP document,
 637–638
 superglobals, 18
 support
 for Windows 95, 11

for Zend Studio,
 652–653
 Suraski, Zeev, 643
 switch statements, 41–42
 symbols
 naming in PCS, 403
 classes, 404
 constants, 403
 functions, 404
 global variables, 404
 member variables,
 406
 methods, 405
 undefined symbols, 194
 array indexes, 196
 functions and
 classes,
 196–197
 logical errors, 197
 variables and
 constants, 195
 syntax
 array access syntax,
 overloading,
 88–89
 regular expressions,
 279–280
 escape sequences,
 285, 288
 lazy matching,
 288–289
 metacharacters,
 280, 283, 285
 modifiers, 289, 293
 pattern syntax, 280
 syntax errors, 192
 system layers, 360
 System_Command, 532
 System_ProcWatch, 598
 System_Socket, 598
 systems
 statgrab, 598

System_ProcWatch,
 598
 System_Socket, 598

T

tag references, 615
 abstract, 615–616
 access, 616–617
 author, 617
 category, 618
 copyright, 618
 deprecated, 618
 example, 619
 filesource, 620
 final, 620–621
 global, 621
 ignore, 622
 inheritdoc (inline), 622
 internal (inline),
 622–623
 licence, 623
 link, 623
 link (inline), 623
 name, 624
 package, 624, 626
 param, 626–627
 return, 627
 see, 627
 since, 628
 static, 628
 staticvar, 629
 subpackage, 629
 todo, 630
 uses, 630–631
 var, 631
 version, 631
 tags, 632
 <?, 113
 closing tags, 221
 Flexy markup
 tags, 390

- opening tags, 221
- short tags, 113
- tarballs, 416
 - building packages in PEAR, 414
 - creating, 431
 - PEAR Release Process, 430
- TCLink, 586
- tcpwrap, 584
- template systems, 383–384
 - HTML_Template_Flexy, 387
 - basic Flexy template, 387–388
 - blocks, 388–389
 - HTML attribute handling, 390–391
 - HTML element handling, 391–392
 - markup format, 390
 - HTML_Template_IT, 384
 - basic IT template, 384–385
 - IT with blocks, 386–387
 - placeholder syntax, 384
- templates, 384, 638
 - compiled templates, 456
- temporary files, 278–279
- ternary operators, 39
- Test base directory, 353
- testing
 - extensions, 501
 - PEAR Release Process, 430
- performance versus real traffic, 459
- regression tests, building packages in PEAR, 416
- text
 - aligning, 325
 - enchant, 599
 - lzf, 599
 - panda, 599
 - ps, 599
 - Text_Diff, 600
 - Text_Password, 600
 - Text_Wiki, 600
 - xdiff, 600
- Text_Diff, 600
- Text_Password, 600
- Text_Statistics, 597
- Text_Wiki, 600
- third-party extensions
 - accessing resources, 497–498
 - wrapping, 492
 - file functions, 496
 - motivation, 492–493
 - registering
 - resources types, 494–495
 - resources, 493–494
- Thread-Safe Resource Manager. *See* TSRM
- throw exceptions, 216–218
- throwError(), 207–208
- tidy, 560
- Tidy extension, 10
- time
 - daylight savings time, 310
 - formatting, 305, 309–312
 - ISO 8601 year format, 309
 - retrieving date and time information, 301–305
 - showing local time in other time zones, 312
 - time zones, showing local time in other time zones, 312
- time(), 303
- tmpfile(), 278
- todo, 630
- tools. *See also* command-line tools
 - for PHP
 - crack, 600
 - fann, 601
 - PECL_Gen, 601
 - PhpDocumentor, 601
 - SPL, 602
 - Valkyrie, 602
 - phpDocumentor, 613, 633–634, 638, 640
 - parameters, 633
 - PHP document with error class, 635
 - PHP document with Sum class, 635, 637
 - PHP document with SumNumberElements class, 637
 - PHP document with SumNumbers class, 637–638
 - portability tools, 200–201
 - trace data, analyzing (APD), 462–465

tracing script execution
 (Zdebug), 466–467
 track_errors (Boolean),
 203
 transactions (SQLite),
 164–165
 Translation, 567
 Translation2, 568
 transparency, 321
 Traversable interface, 91
 traversing arrays, 30
 each(), 28
 foreach, 26–27
 list(), 28–30
 rest(), 28
 Tree, 597
 trees, creating DOM
 trees, 229–231
 triggers (SQLite), 165
 trimming data (PEAR
 DB), 186
 TrueType, 324
 try exceptions, 216–218
 TSRM (Thread-Safe
 Resource Manager),
 504–505
 tuning SQLite, 174–175
 type attribute, 422–423
 type specifiers, 487

