

PHP 5

Э К С П Р Е С С -
К У Р С



- ▶ Основы языка PHP
- ▶ Простейшие сценарии Web-сервера
- ▶ Реляционные базы данных
- ▶ Работа с XML-данными
- ▶ Использование SOAP и Web-сервисов

УДК 681.3.06+800.92PHP5
ББК 32.973.26-018.1
Б90

Будилов В. А.

Б90 PHP 5. Экспресс-курс. — СПб.: БХВ-Петербург, 2005. — 240 с.: ил.
ISBN 5-94157-615-3

Рассмотрены базовые конструкции языка PHP 5, на примерах показаны основные приемы написания базовых сценариев, наиболее употребительных при разработке Web-приложений. Важное место в книге уделено способам работы с реляционными базами данных и с данными в формате XML. Подробно представлен материал об использовании стилей XSL для преобразования и представления XML-данных. Рассмотрены вопросы создания "с нуля" клиент-серверных приложений, разработки SOAP-приложений. Книга полезна как начинающим разработчикам, так и тем, кто уже имеет опыт создания интернет-приложений. Также она может использоваться как справочник.

Для Web-разработчиков

УДК 681.3.06+800.92PHP5
ББК 32.973.26-018.1

Группа подготовки издания:

Главный редактор	<i>Екатерина Кондукова</i>
Зам. главного редактора	<i>Евгений Рыбаков</i>
Зав. редакцией	<i>Григорий Добин</i>
Редактор	<i>Алия Амирова</i>
Компьютерная верстка	<i>Ольги Сергиенко</i>
Корректор	<i>Зинаида Дмитриева</i>
Дизайн обложки	<i>Игоря Цырульникова</i>
Зав. производством	<i>Николай Тверских</i>

Лицензия ИД № 02429 от 24.07.00. Подписано в печать 14.03.05.

Формат 70×100¹/₁₆. Печать офсетная. Усл. печ. л. 19,35.

Тираж 4000 экз. Заказ №

"БХВ-Петербург", 194354, Санкт-Петербург, ул. Есенина, 5Б.

Санитарно-эпидемиологическое заключение на продукцию
№ 77.99.02.953.Д.006421.11.04 от 11.11.2004 г. выдано Федеральной службой
по надзору в сфере защиты прав потребителей и благополучия человека.

Отпечатано с готовых диапозитивов
в ГУП "Типография "Наука"
199034, Санкт-Петербург, 9 линия, 12

ISBN 5-94157-615-3

© Будилов В. А., 2005
© Оформление, издательство "БХВ-Петербург", 2005

Оглавление

Введение	7
Глава 1. Первые странички	9
Язык гипертекстовой разметки HTML	9
Простейший HTML-файл	10
Параграфы	11
Разрыв строки	11
Заголовки	11
Выравнивание текста	12
Горизонтальная линия	13
Комментарии	14
Цвет фона документа	14
Варианты форматирования текста	15
Заранее отформатированный текст	16
Задание гиперссылок	17
Отображение документа по ссылке в новом окне	17
Фреймы	18
Горизонтальные фреймы	18
Плавающие фреймы	19
Таблицы	20
Ненумерованный список	22
Нумерованный список	22
Текстовое поле	23
Флажок	24
Выпадающий список	24
Текстовая область	26
Кнопка	26
Почта	27
Позиционирование HTML-элементов с помощью таблиц	28
Простая таблица	28
Вложенные таблицы	30
Объединение ячеек в таблице	32

