

# Содержание

Об авторах	22
<b>Введение</b>	<b>24</b>
Почему именно PHP?	26
Для кого предназначена эта книга	26
Темы, которые рассматриваются в данной книге	26
Как организована эта книга	27
Что требуется для использования данной книги	29
Использование командной строки	30
Соглашения	31
Исходный код	31
Ошибки	32
p2p.wrox.com	32
<b>Глава 1. Установка и использование PHP</b>	<b>34</b>
Истоки PHP	35
Установка, конфигурирование и запуск PHP	36
Системные требования	36
Конфигурационный файл PHP, php.ini	37
Настройка тестовой машины	37
Сетевые соединения	38
С чего начать?	38
Использование PHP 5	38
Установка PHP на Linux и Apache	39
Выбор метода установки	40
RPM-установка PHP 4	40
Компиляция PHP 5 из исходного кода	43
Настройка Apache для работы с PHP	50
Конфигурирование Apache для работы с PHP 5	51
Запуск или перезапуск Apache	53
Установка PHP 5 на Windows 2000/Internet Information Server 5	53
Загрузка дистрибутива PHP 5	55
php.ini и расширения	57
Тестирование и устранение неисправностей	61
Конфигурирование PHP	62
Файл php.ini	62
PHP-расширения	63
Кэширование	63
Резюме	63
Упражнения	64

---

<b>Глава 2. Написание простых программ</b>	65
Создание PHP-программы	66
Некоторые детали	67
Как работает PHP-код	68
Web-страница (пользовательский интерфейс)	68
Расширения файлов	69
PHP-разделители	69
Корректный PHP-код	70
Общие маркеры в коде	70
Как работают PHP-программы в Web-среде	70
Web-соединения: Internet-протоколы и HTTP	71
TCP/IP	72
HTTP-протокол	72
HTTP-запрос	73
HTTP-ответ	74
Тело ответа	75
Запуск PHP-сценариев посредством HTTP-запроса	76
Web-сервер	76
PHP-процессор	76
Использование переменных в PHP	76
Создание переменных	77
Именованые переменных	77
Типы данных	78
Область видимости переменной	80
Ключевое слово <code>global</code>	80
Статические переменные	80
Определение констант	81
Операторы и выражения	81
PHP-операторы	81
PHP-выражения	82
Типы операторов	83
Строковые операторы и функции	83
Как это работает	87
Арифметические операции в PHP	88
Как это работает	93
Массивы	94
Индексы массивов	94
Использование строк в качестве индексов массивов	95
Инициализация массивов	95
Работа с массивами	96
Сортировка массивов с помощью функций <code>sort()</code> и <code>asort()</code>	97
Резюме	98
Упражнения	98
<b>Глава 3. PHP, HTML и состояние сеанса</b>	99
Основы HTML	99
HTML DTD	101
Дескрипторы <code>form</code> и <code>input</code>	101

Доступ к PHP- и HTTP-данным	103
Предопределенные переменные	104
Переменные в HTTP-запросах и ответах	104
Как это работает	106
Суперглобальные массивы	106
Гиперссылки	109
Строки запросов	110
HTML-формы (или Web-формы)	111
Дескрипторы HTML-форм	111
Дескриптор <form>	111
Атрибуты дескриптора <form>	112
URL-кодирование	114
PHP и поля (элементы управления) HTML-форм	116
Текстовые поля (текстовые окна)	116
Как это работает	118
Почему этот пример может не работать	119
Текстовая область	119
Как это работает	120
Флажки	122
Как это работает	124
Использование нескольких флажков	124
Как это работает	126
Переключатели	127
Как это работает	129
Списки	130
Как это работает	131
Скрытые поля форм	134
Как это работает	136
Поля ввода паролей	137
Кнопки submit и reset	138
Использование в PHP-сценариях значений, возвращаемых формами	138
Как это работает	140
Возможные усовершенствования приложения	143
Понятие состояния	144
Контроль состояния	145
Скрытые поля форм	145
Строки запроса	146
Базы данных	146
Cookie-файлы	146
Как это работает	151
Собственные сеансы в PHP	153
Как это работает	155
Резюме	157
Упражнение	157
<b>Глава 4. Логические операторы, циклы и массивы</b>	<b>159</b>
Проектирование логики PHP-программы	159
Постановка задачи	160