U

-u, 357
 UCS-2BE, 337
 converting spaces, 339
 UDDI, 603
 UDFs (user-defined func-
 tions), 48–49, 165
 aggregate UDFs
 (SQLite), 173–174
 SQLite, 165–168
 unary operators, 36

unbuffered queries
 (MySQL), 154
 undefined symbols, 194
 array indexes, 196
 functions and classes,
 196–197
 logical errors, 197
 variables and
 constants, 195
 Unicode, 330
 uniform resource identifi-
 cator (URI), 178
 Unix file mask (umask),
 360
 Unix PHP distribution,
 installing PEAR,
 350
 UNIX timestamp, 301
 unlink(), 277
 unlinking files from
 directories, 277
 unserialize_callback_func,
 198
 unset(), 17
 upload directory, 139
 uploading
 files, 137–142
 PEAR, 432
 uploading release, PEAR
 Release Process, 430
 URI (uniform resource
 identifier), 178
 URL streams, 271–276
 usage messages, shell
 scripts, 515–516
 user element, 418
 user information,
 114–117, 394–396
 user input, 114–117
 CLI PHP shell scripts
 environments, 510
 protecting, 117–120

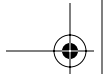
user layers, 360
 user streams, 270
 boolean stream_eof
 (void), 270
 boolean stream_flush
 (void), 271
 boolean stream_open,
 270
 boolean stream_seek
 (int offset, int
 whence), 271
 int stream_tell (void),
 271
 int stream_write
 (string data), 270
 string stream_read (int
 count), 270
 void stream_close
 (void), 270
 user-defined functions.
 See UDFs
 uses, 630–631
 UTF-8 characters, 337
 UTF-8 encoded output,
 335
 uuid, 584

V

-V, 356–357
 Validate, 593
 Valkyrie, 602
 var, 631
 E_STRICT, 441
 Var_Dump, 593
 variables, 15
 binding variables
 (MySQL),
 156–158
 data types, 18
 declaring, 15

- global variables,
 - protecting user input, 117–118
 - indirect references
 - to, 16
 - managing
 - empty(), 17
 - isset(), 16–17
 - unset(), 17
 - member variables,
 - symbol naming (PCS), 406
 - PEAR Installer
 - environment variables, 409
 - static variables, 53
 - superglobals, 18
 - undefined symbols, 195
 - VCWD (virtual current working directory), 496
 - verbose, 356
 - version, 631
 - version number format (PEAR), 347
 - version numbers, 149
 - PEAR, 347
 - comparing version numbers, 348–349
 - version number format, 347
 - version_compare()
 - function, 408
 - versioning standards (PEAR), 347
 - VFS, 549
 - virtual current working directory. *See* VCWD
 - vld, 593
 - void stream_close (void), 270
 - voting process, PEAR
 - Release Process, 429
 - vpopmail, 571
- W**
- weaknesses
 - of MySQL, 150
 - of SQLite, 161
 - web Server API. *See* SAPI
 - Web Services
 - Services_ExchangeRates, 603
 - Services_Weather, 603
 - SOAP, 603
 - SOAP_Interop, 603
 - UDDI, 603
 - XML_RPC, 604
 - while loops, 42
 - win32std, 533
 - Windows 95, support
 - for, 11
 - wrappers
 - OO wrappers for built-in functions, 456
 - Strategy wrapper, 455
 - wrapping third-party
 - extensions, 492
 - file functions, 496
 - motivation, 492–493
 - registering resources
 - types, 494–495
 - resources, 493–494
 - writing
 - comments, 14
 - extensions, 484–489
 - memory management, 489
 - reasons for writing your own, 483
 - returning values
 - from PHP functions, 490
 - self_concat(), 490–491
 - resource-enabled PHP functions, 497
 - shell scripts
 - error messages, 516
 - exit code, 516
 - usage messages, 515–516
 - WSDL, 257
- X**
- Xdebug, 594
 - xdiff, 600
 - XHTML 1.0 Transitional DTD, 221
 - XHTML documents, 220
 - XML, 8
 - communicating
 - with, 244
 - converting XSLT, 239–243
 - DOM trees, creating, 229–231
 - DTD files, 220
 - entities, 221
 - package.xml, 416
 - package information, 417–419
 - release information, 419–422
 - parsing
 - DOM, 226–229
 - PEAR. *See* PEAR
 - SAX, 222–226
 - RSS files, 235
 - SimpleXML extension, 9–10, 222, 231
 - browsing
 - SimpleXML objects, 233–234