Рисунки и карты ссылок	32
Внедрение рисунка в страницу	32
Рисунок фона	34
Свойства, указываемые в тэге <code></code>	35
Расположение картинки на странице	35
Форматирование изображений с использованием таблиц	37
Использование изображения в качестве ссылки	40
Динамический HTML	40
Положение элемента	41
Относительное положение <i>position: relative</i>	41
Видимость объектов	45
Свойство <i>z-index</i>	46
Объектная модель документа и DHTML	48
Обработка событий в DHTML	49
Рецепты для работы с объектами DHTML	53
Объект <i>window</i>	53
Объект <i>navigator</i>	59
Объект <i>event</i>	61
Объект <i>collection</i>	68
Объект <i>document</i>	71
Глава 2. Основы работы с PHP	75
Обработка HTML-форм	76
Простейший счетчик посещений	78
Отправка писем с помощью PHP	80
Передача файла серверу	84
Сессии	85
О правилах хорошего тона	86
Изображения в PHP	87
Глава 3. Основные понятия	95
Типы данных	95
Логический тип	95
Целые числа	96
Числа с плавающей точкой	96
Строки	96
Массивы	97
Приведение типов	98
Приведение к типу <i>boolean</i>	98
Приведение к типу <i>integer</i>	98
Приведение строк к числам	98
Переменные	99
Константы	99
Операторы	100
Арифметические операторы	101
Операторы присваивания	101
Побитовые операторы	101
Операторы сравнения	102

Оператор управления ошибками	102
Операторы увеличения и уменьшения на единицу	103
Логические операторы	103
Строковые операции	104
Управление ходом выполнения программы	104
Оператор <i>if</i>	104
Оператор <i>else</i>	104
Оператор <i>elseif</i>	105
Оператор <i>while</i>	105
Оператор <i>do ... while</i>	106
Оператор <i>for</i>	106
Оператор <i>foreach</i>	107
Оператор <i>break</i>	108
Оператор <i>continue</i>	109
Оператор <i>switch</i>	110
Функции	111
Классы	112
Оператор <i>extends</i>	113
Конструкторы	114
Деструктор	115
Область видимости членов класса	115
Оператор <i>::</i>	117
Ключевое слово <i>static</i>	118
Абстрактные классы	119
Интерфейсы	120
Оператор <i>final</i>	122
Глава 4. Работаем с данными	123
Данные в PHP	123
Работаем с MySQL	131
Выбор отдельного ряда	131
Выбор столбца	134
Сортировка строк результата	137
Использование нескольких таблиц одновременно	141
Получаем информацию о базах данных и таблицах	144
Некоторые классические примеры	147
Выбор максимального значения	147
Язык XML	152
Элементы XML	156
Имена элементов	158
Атрибуты XML	158
Корректно составленный XML-документ	161
Описание типа XML-документа	162
Стили XSL	174
Язык XSLT	174
Элемент <i><xsl:template></i>	178
Элемент <i><xsl:value-of></i>	180
Элемент <i><xsl:for-each></i>	181

Элемент <code><xsl:sort></code>	184
Элемент <code><xsl:if></code>	185
Элемент <code><xsl:choose></code>	187
Элемент <code><xsl:apply-templates></code>	191
Краткая справка по элементам XSLT.....	192
Язык XPath.....	194
Синтаксис языка XPath.....	195
Выбор ветвей.....	196
Выбор нескольких путей.....	196
Выбор атрибутов.....	196
Путь в XPath.....	197
Выражения XPath.....	201
Функции в XPath.....	202
XML и MS SQL.....	204
Глава 5. Сетевое программирование на PHP.....	209
Создание клиент-серверных приложений средствами PHP.....	209
Создание автономного сервера.....	210
Функция <code>socket_create</code>	212
Функция <code>socket_bind</code>	214
Функция <code>socket_listen</code>	214
Функция <code>socket_accept</code>	214
Функция <code>socket_write</code>	215
Функция <code>socket_read</code>	215
Создание клиентского приложения.....	215
Создание Web-служб средствами языка PHP.....	220
Web-службы.....	221
Создаем Web-службу.....	223
Доступ к простым объектам SOAP.....	226
Синтаксис SOAP.....	227
Элементы SOAP.....	228
Элемент <code><Envelope></code>	228
Конверт SOAP.....	228
Элемент <code><Header></code>	229
Элемент <code><Body></code>	229
Элемент <code><Fault></code>	229
Атрибуты SOAP.....	230
Атрибут <code>actor</code>	230
Атрибут <code>encodingStyle</code>	230
Атрибут <code>mustUnderstand</code>	231
Пример работы с SOAP.....	231
Ошибки SOAP.....	232
Заключение.....	235
Предметный указатель.....	237

ГЛАВА 1



Первые странички

Создать собственный сайт или страничку в Интернете нетрудно, для этого используется язык гипертекстовой разметки HTML. Чтобы ваша страничка ожила, стала интерактивной, необходимо, чтобы сервер, на котором она расположена, смог выполнить определенные действия перед отправкой странички в окно браузера. Описание таких действий содержится в программах. Программы, работающие совместно с Web-сервером, могут быть написаны на разных языках. Эта книга посвящена одному из наиболее популярных языков программирования для Web — языку PHP, но изучение этого языка невозможно без знания основ работы с языком гипертекстовой разметки HTML.

