

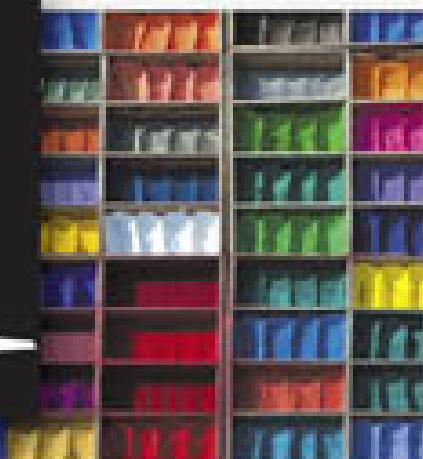
ДЕНИС КОЛИСНИЧЕНКО



# PHP и MySQL

## Разработка Web-приложений

4-е издание



ОСОБЕННОСТИ PHP 5.4

ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ НА PHP 5.4

MySQL и SQLite

ПОЛЕЗНЫЕ PHP-ФУНКЦИИ

ШАБЛОНИЗАТОР Smarty

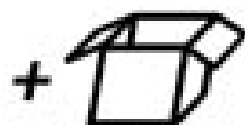
ОСНОВЫ Ajax

РАБОТА С SMTP, POP, RSS, MP3, reCAPTCHA, Disqus, Zend Guard, SPAW2

ИСХОДНЫЙ КОД СЛОЖНОГО ПРОЕКТА

SSL-СОЕДИНЕНИЯ

СБОРКА XAMPP



Материалы  
на [www.bhv.ru](http://www.bhv.ru)

УДК 681.3.06  
ББК 32.973.26-018.2  
К60

**Колисниченко Д. Н.**

К60 PHP и MySQL. Разработка Web-приложений. — 4-е изд., перераб. и доп. — СПб.: БХВ-Петербург, 2013. — 560 с.: ил. — (Профессиональное программирование)

ISBN 978-5-9775-0876-6

Даны начала программирования на PHP: установка и настройка Apache, PHP, MySQL и кроссплатформенной сборки XAMPP, выбор редактора PHP-кода, синтаксис языка и самые полезные функции PHP 5.4. Рассмотрено создание собственного движка сайта и ряда дополнительных модулей — фотогалереи, RSS-граббера, модуля для работы с MP3, сервиса reCAPTCHA, а также применение мощного шаблонизатора Smarty и создание простейшего собственного шаблонизатора. Показано, как с помощью технологии Ajax добиться обновления данных на странице без ее перезагрузки. Для закрепления практики разработки Web-приложений рассмотрен сложный проект, приведены его подробное описание, исходный код и интеграция с популярным движком WordPress. В четвертом издании добавлен материал, знакомящий с механизмом кэширования MySQL, продуктом Zend Guard и процессом создания WAP-сервера.

На сайте издательства находятся дополнительные главы, все листинги из книги, а также необходимое программное обеспечение.

*Для Web-программистов*

УДК 681.3.06  
ББК 32.973.26-018.2

#### **Группа подготовки издания:**

Главный редактор	<i>Екатерина Кондукова</i>
Зам. главного редактора	<i>Евгений Рыбаков</i>
Зав. редакцией	<i>Екатерина Капалыгина</i>
Редактор	<i>Владимир Красовский</i>
Компьютерная верстка	<i>Ольги Сергиенко</i>
Корректор	<i>Зинаида Дмитриева</i>
Дизайн серии	<i>Инны Тачиной</i>
Оформление обложки	<i>Марины Дамбиевой</i>

Подписано в печать 31.08.12.

Формат 70×100<sup>1/16</sup>. Печать офсетная. Усл. печ. л. 45,15.

Тираж 1500 экз. Заказ №

"БХВ-Петербург", 191036, Санкт-Петербург, Гончарная ул., 20.

Первая Академическая типография "Наука"  
199034, Санкт-Петербург, 9 линия, 12/28

ISBN 978-5-9775-0876-6

© Колисниченко Д. Н., 2013  
© Оформление, издательство "БХВ-Петербург", 2013

# Оглавление

- Введение ..... 14**
  - Что нового в 4-м издании..... 14
  - Немного истории ..... 15
  - PHP 5.4, или Что случилось с PHP 6..... 15
  - Новое в версии 5.4..... 16
    - Типажи..... 16
    - Разыменовывание массива..... 17
    - Встроенный Web-сервер..... 17
    - Индикатор загрузки файла..... 17
  - MySQL..... 18
  - Поддержка читателей..... 18
- ЧАСТЬ I. ТЕОРИЯ..... 19**
  - РАЗДЕЛ 1. БЫСТРЫЙ СТАРТ ..... 21**
    - Глава 1. Установка необходимого программного обеспечения..... 23**
      - 1.1. Нужно ли устанавливать программное обеспечение..... 23
      - 1.2. Выбор PHP-редактора и FTP-клиента ..... 24
      - 1.3. Установка Apache + PHP + MySQL в Windows..... 27
      - 1.4. Связываем PHP 5.4 и Apache..... 34
      - 1.5. Установка Apache + PHP + MySQL в Linux ..... 36
      - 1.6. Несколько советов..... 37
    - Глава 2. Первая PHP-программа..... 38**
      - 2.1. Ваша первая программа ..... 38
      - 2.2. Запуск PHP-программы..... 39
      - 2.3. Вывод текста без *echo* ..... 40
      - 2.4. Совместимость с PHP 4..... 41
    - Глава 3. Основы синтаксиса PHP..... 43**
      - 3.1. Переменные..... 43
        - 3.1.1. Правила объявления переменных. Имена переменных..... 43
        - 3.1.2. Типы данных переменных ..... 44
        - 3.1.3. Булевы переменные..... 45

3.1.4. Операции над переменными.....	46
3.1.5. Ссылки.....	47
3.2. Константы .....	48
3.3. Выражения и операции .....	49
3.3.1. Что такое выражение.....	49
3.3.2. Арифметические операции .....	49
3.3.3. Логические выражения .....	50
3.3.4. Приоритеты операций.....	50
3.3.5. Операторы эквивалентности == и === .....	51
3.3.6. Операции со строками.....	52
3.4. Условный оператор .....	53
3.5. Циклы .....	54
3.5.1. Цикл со счетчиком.....	54
3.5.2. Цикл <i>while</i> .....	54
3.5.3. Цикл <i>do-while</i> .....	55
3.5.4. Принудительное завершение цикла и пропуск итерации.....	55
3.6. Оператор выбора <i>switch-case</i> .....	56
<b>РАЗДЕЛ 2. ПЕРЕДАЧА ПАРАМЕТРОВ PHP-ПРОГРАММАМ.....</b>	<b>59</b>
<b>Глава 4. Методы GET и POST .....</b>	<b>61</b>
4.1. Интерфейс CGI .....	61
4.2. Метод GET .....	63
4.3. Метод POST .....	63
<b>Глава 5. Протокол HTTP и интерфейс CGI .....</b>	<b>65</b>
5.1. Специальные переменные окружения CGI.....	65
5.2. Заголовки протокола HTTP .....	66
5.3. Коды ответов протокола HTTP .....	67
<b>Глава 6. Передача параметров посредством HTML-формы .....</b>	<b>69</b>
6.1. Создание простейшей формы и ее обработка в сценарии.....	69
6.2. Создание пользовательского интерфейса с помощью формы .....	72
6.2.1. Ввод текста. Теги <i>INPUT</i> и <i>TEXTAREA</i> .....	73
6.2.2. Зависимые и независимые переключатели.....	74
6.2.3. Списки выбора .....	75
6.2.4. Форма для передачи файлов .....	77
6.2.5. Кнопки .....	77
6.3. Проверка параметров формы.....	78
6.3.1. Проверка корректности e-mail.....	79
6.3.2. Проверка правильности номера кредитной карты.....	80
6.3.3. Удаление лишних пробелов.....	81
6.4. Защита от спама с помощью CAPTCHA .....	81
6.5. Форма поиска в Яндексе .....	84
<b>Глава 7. Запоминание параметров с помощью Cookies и сессий .....</b>	<b>85</b>
7.1. Что такое Cookies и как с ними работать.....	85
7.2. Механизм сессий .....	87
7.2.1. Сессии и Cookies: преимущества и недостатки .....	87
7.2.2. Для чего нужны сессии .....	87

7.2.3. Как работает механизм сессий .....	89
7.2.4. Обход Cookies .....	90
7.3. Массивы и Cookies .....	91

## **Глава 8. Отдельное слово о директиве *register\_globals* .....**

8.1. Почему опасно использовать <i>register_globals</i> .....	92
8.2. Если <i>register_globals</i> отключена .....	96
8.3. PHP 6 и <i>register_globals</i> .....	98

## **РАЗДЕЛ 3. МАССИВЫ И СПИСКИ .....**

### **Глава 9. Основные операции над массивами и списками .....**

9.1. Массив и список. Цикл <i>foreach</i> .....	101
9.2. Функции <i>list()</i> и <i>array()</i> .....	103
9.3. Удаление массива .....	104
9.4. Слияние массивов .....	104
9.5. Функция <i>print_r()</i> .....	106