---

Написание псевдокода	160
Булева логика	161
Булевы термы	161
Булевы значения	161
Использование булевых термов и значений	162
Условные операторы или операторы ветвления	164
Пример ветвления	164
Оператор if	165
Использование булевых операторов в структурах управляющей логики	166
Операторы > и <	166
Как это работает	168
Операторы == и ===	169
Операторы != и <>	170
Как это работает	172
Логические операторы (AND, OR, !)	172
Как это работает	174
Операторы switch	176
Как это работает	179
Циклы и массивы	181
Циклы	181
Цикл while	181
Как это работает	184
Бесконечные циклы	187
Циклы do while	188
Как это работает	189
Циклы for	191
Как это работает	193
Массивы	196
Инициализация массивов	197
Итерации в массиве	198
Как это работает	200
Усовершенствование программы	202
Итерации в неупорядоченных массивах	203
Функции current() и key()	203
Функции next() и prev()	204
Функции list() и each()	205
Итерации в ассоциативных массивах	206
Сортировка массивов	207
Функция sort()	207
Функция asort()	207
Функции rsort() и arsort()	208
Функция ksort()	209
Многомерные массивы	209
Практическое использование массивов	210
Как это работает	212
Функция array_multisort()	214

Циклы foreach	215
Как это работает	217
Резюме	217
Упражнения	218
<b>Глава 5. Надежный и понятный код</b>	<b>220</b>
Тестирование и отладка	221
Значения, нарушающие работу кода	221
Основные типы ошибок	222
Отладка PHP-сценария	223
Сообщения об ошибках PHP	223
Конфигурирование PHP для обработки ошибок	223
Типы ошибок в PHP	224
Синтаксические ошибки	224
Логические ошибки	227
Ошибки времени выполнения	227
Деление на ноль	228
Бесконечные циклы	230
Логические ошибки вывода	231
Присвоение значений вместо сравнения значений	232
Отладка и обработка ошибок в PHP5	232
Предотвращение отображения конфиденциальной информации	232
Создание собственных инструментов для отладки	232
Использование оператора echo()	233
Ошибки внутри HTML-кода	233
Проверка данных форм	234
Использование оператора exit	234
Как это работает	236
Предотвращение ввода пользователем HTML-кода: функция htmlspecialchars()	238
Проверка строк и регулярные выражения	239
Проверка строк	239
Регулярные выражения	241
Использование функции ereg()	242
Специальные символы	243
Некоторые распространенные образцы и их варианты	244
Проверка ввода данных	247
Как это работает	249
Использование регулярных выражений для проверки URL-указателей	250
Как это работает	251
Использование регулярных выражений для проверки параметров файловых путей	252
Как это работает	253
Изысканная обработка ошибок	254
Конфигурирование обработки ошибок в PHP	254
Подавление сообщений об ошибках	254
Проверка журнала ошибок	255

---

Try/Catch — нововведения в PHP5	255
Как это работает	259
Резюме	262
Упражнение	263
<b>Глава 6. Создание высококачественного кода</b>	<b>264</b>
Планирование разработки	265
Формальный процесс разработки программного обеспечения	265
Написание спецификации	265
Процесс написания кода	266
Тестирование, отладка и сопровождение	267
Оптимизация кода	267
Использование стандартов написания кода	268
Написание пользовательских функций в PHP	270
Структура функций	271
Определение и вызов функций	272
Как это работает	276
Функции переключения	277
Как значения попадают в функции	278
Передача по значению	279
Передача по ссылке	279
Установка значений по умолчанию	280
Важность порядка аргументов	280
Область видимости переменных	280
Глобальные и локальные переменные	281
Создание статических переменных в функциях	282
Как это работает	283
Вложенность функций	285
Рекурсия	285
Как это работает	287
Операторы include и require	288
Как это работает	290
О чем следует позаботиться при использовании подключаемых файлов	290
Резюме	291
Упражнение	291
<b>Глава 7. Файлы и каталоги</b>	<b>292</b>
Обработка файлов и каталогов	293
Работа с файлами	294
Открытие и закрытие файлов	294
Функция fopen()	294
Функция fclose()	297
Получение информации о файле	297
Чтение и запись файлов	298
Функция fread()	298
Функция fwrite()	298
Как это работает	300
Чтение и запись символов в файлы	302