Язык гипертекстовой разметки HTML

HTML — это язык разметки гипертекста. Web-консорциум принял этот язык как стандарт представления информации в Интернете. HTML-файл можно создать в простейшем текстовом редакторе, например, в Блокноте. Для облегчения работы с HTML существует большое количество специально созданных редакторов, в числе наиболее популярных — редактор Microsoft FrontPage.

При загрузке HTML-файла браузер автоматически анализирует код HTML и на основе описанных в нем HTML-элементов создает соответствующие объекты. Созданные таким образом объекты подчиняются набору стандартных правил, образуя объектную модель документа (DOM). Элементы HTML описываются при помощи тэгов HTML. Сформированные браузером объекты обладают стандартным набором методов, свойств и событий, присущих им. Существуют средства, позволяющие управлять методами и свойствами объектов, а также реагировать на возникающие в них события. Наиболее

удобные средства для работы с объектами предоставляют языки сценариев, такие как JavaScript, VBScript.

Язык разметки гипертекста позволяет легко и просто организовать форматирование текста и графической информации.

Самый простой и эффективный способ научиться пользоваться средствами HTML — рассмотреть примеры.

Простейший HTML-файл

Каждый HTML-файл должен открываться тэгом `<html>`. Тело HTML-документа помещается в элементе, описываемом тэгом `<body>`. В листинге 1.1 приведен пример предельно простого HTML-файла.

Листинг 1.1. Простейший HTML-файл

```
<html>
<body>
Это простой текст в элементе 'body'.
</body>
</html>
```

Наберем приведенный выше код в простейшем редакторе, например в Блокноте, и сохраним под именем `file1.html`. Для просмотра в браузере нужно открыть этот файл, щелкнув на нем мышью, в результате появится наша страничка (рис. 1.1).

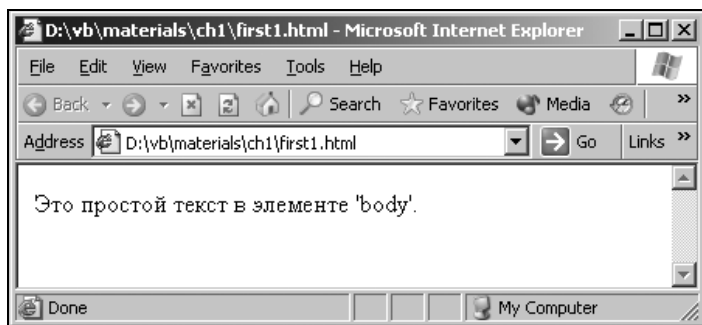


Рис. 1.1. Страничка загружена

Элемент `<body>` обладает набором свойств. В частности, оказывается полезным задание цвета фона или картинки, используемой в качестве такого фона. Цвет фона задается при помощи атрибута `bgcolor`, значением которого можно указать цвет в формате RGB (`<body bgcolor="#bcdebc">`) или задать цвет в виде его названия: `<body bgcolor="red">`. По правилам все значения

атрибутов элементов заключаются в кавычки. Заданные атрибуты преобразуются браузером в значения соответствующих свойств объектов.

В дальнейшем, чтобы проверить, как работают приведенные примеры, нужно сохранить их в файле с произвольным именем и расширением либо `html`, либо `htm`, после чего загрузить файл в браузер, щелкнув на нем мышью.

Параграфы

Для формирования параграфов используем тэг `<p>` и соответствующий ему закрывающий тэг `</p>` (листинг 1.2).

Листинг 1.2. Описываем параграфы

```
<html>
<body>
<p>Первый параграф.</p>
<p>Еще параграф.</p>
<p>Новый параграф.</p>
<p>Элемент параграфа описывается при помощи тэга p.</p>
</body>
</html>
```

Разрыв строки

Разрыв строки задается с помощью тэга `<br>` (листинг 1.3).