### **Глава 10. Функции сортировки массивов .....**

10.1. Функции для сортировки массивов .....	108
10.2. Функция <i>sort()</i> — сортировка списка .....	108
10.3. Функция <i>asort()</i> — сортировка массива по значениям .....	109
10.4. Функция <i>ksort()</i> — сортировка по ключам .....	110
10.5. Функции <i>array_reverse()</i> и <i>shuffle()</i> .....	111
10.6. Собственная функция сортировки .....	111
10.7. Натуральная сортировка .....	112

### **Глава 11. Особые операции над массивами .....**

11.1. Добавление и удаление элементов массива .....	114
11.2. Упаковка переменных в массив и их извлечение .....	115
11.3. Получение части массива .....	117
11.4. Функции автоматического заполнения массива .....	117
11.5. Сравнение массивов .....	118
11.6. Полезные операции над массивом .....	119
11.6.1. Вычисление суммы и произведения всех элементов массива .....	119
11.6.2. Проверка существования элемента в массиве .....	119
11.6.3. Получение случайного элемента из массива .....	120
11.6.4. Удаление дубликатов из массива .....	120
11.6.5. Получение значений и ключей массива .....	121
11.6.6. Замена местами значений и ключей .....	121
11.6.7. Подсчет значений в массиве .....	121
11.6.8. Замена в массиве .....	122
11.6.9. Поиск в массиве .....	122
11.6.10. Прогулка по массиву .....	123

## **РАЗДЕЛ 4. ФУНКЦИИ В PHP .....**

### **Глава 12. Полезные стандартные функции .....**

12.1. Генератор случайных чисел .....	127
12.2. Дата и время .....	128
12.2.1. Кратко о <i>timestamp</i> .....	128

12.2.2. Функции <i>strtotime()</i> и <i>checkdate()</i> .....	128
12.2.3. Вывод даты.....	129
12.3. Математические функции .....	131
<b>Глава 13. Функции для работы со строками .....</b>	<b>132</b>
13.1. Основные строковые функции .....	132
13.2. Специальные функции замены .....	135
13.3. Преобразование строки.....	137
13.4. Функции преобразования кодировок .....	139
13.5. Функции для работы с отдельными символами строки .....	140
13.6. Функция <i>md5()</i> и другие функции шифрования/хэширования .....	140
13.7. Функция <i>explode()</i> : выделение подстрок .....	141
13.8. Статистические функции .....	142
13.9. Функции вывода текста.....	142
13.10. Установка локали.....	144
13.11. Форматирование чисел и денежных величин.....	145
13.12. Преобразование систем счисления .....	146
<b>Глава 14. Работаем с файлами и каталогами .....</b>	<b>147</b>
14.1. Права доступа в UNIX.....	147
14.2. Чтение файла.....	149
14.2.1. Использование функций <i>fopen()</i> и <i>fread()</i> .....	150
14.2.2. Использование функции <i>file()</i> : построчное чтение файла.....	152
14.2.3. Чтение всего файла: функция <i>file_get_contents()</i> .....	153
14.3. Запись файла .....	153
14.4. Создание временных файлов .....	154
14.5. Работа с CSV-файлами .....	154
14.6. Специальные функции для работы с файлами .....	156
14.6.1. Функции для работы с именами файлов.....	157
14.6.2. Работа с правами доступа .....	157
14.6.3. Копирование, переименование и удаление файлов .....	158
14.6.4. Время доступа к файлу.....	159
14.6.5. Другие полезные функции .....	159
14.7. Совместный доступ к файлу .....	160
14.8. Функции для работы с каталогами.....	161
<b>Глава 15. Вывод графических изображений средствами PHP .....</b>	<b>163</b>
15.1. Библиотека GD.....	163
15.1.1. Получение информации об изображении .....	163
15.1.2. Конвертирование графических форматов .....	165
15.1.3. Вывод текста поверх картинки. Задание цвета .....	168
15.1.4. Прозрачность .....	171
15.1.5. Рисование графических примитивов .....	171
15.1.6. Поворот изображения.....	174
15.2. Изменение размера изображения .....	174
15.3. Создание водяных знаков .....	176
<b>Глава 16. Работа с сетевыми сокетами в PHP. Сетевые функции .....</b>	<b>178</b>
16.1. Еще раз о том, что такое сокет .....	178
16.2. Использование сокетов .....	178

16.3. Пример использования сокетов .....	179
16.4. Блокирующий и неблокирующий режимы сокета .....	182
16.5. DNS-функции .....	183

## **Глава 17. Собственные функции..... 184**

17.1. Зачем нужны собственные функции .....	184
17.2. Особенности функций в PHP .....	184
17.3. Объявление функции .....	185
17.4. Области видимости функции .....	186
17.5. Вложенность функций .....	186
17.6. Переменное число аргументов .....	188
17.7. Передача массивов в качестве параметров .....	189

## **РАЗДЕЛ 5. СЕРВЕРЫ БАЗ ДАННЫХ MYSQL 6 И SQLITE ..... 191**

### **Глава 18. Основы SQL ..... 193**

18.1. Немного истории .....	193
18.2. Преимущества SQL .....	193
18.3. Как выглядят запросы .....	194
18.4. Что такое база данных .....	195
18.5. Создание таблиц .....	196
18.6. Добавление записей в таблицу .....	199
18.7. Обновление записей .....	200
18.8. Выборка записей .....	201
18.9. Удаление записей .....	202
18.10. Встроенные функции .....	202
18.11. Группировка записей. Сложные запросы .....	204
18.12. Копирование записей из одной таблицы в другую .....	207
18.13. Кэширование запросов .....	207

### **Глава 19. Функции для работы с MySQL..... 209**

19.1. Подключение к серверу MySQL .....	209
19.2. Несколько MySQL-соединений .....	211
19.3. Передача запросов серверу .....	212
19.4. Работа с базой данных. Создание базы данных .....	215
19.5. Функция <i>mysql_real_escape_string(\$content)</i> .....	216
19.6. Получение информации о результате запроса .....	216

### **Глава 20. Альтернативная база данных SQLite..... 218**

20.1. MySQL vs SQLite: что лучше .....	218
20.2. Открытие базы данных .....	220
20.3. Передача запросов .....	220
20.4. Работа с результатом запроса .....	221
20.5. Список PHP-функций для работы с SQLite .....	222

### **Глава 21. Полезные приемы при работе с MySQL 6 ..... 224**

21.1. Выбор кодировки .....	224
21.2. Сортировка: вывод новинок. Вывод случайных записей .....	225
21.3. Постраничный вывод таблицы .....	226

## РАЗДЕЛ 6. ИНСТРУМЕНТЫ ДЛЯ СОЗДАНИЯ СЛОЖНЫХ ПРОЕКТОВ ... 231

### Глава 22. Разработка собственного шаблонизатора ..... 233

22.1. Организация файлов и каталогов проекта .....	233
22.2. Выносим параметры в отдельный файл .....	236
22.3. Подключение дополнительных файлов .....	237
22.3.1. Инструкции <i>include</i> и <i>require</i> .....	237
22.3.2. Альтернативный способ подключения сценариев .....	237
22.3.3. Инструкции <i>include_once</i> и <i>require_once</i> .....	239
22.4. Шаблоны .....	239

### Глава 23. Шаблонизатор Smarty ..... 243

23.1. Что такое Smarty .....	243
23.2. Установка Smarty .....	243
23.3. Создание <i>setup.php</i> .....	246
23.4. Разработка шаблонов Smarty .....	247
23.4.1. Комментарии в шаблонах .....	247
23.4.2. Переменные в Smarty .....	248
23.4.3. Файлы конфигурации шаблонов .....	249
23.4.4. Служебная переменная <i>{Smarty}</i> .....	250
23.4.5. Модификаторы переменных .....	251
23.4.6. Стандартные (встроенные) функции Smarty .....	254
Функции <i>{include}</i> и <i>{insert}</i> .....	254
Функция <i>{foreach}</i> .....	254
Функции <i>{if}</i> , <i>{elseif}</i> , <i>{else}</i> .....	256
Функция <i>{capture}</i> .....	257
Функция <i>{php}</i> .....	257
Функция <i>{strip}</i> .....	257
23.4.7. Пользовательские функции Smarty .....	258
Функция <i>{assign}</i> .....	258
Функция <i>{cycle}</i> .....	258
Функция <i>{fetch}</i> .....	259
Функции <i>{html_checkboxes}</i> и <i>{html_radios}</i> .....	259
Функция <i>{html_image}</i> .....	260
Функция <i>{html_select_date}</i> .....	260
Функция <i>{html_select_time}</i> .....	261
Функция <i>{html_table}</i> .....	262
23.5. Smarty для программиста .....	263
23.5.1. Специальные переменные .....	263
23.5.2. Полезные методы класса <i>Smarty</i> .....	265