Чтение файлов целиком	305
Произвольный доступ к данным файла	306
Как это работает	309
Получение информации о файлах	310
Временные свойства файлов	311
Как это работает	313
Принадлежность и права доступа к файлам	314
Функции <code>is_dir()</code> и <code>is_file()</code>	316
Как это работает	318
Пользовательские функции для работы с файлами	319
Разделение имени файла и пути	319
Копирование, переименование и удаление файлов	320
Работа с каталогами	322
Как это работает	323
Другие функции для обработки каталогов	324
Обход дерева каталогов	325
Сценарий для навигации по каталогам	327
Создание текстового редактора	332
Загрузка файлов на сервер	337
Как это работает	341
Резюме	342
Упражнение	342
<b>Глава 8. XML</b>	<b>343</b>
Что такое XML	344
Структура XML-документа	345
Основные части XML-документа	346
Правильно сформированные XML-документы	346
Использование XML-элементов и атрибутов	347
Корректные XML-документы: DTD-определения и XML-схемы	348
DTD-определение для XHTML	349
Ссылки на DTD-определения и XML-схемы	350
Внешние XML-схемы	351
Написание XML-документов с помощью XHTML	352
Web-службы	352
PHP и XML	353
XML-функции в PHP4	353
Как это работает	358
XML-анализаторы	359
Как это работает	362
Объектная модель документа	363
DOM-расширение	363
Использование функций DOM-расширения PHP	363
XML-функции PHP5	364
Расширение SimpleXML	364
Использование функции <code>simplexml_load_string()</code>	365
Использование функции <code>simplexml_load_file()</code>	366

---

Изменение значений с помощью simpleXML	368
Как это работает	370
Резюме	371
Упражнение	371
<b>Глава 9. Введение в базы данных и SQL</b>	<b>372</b>
Хранение данных	373
Базы данных	374
Архитектуры баз данных	374
Встроенные базы данных	374
Клиент/серверные базы данных	375
Выбор базы данных	375
Установка MySQL	376
Установка на Windows	377
Установка MySQL на Linux	377
Установка MySQL из RPM-пакетов	378
Установка MySQL из исходного кода	379
Конфигурирование MySQL	380
Реляционные базы данных	381
Нормализация	382
Первая нормальная форма (1NF)	382
Вторая нормальная форма (2NF)	384
Третья нормальная форма (3NF)	384
Другие нормальные формы	385
Обращение к базам данных с помощью SQL	385
Типы данных в SQL	385
Индексы и ключи	387
Запросы	388
Практическое применение MySQL	390
Запуск клиентской программы mysql	390
Выбор используемой базы данных	391
Просмотр таблиц в базе данных	392
Использование SQL для просмотра данных	393
Манипуляции данными	394
Использование команд GRANT и REVOKE	395
GRANT	395
REVOKE	396
Подключение к базам данных MySQL из PHP-программ	397
Связь PHP и MySQL	397
Основные функции соединения	398
Как это работает	400
Обработка серверных ошибок	401
Как это работает	404
Создание баз данных и таблиц с помощью MySQL-клиента	405
Создание демонстрационной базы данных и таблиц с помощью PHP	409
Как это работает	411
Изменение структуры таблицы	412
Вставка данных в таблицу	414



Экранирование кавычек	415
Заполнение таблиц данными	416
Резюме	417
<b>Глава 10. Получение данных от MySQL с помощью PHP</b>	<b>418</b>
Получение данных с помощью PHP	418
SQL-операторы для выборки данных	421
Серверные функции	421
Выбираемые поля	422
Ограничение количества возвращаемых результатов	423
Упорядочение результатов	425
Сравнение с образцом	426
Получение итоговых данных	427
Более сложные выборки	427
Практический пример сценария	431
Содержимое файла common_db.inc	432
Глобальные переменные	432
Функция html_header()	432
Функция html_footer()	433
Функция error_message()	433
Содержимое файла userviewer.php	433
Функция list_records()	434
Функция view_record()	437
Выбор действия	440
Использование сценария	440
Резюме	440
<b>Глава 11. Использование PHP для управления информацией в базах данных MySQL</b>	<b>442</b>
Вставка записей с помощью PHP	442
Специальные символы	443
Функция htmlspecialchars()	444
Обновление и удаление записей в таблицах	445
Работа с полями даты и времени	447
Получение информации о таблицах в базе данных	450
Как это работает	451
Как это работает	454
Параметры ENUM и стандартные значения полей	456
Как это работает	458
Создание сценария для регистрации пользователей	459
Сценарий register.php	460
Функция in_use()	460
Функция register_form()	461
Функция create_account()	462
Выбор следующего действия	464
Создание сценария для протоколирования посещений страниц	466
Файл auth_user.php	467
Файл access_logger.php	469