Листинг 1.3. Разрыв строки

```
<html>
<body>
Для простого <br> переноса <br> (разрыва) <br> строки используется другой
тэг - <br> br.
</body>
</html>
```

Заголовки

Заголовки шести уровней можно описать при помощи тэгов `<h1>`—`<h6>` (листинг 1.4).

Листинг 1.4. Заголовки

```
<html>
<body>
```

```
<p>Заголовки h1 - h6 используются для описания заголовков разных  
уровней</p>  
<h1>Заголовок первого уровня</h1>  
<h2>Заголовок второго уровня</h2>  
<h3>Заголовок третьего уровня</h3>  
<h4>Заголовок четвертого уровня</h4>  
<h5>Заголовок пятого уровня</h5>  
<h6>Заголовок шестого уровня</h6>  
</body>  
</html>
```

Заголовки помогают организовать представление текста наиболее выгодным образом (рис. 1.2).

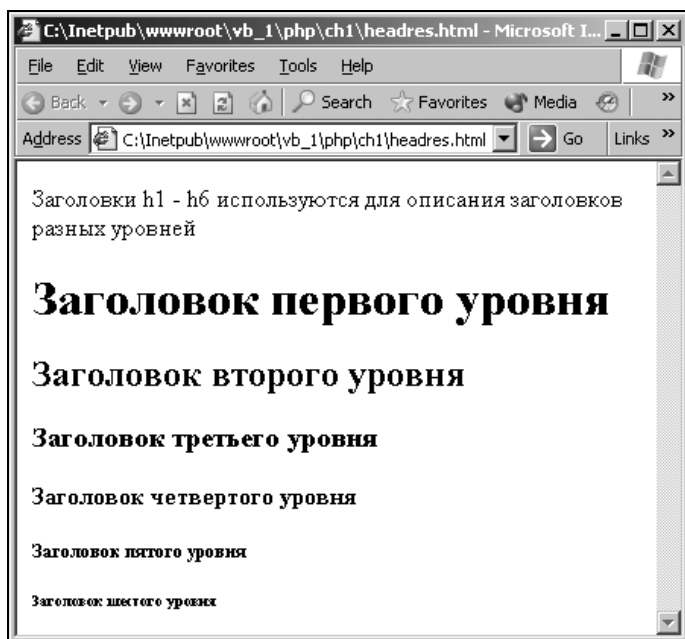


Рис. 1.2. Заголовки

Выравнивание текста

Приведем еще один пример использования атрибутов при работе с текстом (листинг 1.5). Текст можно выравнивать с применением атрибута `align`, значениями этого атрибута могут быть: `left` (выравнивание по левому краю), `right` (выравнивание по правому краю), `center` (выравнивание по центру).

Листинг 1.5. Выравнивание

```
<h1>Выравнивание</h1>
<p align="right">Многие текстовые элементы позволяют
задавать способ выравнивания текста.
Текст этого параграфа будет
выравнен по правому краю.</p>
```

На рис. 1.3 показан пример выравнивания текста по правому краю.

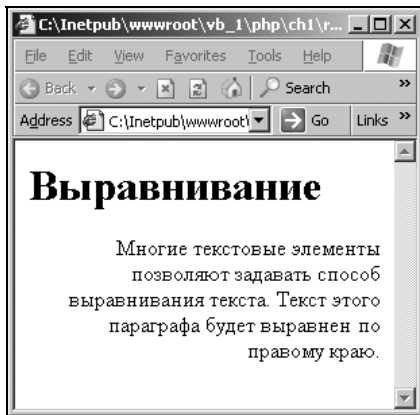


Рис. 1.3. Выравнивание текста по правому краю

Горизонтальная линия

Горизонтальная линия вставляется с помощью тэга `<hr>` (листинг 1.6).

Листинг 1.6. Горизонтальная линия

```
<html>
<body>
<p>тэг hr используется для вставки горизонтальной черты:</p>
<hr>
<p>Текст параграфа</p>
<hr>
<p>Текст параграфа</p>
<hr>
<p>Текст еще одного параграфа</p>
</body>
</html>
```

Для этого элемента можно указать его ширину с помощью атрибута `width`.