### Глава 24. Объектно-ориентированное программирование ..... 266

24.1. Основы ООП .....	266
24.2. Классы и объекты .....	267
24.3. Конструкторы и деструкторы класса .....	269
24.4. Наследование классов. Полиморфизм .....	270
24.5. Новые возможности PHP 5 и 5.4 .....	271
24.5.1. Область видимости членов класса .....	271
24.5.2. Абстрактные классы и методы .....	272



24.5.3. Служебное слово <i>final</i> .....	272
24.5.4. Клонирование объектов .....	273
24.5.5. Обработка исключительных ситуаций.....	274
24.5.6. Константы-члены класса.....	275
24.5.7. Статические члены класса .....	275
24.5.8. Оператор <i>instanceof</i> .....	276
24.5.9. Итераторы .....	276
24.5.10. Пространства имен .....	276
24.5.11. Типажи (traits).....	277
25.5.12. Вызов метода или свойства класса выражением .....	278
<b>Глава 25. Механизм сессий.....</b>	<b>279</b>
25.1. Для чего нужны сессии .....	279
25.2. Как работает механизм сессий .....	280
25.3. Обход Cookies .....	282
25.4. Сценарий аутентификации.....	282
<b>Глава 26. Введение в PEAR.....</b>	<b>287</b>
26.1. Серьезные проекты и PEAR.....	287
26.2. Пример использования класса <i>DB</i> .....	288
<b>Глава 27. Контроль версий.....</b>	<b>292</b>
27.1. Выбор системы контроля версий .....	292
27.2. Практическое использование TortoiseHG (Mercurial) .....	293
27.3. Просмотр внесенных изменений .....	294
<b>Глава 28. Тестирование PHP-сценариев.....</b>	<b>298</b>
28.1. Программа работает, но не так, как нам нужно .....	298
28.2. "Самодельные" точки останова .....	299
28.3. Система автоматического тестирования.....	300
28.4. Директива <i>error_reporting</i> .....	304
<b>ЧАСТЬ II. ПРАКТИКА.....</b>	<b>305</b>
<b>РАЗДЕЛ 7. РАЗРАБОТКА ОСНОВНЫХ ЭЛЕМЕНТОВ САЙТА .....</b>	<b>307</b>
<b>Глава 29. Загрузка файлов на сервер .....</b>	<b>309</b>
29.1. Что нужно знать о загрузке файлов на сервер.....	309
29.2. Реализация загрузки файла .....	312
29.3. Загрузка нескольких файлов.....	314
29.4. Проблемы при загрузке файлов.....	316
<b>Глава 30. Использование FTP-функций .....</b>	<b>317</b>
30.1. Функции для работы с FTP .....	317
30.2. Примеры использования FTP-функций .....	320
<b>Глава 31. Отправка и прием почты.....</b>	<b>323</b>
31.1. Отправка почты средствами PHP — функция <i>mail()</i> .....	323
31.2. Отправка писем с вложениями — класс <i>HtmlMimeMail</i> .....	324
31.2.1. Отправка сообщения .....	324

31.2.2. Проблемы при отправке сообщения .....	328
31.2.3. MIME-типы .....	329
31.3. Класс <i>PHPMailer</i> . Разработка сценария автоматической рассылки прайс-листа .....	330
31.4. Получение писем по протоколу POP3 .....	334
<b>Глава 32. Работа с RSS: получаем новости на сайт .....</b>	<b>338</b>
32.1. Краткие сведения о RSS .....	338
32.2. Формат RSS-файла .....	338
32.3. Написание сценария импорта новостей.....	341
32.4. Подключение файла <code>import.php</code> к сайту.....	343
32.5. Создание граббера новостей.....	344
<b>Глава 33. Поиск с использованием регулярных выражений .....</b>	<b>348</b>
33.1. Нужно что-то найти.....	348
33.2. Язык регулярных выражений RegEx.....	349
33.3. Управляющие конструкции .....	351
33.3.1. Квантификаторы .....	351
33.3.2. Альтернативный оператор   .....	351
33.3.3. Скобки .....	351
33.4. Псевдосимволы.....	352
33.5. Практическое использование RegEx-функций.....	352
33.6. Фильтры, или Конец эры регулярных выражений.....	356
<b>Глава 34. Работаем с MP3 .....</b>	<b>358</b>
34.1. Формат MP3 .....	358
34.2. Библиотека PEAR .....	359
34.3. Вывод ID3-тегов .....	360
34.4. Редактирование ID3-тегов .....	361
34.5. Удаление тега.....	362
<b>Глава 35. Технология Ajax .....</b>	<b>363</b>
35.1. Что такое Ajax.....	363
35.2. Ваше первое Ajax-приложение.....	364
<b>РАЗДЕЛ 8. РАЗРАБОТКА ТИПИЧНОГО САЙТА.....</b>	<b>369</b>
<b>Глава 36. Создание простейшего движка сайта .....</b>	<b>371</b>
36.1. Планирование движка .....	371
36.1.1. Зачем нужно разрабатывать собственный движок .....	371
36.1.2. Необходимые нам функции движка.....	372
36.1.3. "Принципиальная схема" движка.....	374
36.2. Основные функции движка.....	375
36.2.1. Разработка TPL-шаблона .....	375
36.2.2. Файл настроек.....	376
36.2.3. Основной файл CMS — <code>index.php</code> .....	377
36.2.4. Проектирование базы данных .....	379
Таблица <code>static</code> .....	379
Таблица <code>cats</code> .....	380
Таблица <code>pages</code> .....	381

36.2.5. Иерархическая структура сайта.....	382
Алгоритм работы меню .....	382
Сценарий menu.php .....	384
Вывод содержимого раздела и страницы.....	389
36.3. Дополнительные функции движка .....	393
36.3.1. Вывод информации из таблицы static .....	393
36.3.2. Функция вывода содержимого HTML-файла.....	394
36.3.3. Версия для печати.....	395
36.4. Где взять листинги этой главы .....	396
<b>Глава 37. Создание фотогалереи .....</b>	<b>397</b>
37.1. Постановка задачи .....	397
37.2. Загрузка изображений на сервер .....	397
37.3. Вывод галереи.....	401
<b>Глава 38. Гостевая книга.....</b>	<b>404</b>
38.1. Пережиток прошлого?.....	404
38.2. Разработка базы данных и структура гостевой книги .....	405
38.3. Вывод гостевой книги .....	405
38.4. Добавление записей в гостевую книгу.....	408
38.5. Сервисный сценарий gb_service.php .....	411
<b>Глава 39. Интеграция галереи LiveJournal и вашего сайта .....</b>	<b>413</b>
39.1. Что такое Живой журнал .....	413
39.2. Интеграция фотогалереи LiveJournal и сайта.....	414
39.3. Настройка внешнего вида галереи .....	418
<b>Глава 40. Создание счетчика сайта .....</b>	<b>420</b>
40.1. Постановка задачи .....	420
40.2. Файл конфигурации.....	420
40.3. Разработка таблиц counter и ipaddr .....	421
40.4. Сценарий counter.php.....	422
40.5. Сценарий reset_counter.php .....	426
<b>Глава 41. Статистика сайта .....</b>	<b>428</b>
41.1. Методы сбора статистики .....	428
41.2. Программы-анализаторы журналов Web-сервера .....	429
41.3. Системы статистики .....	431
<b>Глава 42. Голосования и комментарии для сайта.....</b>	<b>434</b>
42.1. Разработка собственной системы голосования.....	434
42.1.1. Общая структура системы .....	434
42.1.2. Разработка сценария poll_form.php .....	435
42.1.3. Сценарий poll_process.php .....	437
42.1.4. Сценарий poll_results.php.....	438
42.2. Комментарии DISQUS .....	441
42.2.1. Описание платформы .....	441
42.2.2. Установка комментариев на свой сайт .....	442

<b>РАЗДЕЛ 9. СЛОЖНЫЙ ПРОЕКТ .....</b>	<b>449</b>
<b>Глава 43. Постановка задачи .....</b>	<b>451</b>
43.1. Концепция проекта .....	451
43.2. Алгоритм работы .....	452
43.3. Структура базы данных .....	453
<b>Глава 44. Разработка основной части сайта .....</b>	<b>457</b>
44.1. Структура проекта .....	457
44.2. Основной сценарий index.php .....	458
44.2.1. Подключение механизмов WordPress .....	458
44.2.2. Аутентификация пользователя .....	459
44.2.3. Форма добавления цели .....	460
44.2.4. Добавление цели .....	461
44.2.5. Просмотр всех целей .....	465
44.3. Сценарий v.php — просмотр цели .....	469
44.4. Сценарии отправки уведомлений. Сценарии yes.php и no.php .....	470
<b>Глава 45. Панель управления WordPress.....</b>	<b>478</b>
45.1. О чем эта глава?.....	478
45.2. Вход в панель управления.....	478
45.3. Просмотр целей .....	478
45.4. Не забывайте о UTF-8 .....	480
<b>РАЗДЕЛ 10. ОБЕСПЕЧЕНИЕ БЕЗОПАСНОСТИ САЙТА.....</b>	<b>481</b>
<b>Глава 46. SSL-соединения.....</b>	<b>483</b>
46.1. Защищаем передаваемые данные.....	483
46.2. Настройка SSL в DirectAdmin.....	484
46.3. SSL-переменные .....	486
<b>Глава 47. Защита PHP с помощью конфигурационного файла.....</b>	<b>488</b>
47.1. Конфигурационный файл php.ini.....	488
47.2. Отключение потенциально опасных функций .....	490
47.3. Рекомендованные значения некоторых конфигурационных директив.....	490
<b>Глава 48. Защита сайта от атак .....</b>	<b>491</b>
48.1. Сайт в опасности.....	491
48.2. Два самых распространенных метода взлома .....	492
48.3. Межсайтовый скриптинг.....	492
48.4. SQL-инъекции .....	494
48.5. Флуд.....	498
48.6. Защита форума PHPBB2 от спаммеров .....	499
<b>РАЗДЕЛ 11. ПОЛЕЗНЫЕ СВЕДЕНИЯ.....</b>	<b>501</b>
<b>Глава 49. Визуальный редактор SPAW2 для вашего сайта .....</b>	<b>503</b>
49.1. Знакомство с визуальным редактором и его базовая настройка .....	503
49.2. Адаптация редактора к вашему сайту .....	507
49.2.1. Загрузка в редактор текста статьи.....	507