---

Создание сценария для управления пользователями	473
Файл userman.php	473
Функция user_message()	474
Функция list_records()	474
Функция delete_record()	474
Функция edit_record()	475
Функция edit_log_record()	476
Функция view_record()	477
Выбор действия	480
Резюме	482
Упражнения	483
<b>Глава 12. Введение в объектно-ориентированное программирование</b>	<b>485</b>
Что такое объектно-ориентированное программирование?	485
Основные понятия ОО-программирования	487
Классы	488
Объекты	489
Создание класса	489
Добавление методов	490
Добавление свойства	491
Ограничение доступа к переменным экземпляра	492
Как это работает	493
Использование функций __get и __set	494
Как это работает	495
Инициализация объектов	496
Как это работает	499
Уничтожение объекта	499
Как это работает	501
Наследование	502
Как это работает	506
Переопределение методов	507
Сохранение функциональности родительского класса	510
Как это работает	512
Интерфейсы	512
Как это работает	513
Инкапсуляция	514
Изменения объектно-ориентированных возможностей PHP5	516
Резюме	517
Упражнения	517
<b>Глава 13. Работа с UML и классами</b>	<b>518</b>
Унифицированный язык моделирования	518
Зачем использовать UML?	519
Инструменты для создания UML-диаграмм	519
Диаграммы классов	520
Создание диспетчера контактов	521

UML-диаграммы диспетчера контактов	521
Другие полезные UML-диаграммы	526
Диаграммы активности	526
Диаграммы ситуаций	527
Диаграмма последовательностей	527
Создание класса Entity	529
Сборка классов	535
Класс PropertyObject	535
Классы, представляющие типы контактной информации	537
Класс DataManager	541
Классы Entity, Individual и Organization	542
Использование системы	549
Резюме	551
<b>Глава 14. PEAR</b>	<b>552</b>
Что такое PEAR?	553
Структура PEAR	553
Базовые классы PHP-кода	554
Общественная библиотека PHP-расширений	554
Диспетчер пакетов PEAR	555
Обзор PEAR-стандартов	555
Управляющие структуры, комментарии и отступы	555
Вызовы и определения функций	556
Соглашения по именованию	557
Установка PEAR-пакетов	557
Поиск PEAR-пакетов на сайте pear.php.net	557
Изучение PEAR-классов и приложений	558
Установка и использование диспетчера пакетов PEAR	558
Установка диспетчера PEAR-пакетов для Windows	559
Использование диспетчера пакетов	559
Использование PEAR-пакетов	569
Как это работает	572
Что делать, если найдена проблема?	574
Создание приложения с использованием двух PEAR-компонентов	574
Приложение	574
Архитектура	575
Генерация XML-кода	575
Сценарий radiogeneratexml.php	577
Сценарий radiorequest.php	578
Как это работает: radiogeneratexml.php	581
Как это работает: сценарий radiorequest.php	585
Резюме	587
<b>Глава 15. PHP5 и электронная почта</b>	<b>589</b>
Основы e-mail	589
Почтовые протоколы Internet	590
Структура e-mail-сообщения	590
Отправка e-mail с помощью PHP	592

---

Использование функции mail()	592
Как это работает	593
Многоцелевые расширения почты в Internet (MIME)	595
Поля MIME-заголовков	595
Многоэлементный MIME-формат	595
Библиотеки почтовых функций в PEAR	597
Создание простого RНР-приложения для работы с e-mail	597
Резюме	604
Упражнение	605
<b>Глава 16. Генерирование графики</b>	<b>606</b>
Основы компьютерной графики	607
Теория цвета	607
Системы координат	607
Типы изображений	608
Работа с растровыми изображениями	609
Создание нового изображения	609
Распределение цветов	609
Основные функции рисования	610
Рисование отдельных пикселей	610
Рисование линий	611
Как это работает	612
Рисование прямоугольников	612
Рисование окружностей и эллипсов	613
Рисование дуг	613
Рисование многоугольников	614
Как это работает	616
Изменение растровых изображений	618
Открытие существующего изображения	618
Как это работает	620
Внедрение водяных знаков	620
Копирование copyright-знака в изображение	620
Использование прозрачности	622
Использование непрозрачности	624
Создание пиктограмм	624
Использование текста в изображениях	627
Добавление стандартного текста	628
Как это работает	628
Использование шрифтов True Type	629
Как это работает	631
Резюме	631
Упражнения	632
<b>Глава 17. Учебный пример: диспетчер     протоколирования на RНР</b>	<b>633</b>
Почему именно диспетчер протоколирования?	634
Smarty	635

