

5 Добавление изображений на страницы

Знакомство с медиа



Улыбнитесь и скажите «сыр». Теперь улыбнитесь и скажите «gif», «jpg» или «png» — это те форматы файлов, которые вы выберете, создавая рисунки для Сети. В этой главе вы узнаете все о том, как добавить на веб-страницу свой первый медиафайл — изображение. У вас есть парочка цифровых фотографий, которые вы хотите поместить в Сеть? Никаких проблем. У вас есть логотип, который нужен на веб-странице? И это легко. Или, может быть, сначала вы хотите более близко познакомиться с элементом ``? К концу этой главы вы будете знать все мельчайшие подробности того, как использовать этот элемент и его атрибуты. Вы также узнаете, как этот небольшой элемент побуждает браузер делать такую серьезную работу по поиску и отображению ваших картинок.

Как браузер работает с изображениями

Браузер обращается с элементом `` немного иначе, чем с остальными элементами. Возьмем, к примеру, элемент `<h1>` или `<p>`. Когда браузер видит их на странице, все, что ему нужно, — отобразить их. Достаточно просто. Но когда браузер видит элемент ``, происходит совершенно другое: сначала он должен извлечь картинку и только потом сможет отобразить ее на странице.

Проще всего понять это на примере. Давайте еще раз взглянем на страницу с напитками из гостевой Head First, в которой есть четыре элемента ``:

```

<html>
  <head>
    <title>Напитки гостевой Head First</title>
  </head>
  <body>
    <h1>Наши напитки</h1>

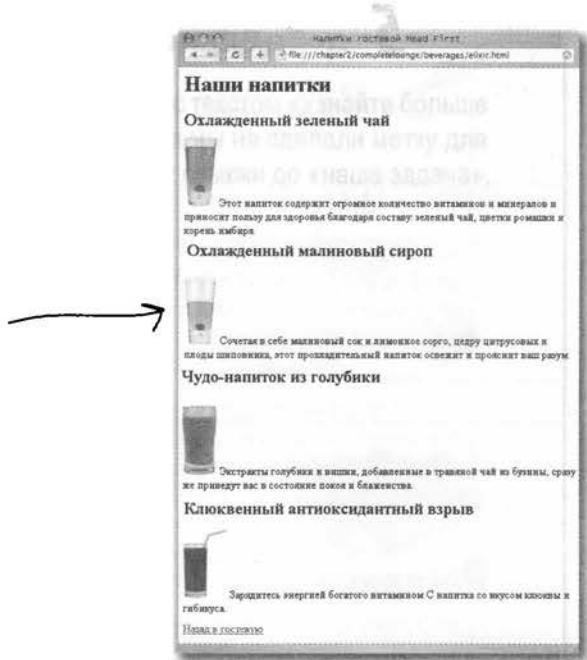
    <h2>Охлажденный зеленый чай</h2>
    <p>
      
      Этот напиток содержит огромное количество витаминов и минералов и приносит пользу для здоровья благодаря составу: зеленый чай, цветки ромашки и корень имбиря.
    </p>

    <h2>Охлажденный малиновый сироп</h2>
    <p>
      
      Сочетая в себе малиновый сок и лимонное сорго, цедру цитрусовых и плоды шиповника, этот прохладительный напиток освежит и прояснит ваш разум.
    </p>

    <h2>Чудо-напиток из голубики</h2>
    <p>
      
      Экстракти голубики и вишни, добавленные в травяной чай из бузины, сразу же приведут вас в состояние покоя и блаженства.
    </p>

    <h2>Клюквенный антиоксидантный взрыв</h2>
    <p>
      
      Зарядитесь энергией богатого витамином С напитка со вкусом клюквы и гибиуса.
    </p>
    <p>
      <a href="../lounge.html">Назад в гостевую</a>
    </p>
  </body>
</html>

```



В этом HTML
документе есть
четыре изображения.

Давайте теперь подробно рассмотрим этот код и поэтапно проследим, как браузер извлекает и отображает эту страницу, когда она запрашивается с сайта <http://lounge.headfirstlabs.com>.

Позади сцены

- Сначала браузер извлекает файл elixir.html с сервера.

Пустое окно браузера, ничего еще не извлечено.



Браузер

«Мне нужен файл elixir.html».