49.2.2. Вывод редактора в TPL-шаблоне .....	507
49.2.3. Запрет изменения размера редактора .....	508
49.2.4. Изменение размеров редактора "на лету" (в процессе выполнения) сценария .....	509
49.2.5. Панели инструментов редактора .....	509
49.2.6. Ошибка Error 105 (net::ERR_NAME_NOT_RESOLVED) .....	512
<b>Глава 50. WAP-сервер средствами PHP .....</b>	<b>513</b>
50.1. Есть ли необходимость в WAP? .....	513
50.2. Настройка Web-сервера Apache .....	514
50.3. PHP-сценарий .....	515
<b>Глава 51. Пишем коммерческий сценарий .....</b>	<b>516</b>
51.1. Zend Guard: за и против .....	516
51.2. Установка Zend Guard .....	517
51.3. Быстрая защита сценария .....	518
51.4. Ошибка Unable to locate XML file .....	522
51.5. Лицензии .....	522
51.6. Zend Guard — панацея? .....	526
<b>ПРИЛОЖЕНИЯ .....</b>	<b>527</b>
<b>Приложение 1. Программа phpMyAdmin .....</b>	<b>529</b>
Действия над таблицами .....	530
Создание новой таблицы. Изменение структуры таблицы .....	532
Вставка записей .....	533
Обзор таблицы .....	533
Выполнение произвольного SQL-кода .....	533
Резервное копирование БД .....	535
<b>Приложение 2. Flash и PHP .....</b>	<b>536</b>
<b>Приложение 3. Оптимизация PHP-кода .....</b>	<b>540</b>
ПЗ.1. Правильное использование <i>echo</i> .....	540
ПЗ.2. Длина переменных .....	541
ПЗ.3. Правильное обращение к элементам массива .....	541
ПЗ.4. Циклы .....	542
ПЗ.5. Чтение файлов .....	543
ПЗ.6. Вывод текста: <i>echo</i> против <i>printf</i> .....	545
ПЗ.7. "Лишние" вызовы функций .....	547
ПЗ.8. Контрольные замеры .....	548
ПЗ.8.1. Чтение файла .....	548
ПЗ.8.2. Вывод содержимого файла .....	549
ПЗ.9. Выводы .....	551
<b>Приложение 4. Описание электронного архива .....</b>	<b>552</b>
<b>Предметный указатель .....</b>	<b>554</b>

# ГЛАВА 1



## Установка необходимого программного обеспечения

### 1.1. Нужно ли устанавливать программное обеспечение

При написании этой книги я вообще сомневался в целесообразности данной главы. Почему? Если вы купили эту книгу, значит, со временем собираетесь профессионально заниматься PHP (разрабатывать сайты, сценарии) или же планируете создать собственный сайт на PHP. Следовательно, вам понадобится хостинг с поддержкой PHP. Учитывая то, что PHP-хостинг сейчас стоит очень дешево, как и безлимитный Интернет, то все, что нужно для PHP-разработки, — это какой-нибудь текстовый редактор с поддержкой PHP-синтаксиса.

Все, что вам нужно, — это написать сценарий. В свою очередь, сценарий — это обычный текстовый файл. Как только сценарий будет готов, вам его нужно загрузить по FTP на ваш хостинг и запустить через браузер. При этом совсем не имеет значения, какую операционную систему вы используете: Windows, Linux или даже MacOS.

Ранее имело смысл устанавливать на свой компьютер Apache + PHP + сервер баз данных MySQL. Сейчас можно производить разработку и отладку сценариев непосредственно на сервере. Так даже правильнее, поскольку конфигурации Apache и PHP на локальном и удаленном сервере в 99% случаев будут различны, поэтому при переносе сценариев могут возникнуть нюансы. Какие именно? Начиная с самых безобидных, вроде отличающихся параметров базы данных MySQL, до разницы в версии PHP. С параметрами базы данных все просто: при переносе сценария с одного сервера на другой не забывайте проверять параметры доступа, иначе ваши сценарии откажутся работать. Чуть позже мы продемонстрируем пример, позволяющий легко избавиться от этой проблемы. А вот с версией PHP и установленными PHP-расширениями вообще могут возникнуть глобальные проблемы. Помню, я столкнулся со следующей проблемой. У меня был отлично работающий проект, отправляющий прайс-листы клиентам одной из поддерживаемых мною организаций. Сценарий был предельно простым: пользователь загружал новый прайс-лист, а сценарий отправлял его всем клиентам, адреса которых были в базе данных. Для отправки сообщений использовался класс `PHPMailer`. После переноса на другой сер-

вер сценарий перестал запускаться, причем даже не выдавал сообщение об ошибке. Первым делом я запустил функцию `phpinfo()`:

```
<?php  
phpinfo();  
?>
```

Результат меня немного поразил. На сервере была установлена четвертая (!) версия PHP. Учитывая, что уже есть версия 5.4, PHP 4 выглядит антиквариатом. Решить проблему удалось путем загрузки класса `PHPMailer` для PHP 4. Хорошо, что такой до сих пор сохранился на серверах разработчиков!

Иная распространенная проблема — отсутствие на другом сервере необходимых вам расширений. Например, вы создали галерею картинок, использующую функции из библиотеки GD. Но ваша галерея не будет работать, если на сервере не установлено расширение GD (PHP-библиотека GD). Поэтому, прежде чем размещать сценарий на сервере, вызовите функцию `phpinfo()` и убедитесь, что на сервере установлена нужная версия PHP и нужные расширения.

Как узнать, какая версия PHP вам нужна? Раньше на серверах могла быть установлена или одна из ранних версий PHP 5, или проверенная временем четвертая версия. Но сейчас PHP 4 на сервере не встретишь, все перешли на PHP 5.

Однако не забывайте, что в каждой версии (PHP 5.1, 5.2, 5.3 и 5.4) есть свои изменения, например новые параметры у функций и новые функции. Когда вы учитесь, по большому счету все равно, какая будет версия. Главное, чтобы вы знали, какая именно, дабы не возникало вопросов, почему та или иная функция не работает как следует.

Хотя, на самом деле, все не так страшно. Большинство функций будут работать, так что вы не испытаете никаких проблем. Исключение может составить разве что версия 5.4. По сути, это версия 6.0 (которая так и не вышла), но без Юникода.

Пока хостинг-провайдеры не спешат устанавливать версию 5.4, хотя она уже стабильна. Так что с ней делать? Поскольку есть релиз версии 5.4, то изучать ее стоит. Но пока будьте готовы, что далеко не все хостеры используют 5.4, а со временем переход обязательно состоится. В данный момент вы можете загрузить PHP 5.4 на свой компьютер и изучить новые возможности этой версии.

Итак, если вы не собираетесь устанавливать Web-сервер Apache, интерпретатор PHP и сервер баз данных на свой компьютер, вам понадобятся две программы — текстовый редактор и FTP-клиент.

Нужно отметить, что в версии 5.4 есть встроенный Web-сервер, но я рекомендую все же использовать Apache, поскольку в большинстве случаев на сервере хостера будет установлен именно этот сервер и условия будут "ближе к боевым".

## 1.2. Выбор PHP-редактора и FTP-клиента

Из всех PHP-редакторов, которыми я пользовался, мне больше всего понравились два редактора — PHP Expert Editor и Zend Studio (вы без проблем найдете сайты

разработчиков этих редакторов). Оба редактора коммерческие, но они стоят своих денег. PHP Expert Editor (рис. 1.1) обеспечивает не только подсветку синтаксиса, но и умеет проверять синтаксис сценария. Правда, для проверки синтаксиса вам нужно установить PHP (об этом позже) и указать его в настройках программы.

### ПРИМЕЧАНИЕ

Условно-бесплатную версию программы PHP Expert Editor вы найдете в архиве с дополнительными материалами, который можно скачать по адресу: [http://dkws.org.ua/mybooks/php\\_adds.zip](http://dkws.org.ua/mybooks/php_adds.zip) (в каталоге software архива). Также последнюю версию программы всегда можно скачать с сайта разработчиков <http://phpexpeditor.com/>.