Комментарии

Комментарии служат для пояснения программного кода и не влияют на то, как будет отображаться HTML-код. Комментарии выделяются при помощи тэгов `<!-- ... -->` (листинг 1.7).

Листинг 1.7. Комментарии

```
<html>
<body>
<!-- Этот текст не будет отображен -->
<p>Комментарий не будет показан в окне браузера</p>
</body>
</html>
```

Использование комментариев особенно полезно тогда, когда код страницы становится сложным и большим.

Цвет фона документа

Цвет фона документа указывается в виде значения атрибута `bgcolor` (рис. 1.4), как это показано в листинге 1.8.

Листинг 1.8. Цвет фона

```
<html>
<body bgcolor="green">
<h2>Задаем цвет фона.</h2>
</body>
</html>
```

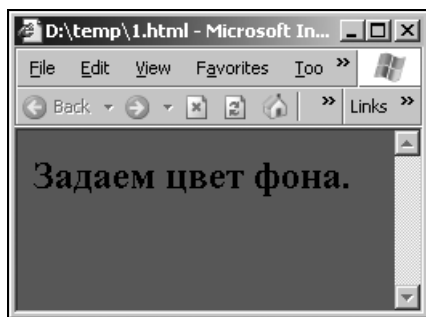


Рис. 1.4. Задан зеленый фон

Варианты форматирования текста

Ниже представлено несколько вариантов форматирования текста (листинг 1.9). Это форматирование осуществляется путем создания самостоятельных элементов, содержащих текст (рис. 1.5).

Листинг 1.9. Форматирование текста

```
<html>
<body>
<h3>Задаем форматирование текста</h3>
<b>Полужирный текст</b>
<br>
<strong>
Еще вариант выделения текста.
</strong>
<br>
<big>
Крупный текст
</big>
<br>
<em>
Выделенный текст
</em>
<br>
<i>
Текст курсивом
</i>
<br>
<small>
Маленький текст
</small>
<br>
Текст с
<sub>
нижним индексом
</sub>
<br>
Текст с
<sup>
верхним индексом
</sup>
</body>
</html>
```

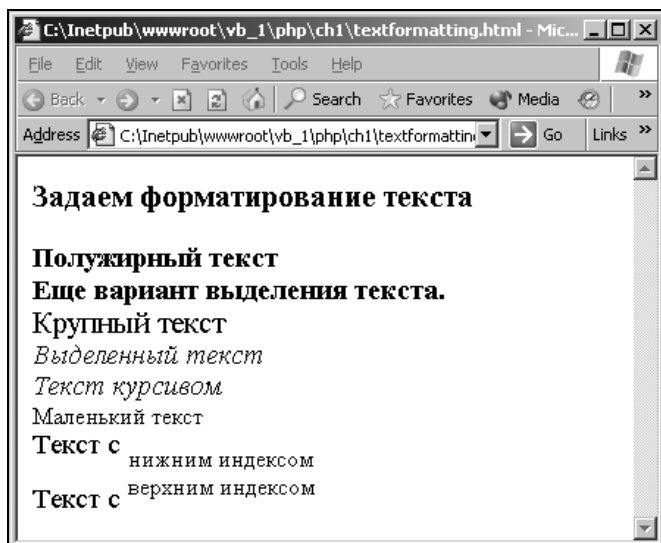


Рис. 1.5. Форматирование текста

Заранее отформатированный текст

Иногда требуется вывести на экран текст в том виде, в каком он изначально отформатирован. Это возможно с помощью тэга `<pre>`. При этом все переносы и пробелы будут выводиться так, как записано в тексте, и не будут игнорироваться, как это обычно происходит при отображении стандартного текста (листинг 1.10).

Листинг 1.10. Заранее отформатированный текст

```
<html>
<body>
<pre>
Это заранее отформатированный текст.
В этом тексте будут сохранены все пробелы
и
переносы
строки.
</pre>
<p>Такой формат часто используется для отображения исходных кодов
программ:</p>
<pre>
for k = 1 to 10
...
next k
```

```
</pre>  
</body>  
</html>
```

Задание гиперссылок

Один из важных моментов — задание гиперссылок. Это делается с помощью тега `<a href='URL'>`, как показано в листинге 1.11. URL — универсальный указатель ресурса (рис. 1.6).