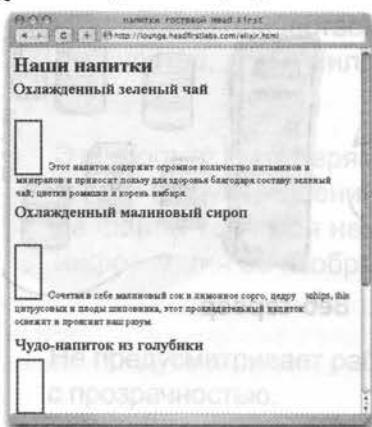
«Нашел, держи его».

Веб-сервер

Позади
сцены

- Затем браузер считывает файл elixir.html, отображает его и видит, что ему еще нужно найти четыре изображения. Итак, необходимо извлечь каждое из них с веб-сервера, начиная с green.jpg.

HTML-страница извлечена, но браузеру все еще нужно получить изображения.



Браузер

«О, кажется, мне еще нужен файл green.jpg».

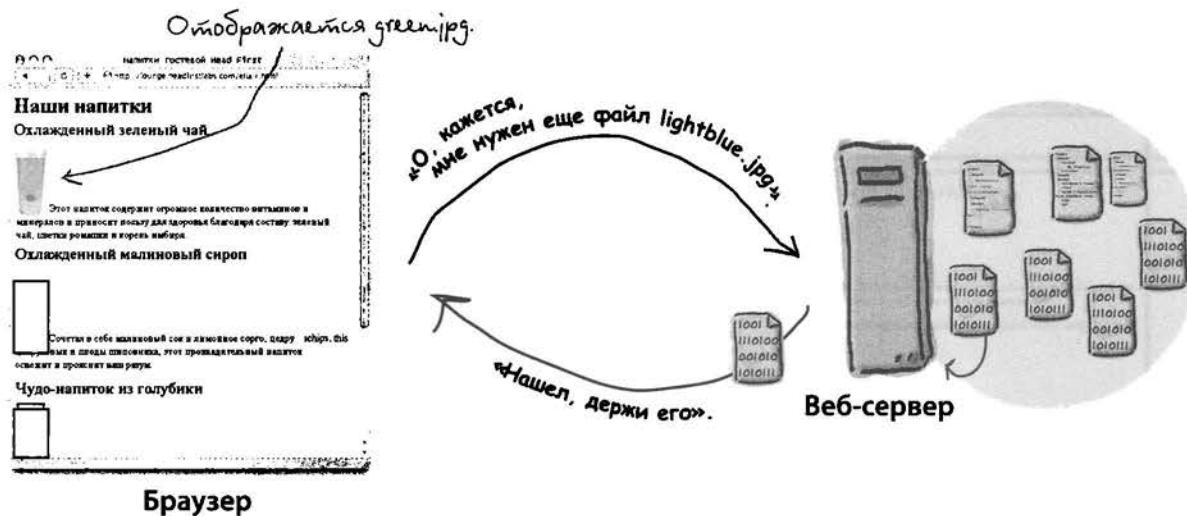
«Нашел, держи его».

Веб-сервер

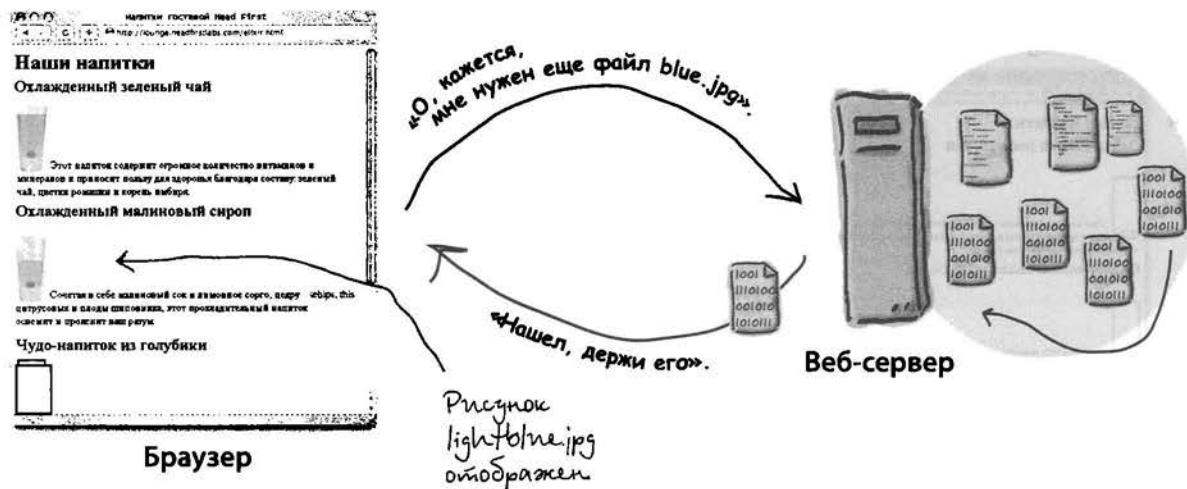
[далее >](#)

191

- 3 Как только браузер извлекает файл green.jpg, он отображает его и переходит к следующему изображению: lightblue.jpg.