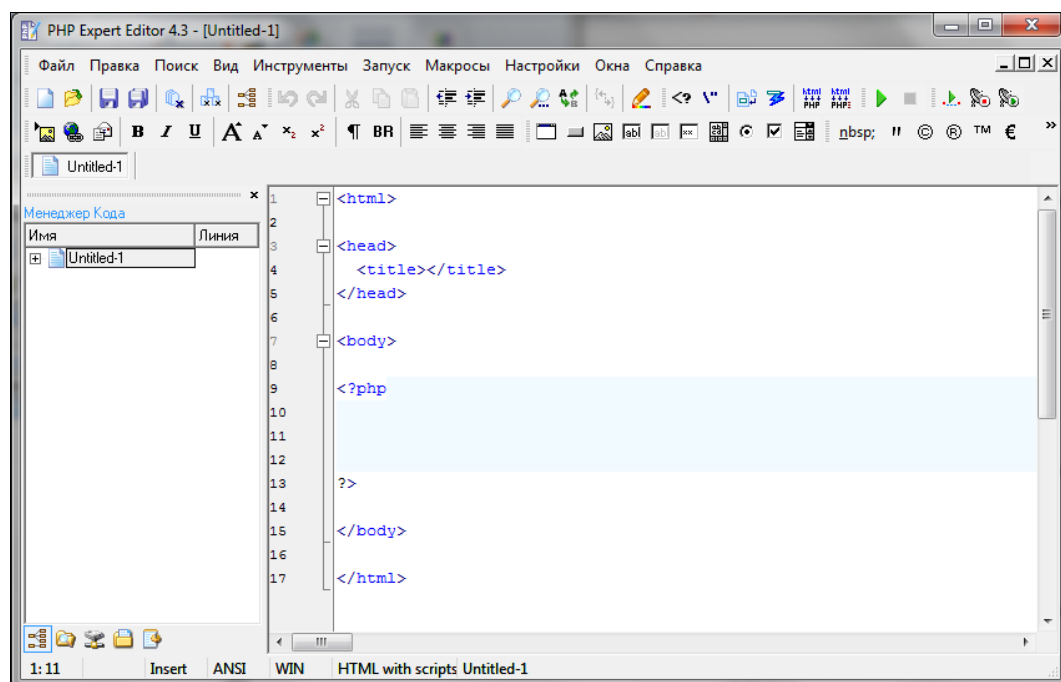


Рис. 1.1. Программа PHP Expert Editor

Для этого выполните команду **Запуск | Настройки**, в появившемся окне перейдите на вкладку **Интерпретаторы скриптов** (рис. 1.2) и укажите путь к интерпретатору PHP. Для проверки синтаксиса нажмите клавишу <F7>.

Возможности Zend Studio куда шире (рис. 1.3). Чего стоит только автодополнение кода, когда программа "читает" ваши мысли и дописывает за вас PHP-код. Если вы работали с Delphi и другими продуктами Borland, то поймете, о чем я говорю. Поверьте, очень удобно.

В качестве FTP-клиента могу порекомендовать бесплатную программу FileZilla (рис. 1.4), которая доступна как для Windows, так и для Linux. Версия 3.5.3 этого FTP-клиента находится в архиве с дополнительными материалами в каталоге software.



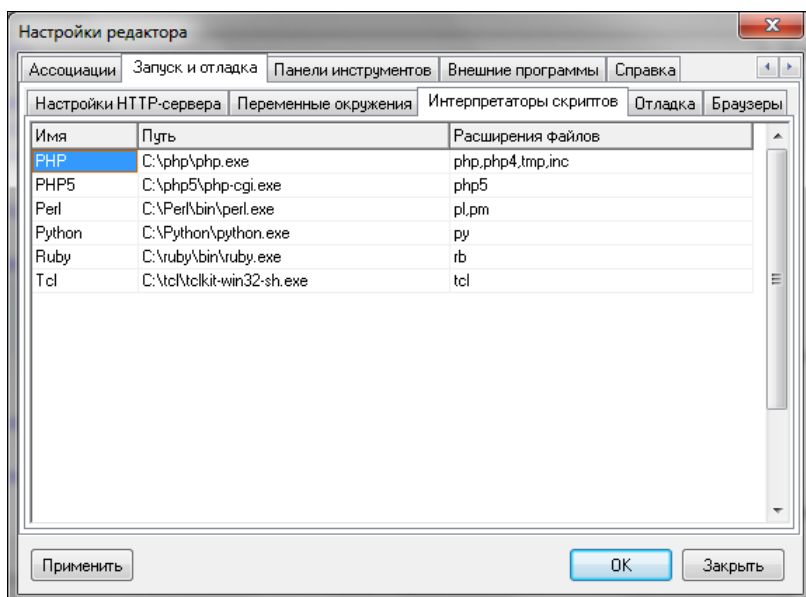


Рис. 1.2. Путь к интерпретатору PHP

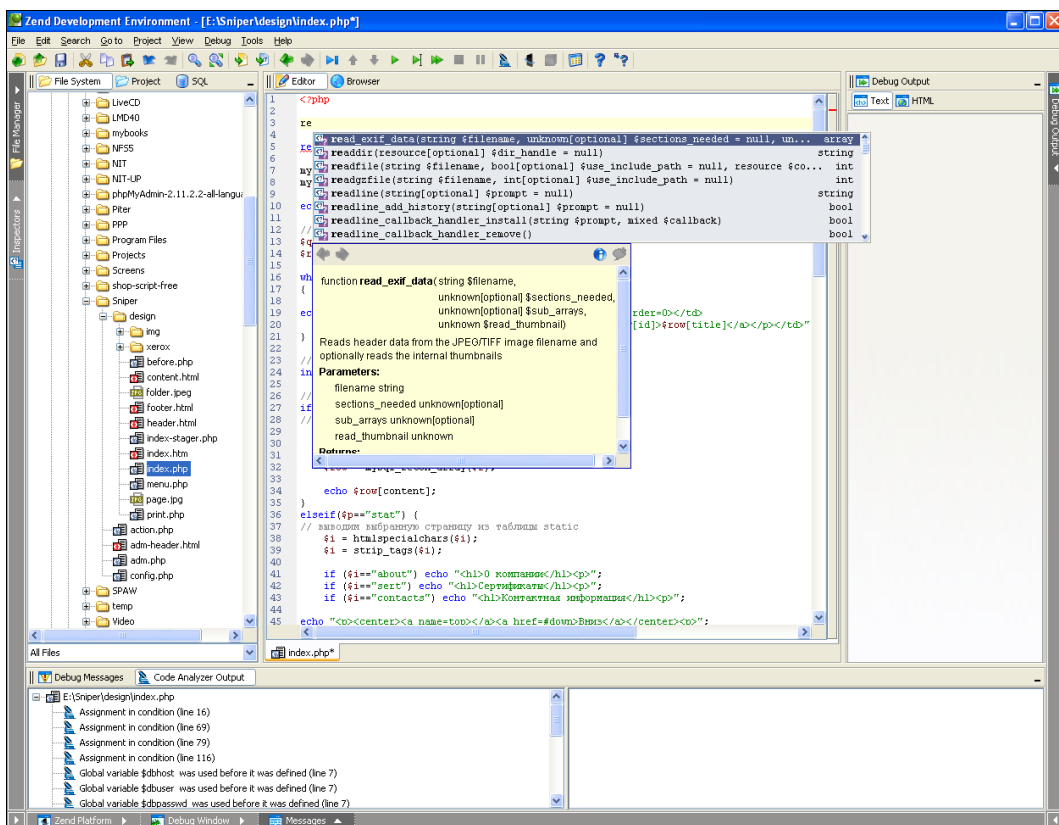


Рис. 1.3. Zend выводит не только имена функций, но и краткую справку по каждой функции