- creating SimpleXML objects, 232–233
 - storing SimpleXML objects, 234
 - XML_Beautifier, 604
 - XML_CSSML, 604
 - XML_DTD, 604
 - XML_fo2pdf, 605
 - XML_FOAF, 605
 - XML_HTMLSax, 605
 - XML_image2svg, 606
 - XML_NITF, 606
 - XML_Parser, 606
 - XML_RDDL, 607
 - XML_RSS, 607
 - XML_SaxFilters, 607
 - XML_Serializer, 608
 - XML_sql2xml, 608
 - XML_Statistics, 609
 - XML_SVG, 609
 - XML_svg2image, 609
 - XML_Transformer, 609
 - XML_Tree, 609
 - XML_Util, 610
 - XML_Wddx, 610
 - XML_XPath, 610
 - XML_XSLT_Wrapper, 611
 - XML_XUL, 611
 - XPath, 229
 - XML attributes, Flexy, 391
 - XML declarations, 220
 - XML RPC
 - clients, 247–249
 - faults, 246–247
 - messages, 244–245
 - requests, 245–246
 - responses, 246
 - retrospection, 249–250
 - servers, 250–252
 - XML_Beautifier, 604
 - XML_CSSML, 604
 - XML_DTD, 604
 - XML_fo2pdf, 605
 - XML_FOAF, 605
 - XML_HTMLSax, 605
 - XML_image2svg, 606
 - XML_NITF, 606
 - XML_Parser, 606
 - XML_RDDL, 607
 - XML_RPC, 604
 - XML_RPC_decode(), 249
 - XML_RSS, 236–239, 607
 - XML_SaxFilters, 607
 - XML_Serializer, 608
 - XML_sql2xml, 608
 - XML_Statistics, 609
 - XML_SVG, 609
 - XML_svg2image, 609
 - XML_Transformer, 609
 - XML_Tree, 235–236, 609
 - XML_util, 610
 - XML_Wddx, 610
 - XML_XPath, 610
 - XML_XSLT_Wrapper, 611
 - XML_XUL, 611
 - xmlrpc_error_number (Integer), 203
 - xmlrpc_errors (Boolean), 203
 - xmms, 533
 - XSL extension, 9
 - XSLT (eXtensible Stylesheet Sheet Language Transformations), 9, 239
- Y-Z**
- y2K_compliance, 198
 - yaz, 584
 - Z, 365
 - Zdebug, 465
 - installing, 466
 - KCachegrind, 468–470
 - tracing script execution, 466–467
 - Zend, 643
 - Zend Engine, 11, 494
 - Zend Optimizer, 471
 - Zend Performance Suite. *See* ZPS
 - Zend Performance Suite Console, 473
 - Zend Server Center, 645
 - Remote Debugger and Profiling, 650–651
 - Zend Studio, 644
 - client server configurations, 645
 - debuggers, 648
 - Debug URL, 650
 - Internal Debugger, 649
 - Remote Debugger, 649–650
 - editing files, 647
 - features of, 653–654
 - installing, 645–647
 - Performance Profiler, 651–652
 - product support, 652–653
 - Profiler, 459–461
 - registering, 645–647
 - Zend Studio Client, 460, 644
 - projects, 648
 - Zend Studio Client Quick Start*, 643



Zend Studio Server, 644
 configuring for Remote
 Debugger and
 profiling, 650–651
ZEND_FETCH_
 RESOURCE macro
 arguments, 498
ZEND_INIT_MODULE_
 GLOBALS macro
 parameters, 502
zend_list_delete(), 499
ZEND_NUM_ARGS(),
 487
ZEND_REGISTER_RES
 OURCE macro
 arguments, 496
zendapi attribute, 422
zip, 548
zlib_filter, 596
ZPS (Zend Performance
 Suite), 470
 Acceleration mode, 472
 Automatic Optimiza-
 tion, 471–472
 Compiled Code
 Caching, 472–473
 content compression,
 476
 dynamic content
 caching, 473–475
zts attributes, 422
zval accessor macros,
 499–501