- 4 Браузер извлек файл lightblue.jpg, выводит эту картинку, после чего переходит к следующей: blue.jpg. Этот процесс повторяется для каждого изображения на странице.



Как работают рисунки

Рисунки – это всего лишь рисунки, верно? На самом деле в мире существует огромное количество форматов изображений, и каждый имеет свои достоинства и недостатки. К счастью, в Сети обычно используются только два формата: JPEG и GIF. Единственная хитрость состоит в том, чтобы выяснить, какой из них и когда лучше использовать.

В чем разница между JPEG и GIF?



**Используйте JPEG
для фотографий и сложной графики**



**Используйте GIF для изображений с чистыми
цветами, логотипов и геометрических фигур**

Подходит для полутоновых не-растровых изображений, таких как фотографии.

Может выводить рисунки с большим количеством различных цветов, до 16 миллионов.

Это формат «с потерями», потому что при уменьшении размера файла теряется некоторая информация об изображении.

Не предусматривает работу с прозрачностью.

Подходит для изображений с несколькими чистыми цветами и таких штриховых изображений, как логотипы, ап-пликации. Его хорошо применять, если в изображении есть небольшой текст.

Может выводить рисунки с количеством цветов, не превышающим 256.

Сжимает файл, чтобы уменьшить его размер, но при этом не теряется никакая информация. Таким образом, это формат «без потерь».

Дает возможность установить «прозрачность» для цвета фона, чтобы сквозь этот фон было видно все, что находится под ним.

Беседа у камина



Вечерний диалог: JPEG и GIF сравнивают свои изображения.

JPEG

Снова здравствуй, GIF. Кажется, я только что видел тебя на веб-странице?

Ха. Как только ты научишься хорошо показывать такие сложные изображения, как фотографии, я уверен, что люди будут счастливы работать только с тобой, но ты до сих пор не знаешь, как отобразить что-нибудь, для чего нужно больше чем несчастные 256 цветов.

Ты хочешь поговорить со мной о качестве? Я позволяю пользователям точно указать, какое качество им нужно.

Это так, но большинство людей это более чем устраивает. Не каждому на его страницах нужны изображения с супервысоким разрешением. С моей помощью пользователи обычно выбирают низкое или среднее качество изображения и остаются вполне довольны этим. А если они используют тебя, то у них получается огромный файл с таким же изображением.

GIF

Да... разве не было бы здорово, если бы все просто работали с GIF? Тогда мне бы не пришлось стоять с тобой так часто.

Эй, нет ничего проще отображения фотографии, если *терять при этом ее качество*. Но для меня качество – самое главное. Если я не могу показать изображение полностью, я не возмусь за это вообще. Только взгляни на парочку логотипов, которые ты пытался отобразить... Тыфу.

Да, но какой ценой? Согласись, чтобы сжать фотографию до размера, приемлемого для пересылки по Сети, тебе придется ухудшить качество изображения.

Конечно, конечно, но смотрел ли ты когда-нибудь на штриховые изображения, логотипы, небольшой текст в изображении, чистые цвета? С JPEG они совсем не так хорошо выглядят.

JPEG

Да, конечно, GIF замечательно с ними спраивается, но только до тех пор, пока количество цветов совсем невелико. Ты – это как всего лишь уменьшенная версия меня. Я умею делать все то, что умеешь ты.

Что ты сказал, GIF? У нас здесь шоу. Куда ты ушел?

Я думаю, ты придаешь слишком много значения прозрачности. Я могу просто встроить цвет фона в изображение.

Ну, я не очень-то об этом беспокоюсь; существует не так уж много фотографий без фона.

И когда же такое может случиться?

Да, верно. Работай со своими логотипами и рисунками с простым текстом, а я буду работать с фотографиями и сложными изображениями. Все знают, что я лучше подхожу для работы со сложными вещами.

GIF

(GIF исчезает, буквально.)

(GIF появляется снова.)

Без паники. Я просто доказываю свою точку зрения. Если JPEG так велик, то как же так случилось, что ты не можешь делать отдельные части своих изображений прозрачными, как я? С использованием прозрачности можно видеть то, что находится под изображением. Если мои пользователи хотят поместить логотип на своих веб-страницах, а фон страницы цветной, то они выберут меня, потому что знают, что я позволяю видеть фон страницы сквозь нецветные части логотипа.