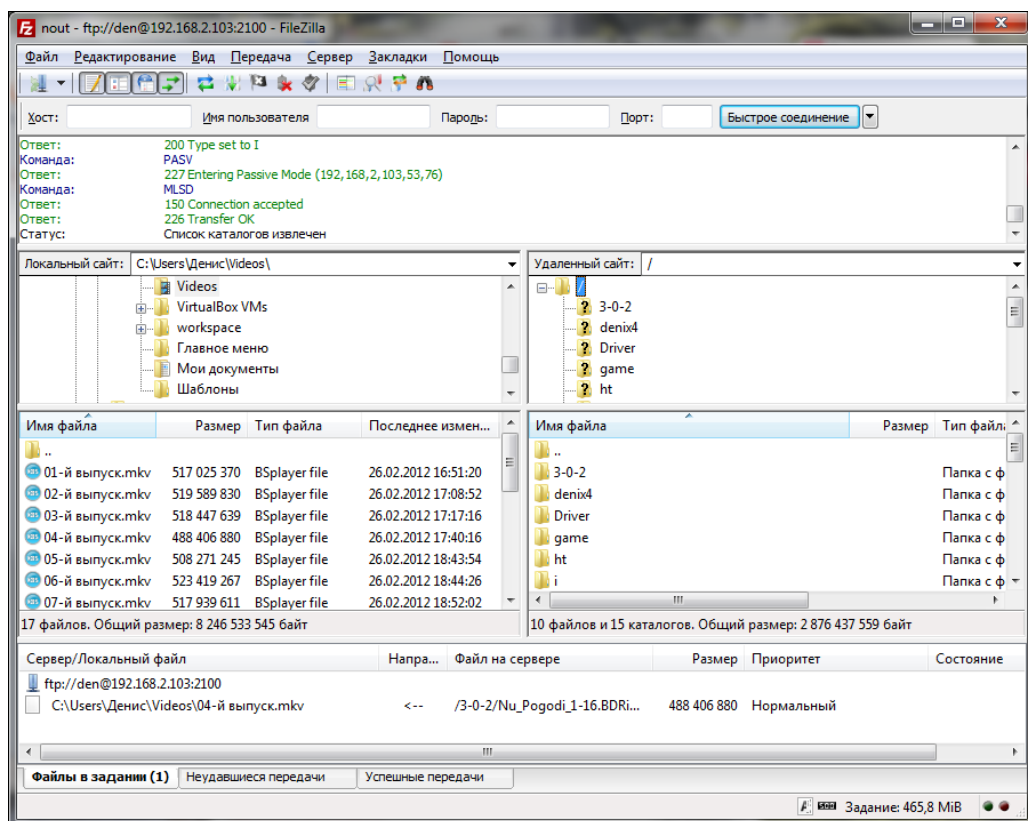


Рис. 1.4. Программа FileZilla

## 1.3. Установка Apache + PHP + MySQL в Windows

Если вы все-таки решили настроить собственный сервер, то приступим к установке. Первым делом вам нужно скачать с сайта [www.apache.org](http://www.apache.org) Web-сервер Apache. Вы скачаете файл `httpd_2.2.X-win32-x86-no_ssl.msi`. Запустите его. Процесс установки обычно проходит без осложнений, но мы все же рассмотрим его.

Версия 2.4 доступна пока только для UNIX, поэтому Windows-пользователям придется довольствоваться только версией 2.2.

### ПРИМЕЧАНИЕ

В архиве с дополнительными материалами, в каталоге `software` вы найдете обычную (no-SSL) и SSL-версию Apache для Windows. Linux-версии на диске нет, поскольку Apache входит в состав любого дистрибутива Linux, вам нужно только установить его. Несмотря на наличие в каталоге `software` установочных файлов Apache, не забудьте посетить страницу <http://archive.apache.org/dist/httpd/binaries/win32/> — вполне возможно, что к моменту покупки книги вышла новая версия Apache. На диске находится версия 2.2.22, а также модуль `mod_ftp` для Windows.

### УСТАНОВКА В WINDOWS VISTA И WINDOWS 7

Установку Apache, PHP и MySQL в Windows Vista и в Windows 7 желательно производить от имени администратора. Для этого щелкните на установочном файле правой кнопкой мыши и выберите команду **Запуск от имени администратора**.

Важно правильно указать информацию о сервере: доменное имя сети, доменное имя компьютера (если вы не планируете использовать сервер во всей локальной сети, лучше установить localhost) и адрес администратора (рис. 1.5). Также вам нужно выбрать способ запуска сервера:

- ☐ **for All Users** — сервер будет установлен как Windows-сервис и будет доступен всем пользователям;
- ☐ **only for the Current User** — сервер нужно будет запускать вручную и он будет использовать порт 8080, т. е. при обращении к серверу нужно будет указывать порт в URL, например **http://localhost:8080/**.

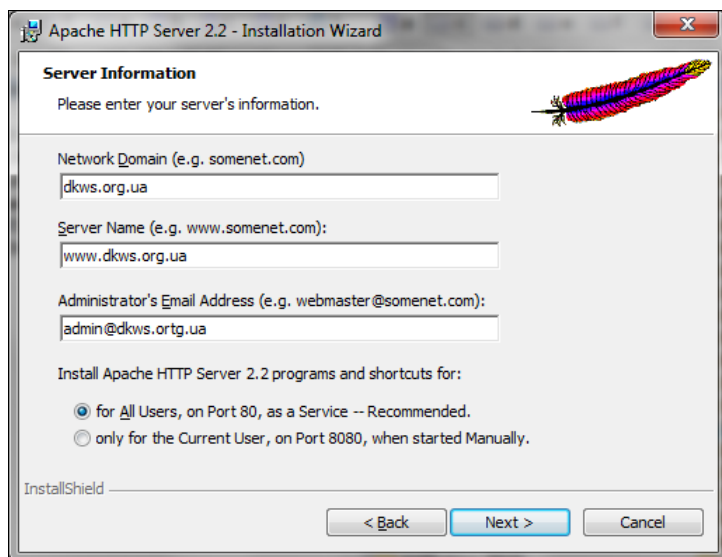


Рис. 1.5. Параметры сервера

Затем выберите Custom-тип установки и измените каталог установки Apache (рис. 1.6). По умолчанию используется каталог C:\Program Files\Apache Software Foundation\Apache2.2\, но намного проще установить сервер в каталог C:\WWW или D:\WWW. Нажав кнопку **Space**, вы можете посмотреть, сколько свободного места осталось на ваших дисках.

После установки Apache запустите браузер и введите следующий URL:

**http://localhost**

Если вы увидите "It works!" (рис. 1.7), значит, установка Apache прошла успешно.

По умолчанию сервер Apache не поддерживает PHP. Поэтому скачайте с сайта **www.php.net** инсталлятор PHP (он тоже выполнен в виде MSI-пакета) и установите

его. Обратите внимание на версию PHP! Если вы хотите использовать последние возможности PHP, нужно скачать PHP 5.3.

### PHP 5.3 и 5.4

В каталоге software архива с дополнительными материалами ([http://dkws.org.ua/mybooks/php\\_adds.zip](http://dkws.org.ua/mybooks/php_adds.zip)) вы найдете установочный файл PHP 5.3 и PHP 5.4. Версия 5.3 представлена в виде инсталлятора, а PHP 5.4 — в виде zip-архива, т. к. пока инсталлятор для нее не был создан. Поэтому связкой PHP + Apache придется заниматься самостоятельно.

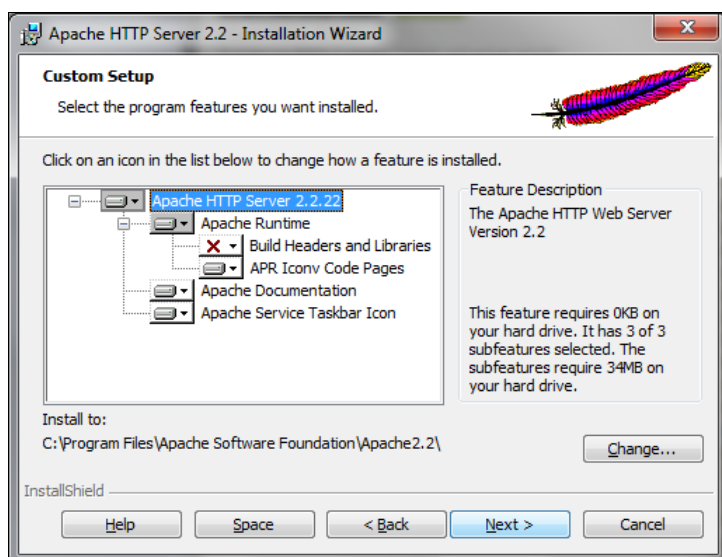


Рис. 1.6. Каталог для установки Apache

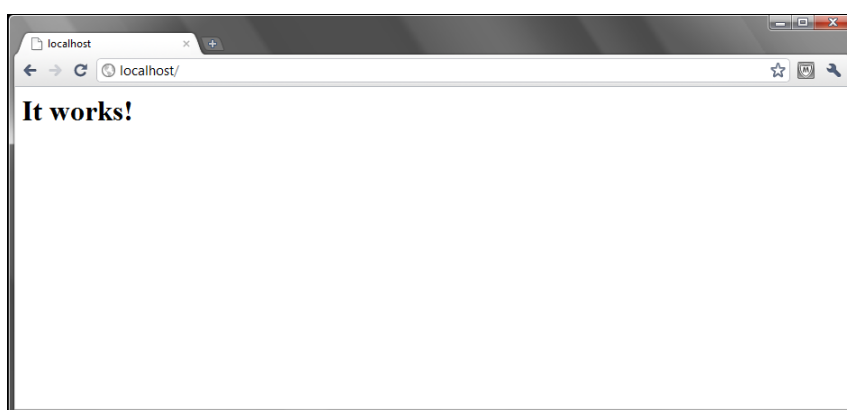


Рис. 1.7. Web-сервер работает!