Установка Smarty	635
Как это работает	637
PHPUnit	640
Работа с PHPUnit	640
Проектирование диспетчера протоколирования	641
База данных sitelogs.db	642
Использование UML для планирования диспетчера протоколирования	644
Планирование классов обработки данных	644
Планирование классов обработки исключений	647
Диаграмма последовательностей диспетчера протоколирования	648
Код приложения	648
Вспомогательные сценарии	648
settings.php	648
common.php	649
setup.php	649
initialize.php	649
Сценарии обработки данных	651
class.LogUtils.php	651
class.PersistableLog.php	655
class.UserLog.php	657
class.LogContainer.php	661
class.UserDemographic.php	662
Сценарии проверки данных и обработки ошибок	664
Сценарии уровня представления и шаблоны	669
index.php	669
report.php	670
report.tpl	671
report-html.tpl	672
Тестирование приложения	672
Работа с диспетчером протоколирования	683
userlog.php	683
Просмотр интерфейса диспетчера протоколирования	683
Резюме	687
<b>Приложение А. Ответы</b>	<b>688</b>
Глава 1	688
Упражнение	688
Решение	689
Глава 2	689
Упражнение 1	689
Решение	689
Упражнение 2	690
Решение	690
Глава 3	690
Упражнение	690
Решение	690

---

Глава 4		691
	Упражнение	691
	Решение	692
Глава 5		694
	Упражнение	694
	Решение	694
Глава 6		694
	Упражнение	694
	Решение	695
Глава 7		696
	Упражнение	696
	Решение	696
Глава 8		696
	Упражнение	696
	Решение	696
Глава 11		697
	Упражнение 1	697
	Решение	697
	Упражнение 2	697
	Решение	698
	Упражнение 3	698
	Решение	698
	Упражнение 4	699
	Решение	699
	Упражнение 5	699
	Решение	699
	Упражнение 6	699
	Решение	699
Глава 12		703
	Упражнение 1	703
	Решение	703
	Упражнение 2	703
	Решение	703
	Упражнение 3	704
	Решение	704
Глава 15		704
	Упражнение	704
	Решение	704
Глава 16		705
	Упражнение 1	705
	Решение	705
	Упражнение 2	706
	Решение	706
<b>Приложение Б. Справочник по PHP-функциям</b>		710
	Web-сервер Apache	710
	Массивы	711
	BCMath	716

BZip2	717
Календарь	717
Классы и объекты	718
Типы символов	719
Curl	720
Дата и время	720
Каталоги	722
Обработка ошибок	722
Файловая система	723
FTP-функции	726
Вызов функций	728
HTTP-функции	729
Библиотека iconv	730
Функции для работы с изображениями	730
IMAP-функции	736
Mail-функции	740
Math-функции	741
MIME-функции	743
Разные функции	743
Функции MS SQL	744
Функции MySQL	746
Сетевые функции	748
ODBC-функции	749
Буферизация вывода	752
Функции PCRE	752
Параметры PHP и информация о PHP	753
Выполнение программ	755
Регулярные выражения	756
Сеансы	757
Simple XML	758
Сокеты	758
Функции SQLite	760
Расширение Streams	762
Строки	764
URL-функции	770
Функции переменных	770
XML-функции	772
ZLib	773
<b>Приложение В. Использование SQLite</b>	<b>774</b>
Что такое SQLite?	774
Как получить SQLite?	775
Почему стоит (или не стоит) использовать SQLite?	775
Использование SQLite в PHP	776
Создание и поддержка соединений	776
Манипуляция данными	777
Сведения об извлекаемых данных	778
Разные функции	779

---

Практическое использование SQLite	780
Приложение “Персональная библиотека”	780
Создание базы данных и таблиц	780
Форма ввода данных	782
Главная страница и листинг книг	785
Редактирование записи	786
<b>Приложение Г. ODBC</b>	<b>790</b>
Общие ODBC-функции	790
ODBC-функции в PHP	791
Другие ODBC-функции	792
Использование ODBC в Windows и Linux	792
ODBC-параметры в конфигурации PHP	793
Пример использования ODBC и SQL Server в PHP-приложении для Windows	793
Создание базы данных Microsoft SQL Server	793
Создание системного DSN	795
Использование ODBC-функций PHP	799
<b>Приложение Д. CLI-интерфейс PHP</b>	<b>804</b>
Начало	804
Некоторые важные моменты	805
Обработка параметров командной строки	807
Аргументы в PHP CLI	807
Запуск shell-команды	808
Автоматизация PHP CLI	809
Интерактивность средствами PHP CLI	811
Заключение	812
<b>Приложение Е. Конфигурация PHP5</b>	<b>813</b>
Описание файла php.ini-dist	813
Настройки языка	815
Ограничения ресурсов	818
Обработка и протоколирование ошибок	819
Обработка данных	821
Пути и каталоги	823
Загрузка файлов на сервер	824
Обработчики функции fopen	824
Динамически загружаемые расширения	825
Настройки расширений	826
<b>Предметный указатель</b>	<b>837</b>