Конечно, а затем кто-то поменяет цвет веб-страницы. И что получится? Прозрачность – это единственный подходящий выход в подобной ситуации. Тебе придется меня использовать.

А как насчет того, чтобы вырезать на рисунке человека или даже дерево и использовать это на веб-странице без фона?

Ты будешь удивлен, узнав, как часто мне приходится отображать фотографии, только потому, что пользователи хотят видеть прозрачный фон.

Эй, кто-то просит меня установить прозрачность... Я побежал.

КАКОЙ ФОРМАТ ИЗОБРАЖЕНИЯ?

Наши поздравления: вы были назначены лучшим по выбору форматов изображений. Для каждого из приведенных ниже рисунков выберите формат, который позволит лучше отобразить этот рисунок в Сети.

JPEG или GIF

Ох, я не хочу показаться невежливым, но мы уже на девятой странице пятой главы, а вы ДО СИХ ПОР не представили меня! JPEG, GIF... Не могли бы вы побыстрее с этим закончить?

Сейчас перейдем к официальному представлению: познакомьтесь с элементом ****.

Мы достаточно долго откладывали процедуру знакомства. Как вы понимаете, с изображениями намного больше дел, чем с HTML-разметкой страницы. Так или иначе, хватит об этом... Настало время познакомиться с элементом ****.

Начнем с того, что более тщательно рассмотрим этот элемент (хотя вы, вероятно, к данному моменту уже имеете представление, как он работает):

Этот элемент ****. ↘

↑

↑

Вы уже знаете, что **** – это пустой элемент.

Атрибут **src** указывает на месторасположение файла с изображением, которое должно быть добавлено на веб-страницу.

Итак, это все? Не совсем. Есть парочка атрибутов, о которых вы захотите узнать. И, конечно же, вы захотите узнать, как использовать элемент ****, чтобы указать на изображения в Сети, которые находятся вне вашего сайта. Хотя в основном вы уже знаете, как работать с элементом ****.

Сейчас мы расскажем вам о некоторых тонкостях использования элемента ****, а затем вы попытаетесь применить полученные знания на практике.

: теперь не только относительные ссылки

Атрибут *src* может быть использован не только для относительных ссылок; вы также можете вставить в него URL-адрес. Изображения хранятся в Сети вместе с HTML-страницами, поэтому у каждого изображения есть свой собственный URL.

URL-адрес обычно применяется, если вы указываете на изображение, находящееся на другом сайте (помните, что для ссылок и изображений, расположенных на *том же* сайте, лучше использовать относительные пути).

Вот как вы ссылаетесь на изображение, используя URL-адрес:

```

```

Чтобы использовать URL, добавить на страницу рисунок, просто укажите весь адрес в качестве значения атрибута *src*.

URL — это путь к рисунку, поэтому имя файла в конце — всегда имя файла с данным рисунком. Нет такого понятия, как изображение, используемое по умолчанию.

Возьми в руку карандаш

Это упражнение на самом деле имеет отношение к карандашам (и к рисункам). В нем поднимается один простой вопрос. У вас есть обычный и совершенно новый карандаш. Если вы нарисуете сплошную линию, исписав все острие карандаша, то какой длины будет эта линия?

Какое это имеет отношение к изображениям? Чтобы найти ответ, вам придется написать небольшой HTML-код. Ответ на этот простой вопрос содержится в изображении, которое вы найдете по URL-адресу <http://www.headfirstlabs.com/trivia/pencil.gif>. Ваша задача — добавить изображение в приведенный код и получить ответ:

```
<html>
  <head>
    <title>Возьми в руку карандаш, простой вопрос</title>
  </head>
  <body>
    <p>Линию какой длины можно нарисовать обычным карандашом?</p>
    <p>Поместите сюда элемент с изображением.</p>
  </body>
</html>
```

часто
Задаваемые
Вопросы

В: Итак, использовать элемент `` достаточно легко. Он просто дает способ указать месторасположение изображения, которое должно быть выведено на странице?

О: Да, но это только краткое описание. Далее мы также поговорим о некоторых атрибутах, которые применяются вместе с этим элементом. Затем вы узнаете, как можно использовать CSS, чтобы поменять стиль изображения.

Но есть также много всего, что нужно знать о самих изображениях. Зачем нужны различные форматы? Когда нужно использовать один формат, а когда другой? Насколько большими должны быть рисунки? Как подготовить изображение к добавлению на веб-страницу?