При установке PHP обязательно выберите версию вашего Web-сервера. Мы установили версию 2.2, поэтому ее нужно выбрать при установке PHP (рис. 1.8). Далее нужно указать путь к конфигурационному каталогу Apache, чтобы инсталлятор PHP

внес необходимые изменения в файлы конфигурации Apache. Если вы установили Apache в D:\WWW, то вам нужно указать каталог D:\WWW\conf.

Теперь нужно перезагрузить Apache. Для этого нажмите **Пуск** и введите команду `services.msc`. Отыщите в списке служб Apache 2.2 и нажмите кнопку **Перезапуск службы** (рис. 1.9). В случае возникновения ошибки перезапустите компьютер.

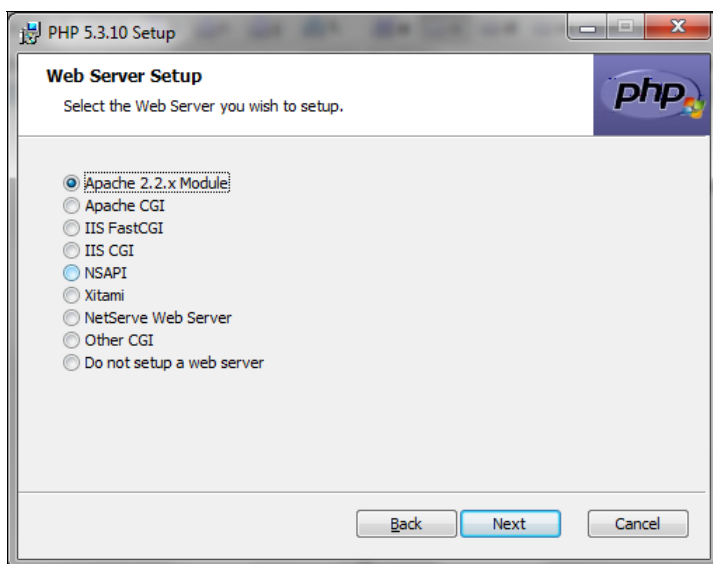


Рис. 1.8. Выбор версии Web-сервера

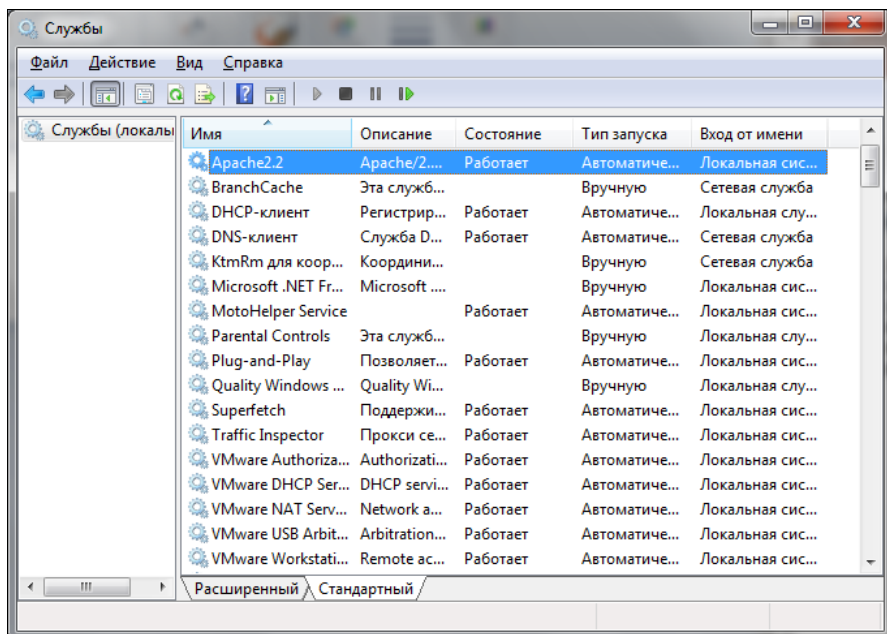


Рис. 1.9. Управление службами Windows

Теперь вам осталось скачать и установить сервер баз данных MySQL (<http://mysql.com/>). Его установка проходит вообще без каких-либо нюансов, поэтому мы не будем ее подробно рассматривать. В каталоге software вы найдете версию 5.5 (на данный момент это рекомендуемая, не экспериментальная версия MySQL). В главе 2 мы напишем сценарий test.php, тестирующий созданную вами конфигурацию. Данный сценарий нужно поместить в подкаталог D:\WWW\htdocs.

Если вам лень устанавливать все отдельно, вы можете установить пакет XAMPP, сочетающий Apache, PHP и MySQL "в одном флаконе". Скачать последнюю версию XAMPP можно по адресу <http://www.apachefriends.org/en/xampp.html>. В каталоге software вы найдете последнюю (на момент написания этих строк) версию XAMPP — 1.7.7.

Основное преимущество XAMPP — простота установки и управления всеми сервисами, входящими в его состав (Web-сервер, FTP-сервер и MySQL). Но у XAMPP есть и недостаток — версии программного обеспечения, входящие в его состав, отстают от текущих. Впрочем, последняя версия XAMPP меня порадовала. В ней Apache версии 2.2.21 (последняя из ветки 2.2 — 2.2.22) и PHP версии 5.3.8.

Установка XAMPP проблем не вызывает. Программа нормально работает в актуальных версиях Windows — XP, Vista и Windows 7. В Windows Vista и Windows 7 программу установки XAMPP нужно запускать от имени администратора.

Желательно устанавливать XAMPP в корневой каталог диска C:\, а не в каталог C:\Program Files, чтобы не было различных проблем с правами доступа — так рекомендуют разработчики XAMPP. Да и вам добираться до ваших файлов будет проще.

При установке Apache и MySQL вас спросят, как вы хотите запускать эти программы — в качестве сервисов (служб) Windows или вручную. Если вы собираетесь программировать на PHP каждый день, тогда удобнее запускать Apache и MySQL как сервисы. Но помните, что даже когда они вам не нужны, они все равно занимают память и потребляют ресурсы процессора. Поэтому, если вы собираетесь программировать на PHP время от времени, лучше запускать эти программы вручную. Для этого выполните команду меню **Пуск | Все программы | XAMPP for Windows | XAMPP Control Panel**. Вы увидите панель управления XAMPP. Панель управления позволяет запустить, остановить и настроить подконтрольные сервисы (рис. 1.10):

- ☐ **Apache** — это и есть Web-сервер;
- ☐ **MySql** — сервер баз данных;
- ☐ **FileZilla** — простейший FTP-сервер для Windows;
- ☐ **Mercury** — простейший почтовый сервер (точнее, агент доставки почты — MTA) для Windows;
- ☐ **Tomcat** — программа-контейнер сервлетов, написана на языке Java и реализует спецификацию сервлетов и спецификацию JavaServer Pages (JSP).

Все сервисы запускать не нужно — вам хватит первых двух. Третий сервис нужно запустить, если вы хотите написать PHP-сценарий, взаимодействующий с FTP-

сервером — вам не нужно искать какой-то FTP-сервер, все будет у вас под рукой. Система Mercury подойдет для проектов, подразумевающих отправку почты — без запуска Mercury ваши проекты не смогут отправлять почту стандартной функцией `mail()` или же вам придется использовать дополнительные классы вроде `PHPMailer` для отправки почты через сторонние SMTP-серверы.

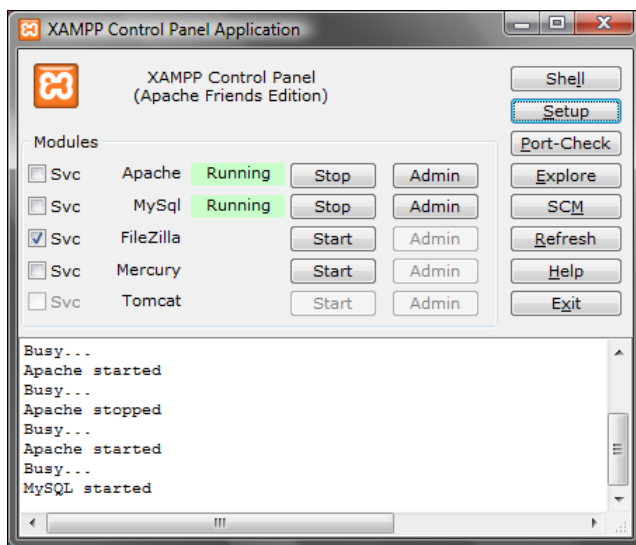


Рис. 1.10. Панель управления XAMPP

Для запуска сервиса нажмите кнопку **Start** напротив его имени. В случае успешного запуска кнопка **Start** превратится в кнопку **Stop** (используется для останова сервиса), а между названием сервиса и кнопкой **Stop** появится надпись **Running**.

Кнопка **Admin** (становится активной после запуска сервиса) позволяет настраивать выбранный сервис. На рис. 1.11 изображена страница настройки Apache, а нажатие кнопки **Admin** для **MySQL** приводит к вызову программы phpMyAdmin (рис. 1.12). Программа phpMyAdmin предоставляет удобный Web-интерфейс для управления вашими базами данных. Подробно программа описана в *приложении 1*.