Листинг 1.11. Гиперссылка

```
<html>  
<a href="http://www.yahoo.com/">Это гиперссылка. Щелкни меня!</a>  
</body>  
</html>
```

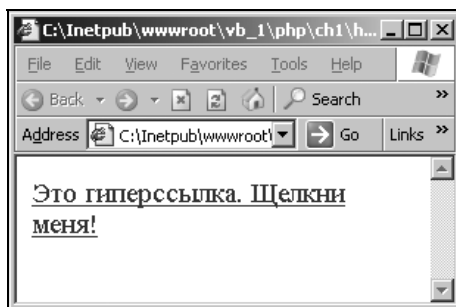


Рис. 1.6. Гиперссылка

Отображение документа по ссылке в новом окне

Чтобы документ, открытый по ссылке был выведен в новом окне, можно задать в качестве значения атрибута `target` для элемента `a` имя несуществующего окна (листинг 1.12).

Листинг 1.12. Гиперссылка в новом окне

```
<html>  
<body>  
<a href="http://www.yahoo.com" target="new_window">  
откроем документ в новом окне.</a>  
<p>
```

Если в качестве значения параметра `target` указать имя несуществующего окна (следует указать имя, которого нет ни у одного из открытых окон), то документ будет показан в новом окне.

```
</p>
```

```
</body>
```

```
</html>
```

Фреймы

Фреймы представляют собой самостоятельные окна в окне браузера. Пример задания фреймов приведен в листинге 1.13, а результат выполнения кода показан на рис. 1.7.

Листинг 1.13. Фреймы

```
<html>
<frameset cols="25%,50%,25%">
<frame src="a.html">
<frame src="b.html">
<frame src="c.html">
</frameset>
</html>
```

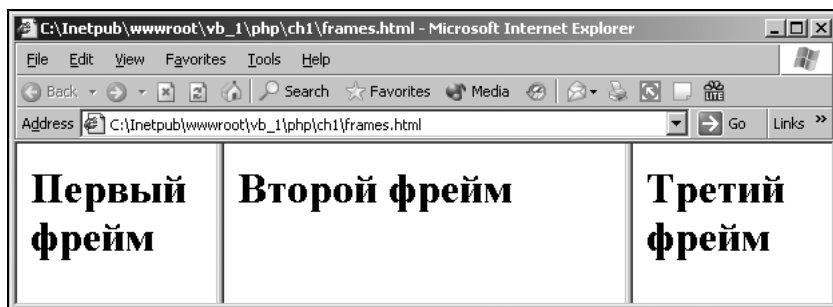


Рис. 1.7. Фреймы

Горизонтальные фреймы

В листинге 1.14 приведен пример задания горизонтальных фреймов (рис. 1.8).

Листинг 1.14. Горизонтальные фреймы

```
<html>
```

```
<html>
```

```
<frameset rows="25%,50%,25%">
<frame src="a.html">
<frame src="b.html">
<frame src="c.html">
</frameset>
</html>
```

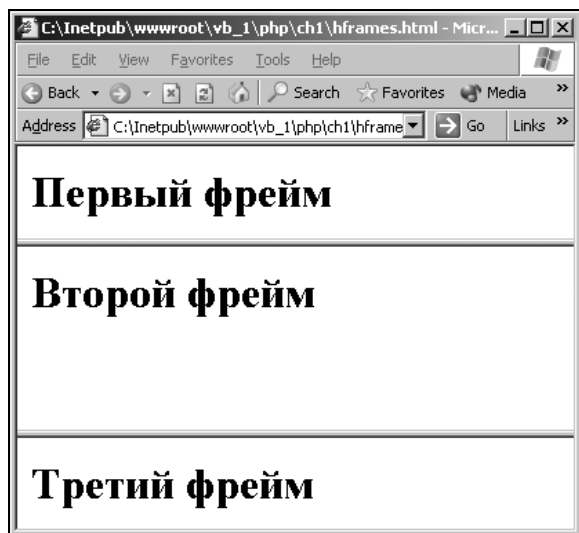


Рис. 1.8. Горизонтальный фреймы

Плавающие фреймы

Фрейм можно встроить в поток HTML-текста. Для этого используется тэг `<iframe>`. Пример кода приведен в листинге 1.15, а результат его выполнения — на рис. 1.9.

Листинг 1.15. Плавающий фрейм

```
<html>
<body>
<iframe src="http://yahoo.com"></iframe>
<p>Плавающий фрейм.</p>
<p>Такой фрейм размещается внутри контекста документа.</p>
</body>
</html>
```

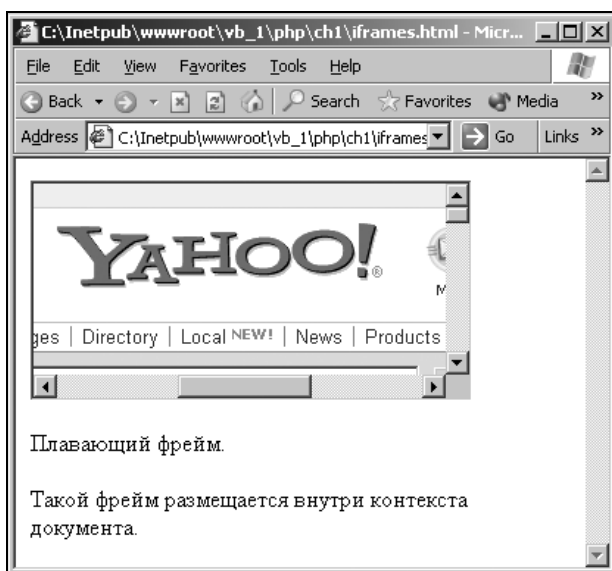


Рис. 1.9. Плавающий фрейм

Таблицы

Таблицы можно описать с применением тэга `<table>` (листинг 1.16 и рис. 1.10).

Листинг 1.16. Таблицы

```
<html>
<body>
<p>
Таблица открывается при помощи тэга table.
Строка таблицы открывается тэгом tr.
Каждая ячейка в строке начинается с тэга td.
</p>
<h4>Один столбец:</h4>
<table border="1">
<tr>
<td>1000</td>
</tr>
</table>
<h4>Одна строка и три столбца:</h4>
<table border="1">
<tr>
```

```
<td>1000</td>
<td>2000</td>
<td>3000</td>
</tr>
</table>
<h4>Две строки и три столбца:</h4>
<table border="1">
<tr>
<td>1000</td>
<td>2000</td>
<td>3000</td>
</tr>
<tr>
<td>4000</td>
<td>5000</td>
<td>6000</td>
</tr>
</table>
</body>
</html>
```

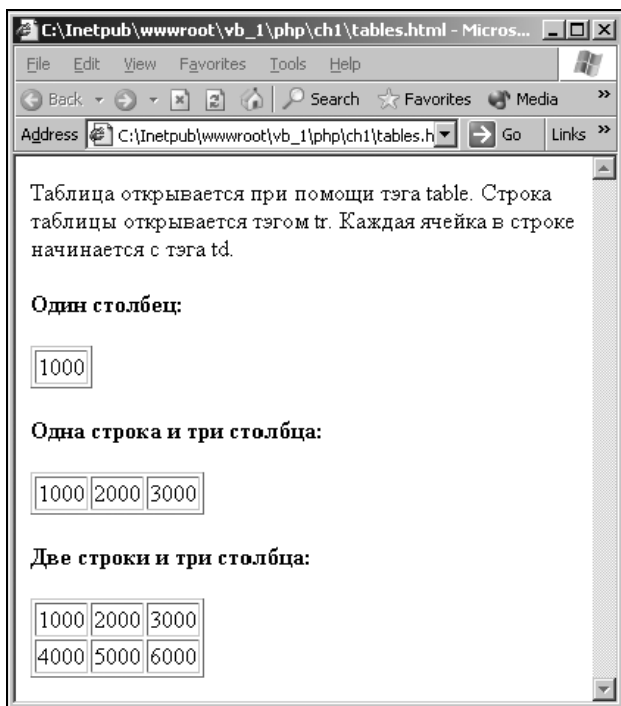


Рис. 1.10. Таблицы