Нажатие кнопки **Setup** (см. рис. 1.10) вызовет сервисное меню XAMPP (рис. 1.13):

- ☐ 1 — запустить панель управления XAMPP (команда для вас неактуальна, поскольку панель управления уже запущена);
- ☐ 2 — изменить каталог установки XAMPP (а вот эта команда может понадобиться);
- ☐ 3 — отключить HTTPS (SSL);
- ☐ 4 — отключить SSI (Server Side Includes);
- ☐ 5 — поддержка только IPv4 (по умолчанию поддерживаются IPv4 и IPv6);
- ☐ 6 — отключить `mod_perl`;
- ☐ 7 — отключить `Apache::ASP`;
- ☐ x — выход.

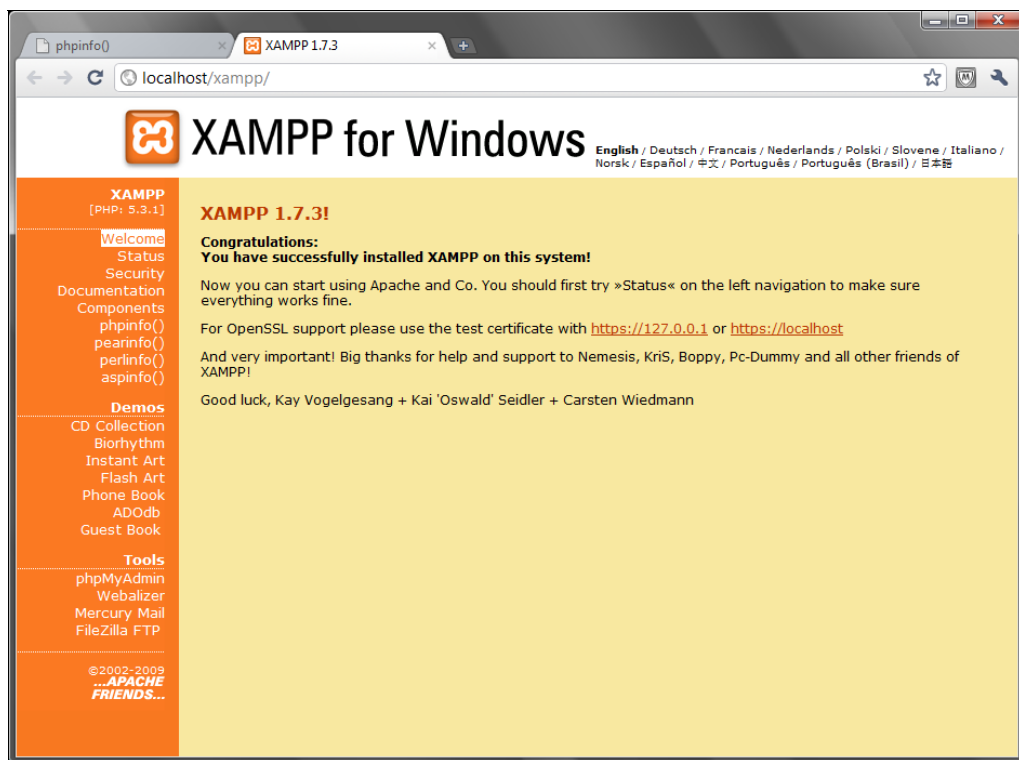


Рис. 1.11. Страница настройки Apache

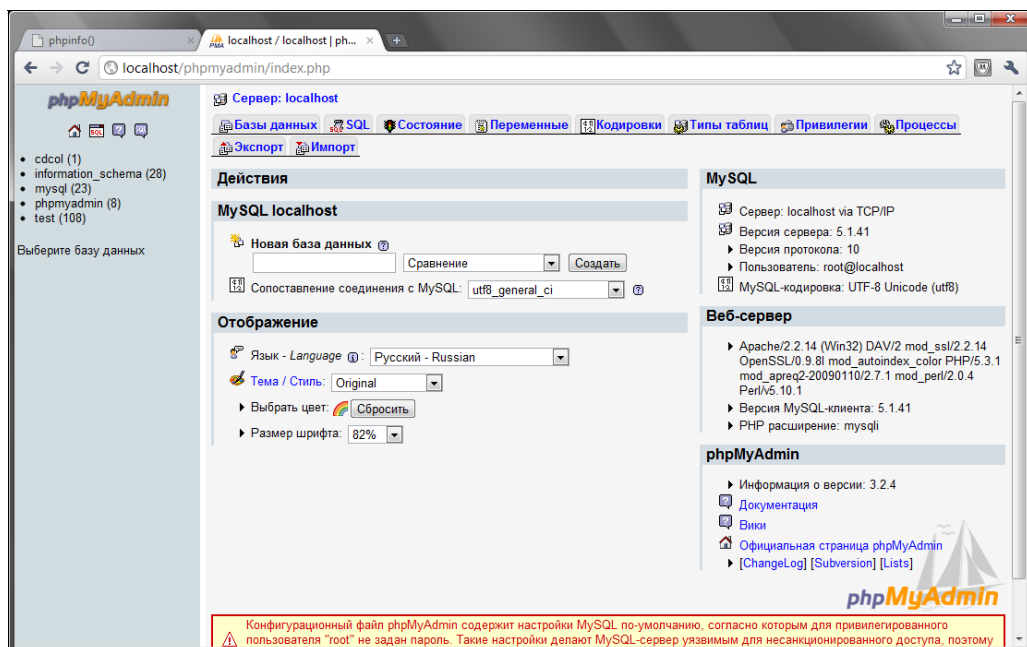


Рис. 1.12. Программа phpMyAdmin



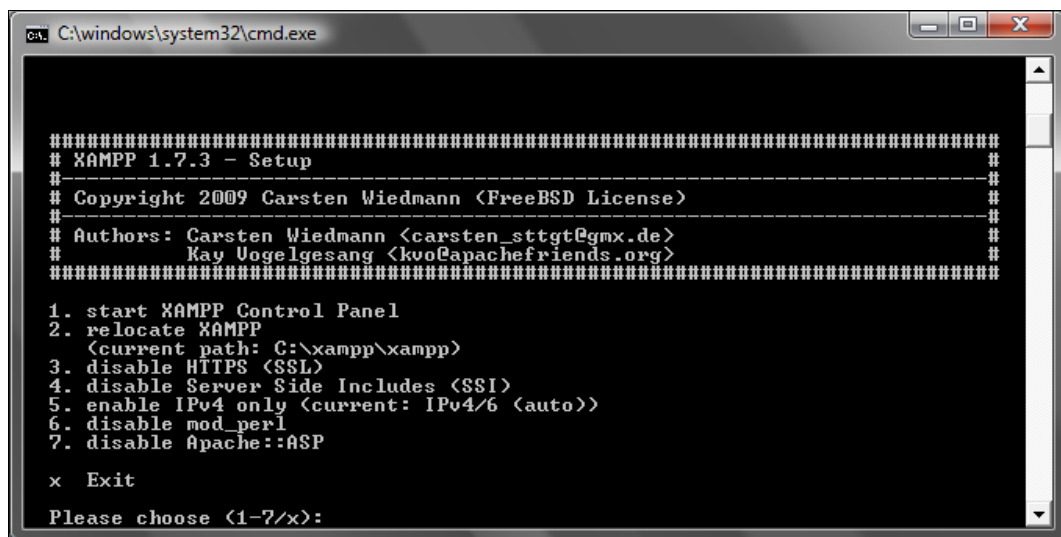


Рис. 1.13. Сервисное меню XAMPP

Для выхода из панели управления XAMPP используется кнопка **Exit** (см. рис. 1.10). Помните, что при выходе из нее запущенные сервисы не будут остановлены. Если вы хотите завершить работу сервисов, нажмите кнопку **Stop** напротив каждого запущенного сервиса, а затем уже нажимайте кнопку **Exit**.

Ваши PHP-сценарии при использовании XAMPP нужно поместить в подкаталог `htdocs` каталога установки XAMPP. Например, если вы установили XAMPP в каталог `c:\xampp`, то PHP-сценарии нужно поместить в каталог `c:\xampp\htdocs`.

При написании сценариев, использующих MySQL, вы можете применить следующие параметры доступа:

- ☐ сервер MySQL — `localhost`;
- ☐ имя пользователя — `root`;
- ☐ пароль — отсутствует;
- ☐ база данных по умолчанию — `test` (но вы можете создать другую БД с помощью `phpMyAdmin`).

## 1.4. Связываем PHP 5.4 и Apache

Все хорошо, но выполнив все предыдущие рекомендации, вы получите установленный сервер Apache с PHP 5.3, а это несколько не то, что мы хотели. Инсталлятор 5.3 производит всю настройку автоматически, но для версии 5.4 пока не создан инсталлятор, поэтому настраивать придется все вручную.

Установите Apache, как было показано ранее. По умолчанию Apache 2.2.22 устанавливается в каталог `C:\Program Files\Apache Software Foundation\Apache2.2\`. Вы же можете его установить в `C:\WWW`, чтобы было удобно добираться к своим сценариям и конфигурационным файлам.