В: Мы учили, что пустые элементы — это элементы без содержимого. Мы также учили, что элемент `` — пустой. Но разве он не имеет содержимого (изображение)?

О: Если говорить точнее, то пустой элемент — это элемент, не имеющий содержимого на HTML-странице, которое заключено между закрывающим и открывающим тегами. Конечно, изображение — это содержимое, но элемент `` лишь ссылается на него. Изображение само по себе не является частью HTML-страницы. Оно как бы заменяет элемент ``, когда браузер отображает страницу. И не забывайте, что HTML-страницы исключительно текстовые,

поэтому изображения никогда не смогут быть их непосредственной частью.

В: Насчет примера про загрузку изображений на веб-странице... Когда я загружал веб-страницу, я не заметил, чтобы изображения загружались по очереди. Почему?

О: Браузер часто извлекает все изображения сразу. Это значит, что он делает запросы на несколько рисунков одновременно. С учетом скорости компьютеров и сетей все это происходит достаточно быстро, поэтому вы обычно видите страницу сразу со всеми изображениями.

В: Если я вижу на веб-странице изображение, как мне определить его URL-адрес, чтобы можно было на него сослаться?

О: В большинстве браузеров есть возможность щелкнуть правой кнопкой мыши на изображении, чтобы появилось контекстное меню. Оно будет содержать пункт `Copy Image Address` (Скопировать адрес изображения) или `Copy Image Link` (Скопировать ссылку на изображение), выбор которых позволит поместить URL-адрес в буфер обмена. Второй способ найти URL — щелкнуть правой кнопкой мыши и выбрать в меню пункт `Open Image in New Window` (Открыть изображение в новом окне). В результате изображение откроется в окне браузера. Затем вы сможете взять URL рисунка из адресной строки браузера. Последний способ — использовать пункт меню браузера `View Source` (Посмотреть ис-

точник) и просмотреть HTML-код. Однако помните, что ссылка на изображение, которую вы найдете, может быть относительной и вам придется «реконструировать» URL, используя доменное имя сайта и путь к изображению.

В: Что делает JPEG-фотографию лучше фотографии GIF или GIF-логотип лучше логотипа JPEG?

О: Понятие «лучше» обычно определяется как комбинация качества изображения и размера файла. JPEG-фотография чаще всего намного меньше, чем фото в формате GIF такого же качества, в то время как логотип GIF обычно выглядит лучше и имеет меньший размер, чем тот же логотип в формате JPEG.

В: Я также слышал о формате изображений PNG. Почему вы не упомянули о нем?

О: PNG — это графический формат, который появился позже остальных. Он также достаточно интересен, так как может поддерживать и JPEG-, и GIF-стили изображений. Кроме того, он предоставляет больше возможностей по работе с прозрачностью, чем GIF. В настоящий момент PNG немного опережает события, потому что не все браузеры его поддерживают. Но его популярность растет очень быстро. Вы можете запросто использовать PNG, только знайте, что он будет работать не во всех браузерах.

Всегда имейте запасной вариант

То, в чем вы всегда можете быть уверены, когда имеете дело с Сетью, — что вы никогда не узнаете заранее, какие браузеры и устройства будут использоваться для просмотра вашей страницы. Всегда есть вероятность, что пользователи будут применять мобильные устройства, экранные дикторы, браузеры, которые работают с очень медленным Интернетом (и могут извлечь только текст сайта, но не изображения), мобильные телефоны, гостевые с выходом в Интернет... Кто знает?

Однако, несмотря на всю эту неопределенность, вы можете быть *ко всему подготовленными*. Даже если браузер не может показать изображения на вашей странице, есть альтернатива. Вы можете сообщить пользователю определенную информацию о том, что представляет собой изображение, благодаря атрибуту alt элемента ``. Вот как это работает:

```

```

Атрибуту alt просто нужен
небольшой кусок текста с описа-
нием изображения.

Если изображение не может
быть показано, то на его ме-
сте выводится этой текст.



Упражнение

В этом упражнении вы увидите, как браузер работает с атрибутом alt, когда появляется «отсутствующее» изображение. Теоретически, если рисунок не может быть найден, вместо него выводится значение атрибута alt. Однако не все браузеры делают это, поэтому ваши результаты в разных браузерах могут различаться. Рассмотрим, что вам нужно сделать.

- 1 Взмите HTML-код из предыдущего упражнения.
- 2 Подкорректируйте элемент `` так, чтобы в нем был атрибут alt со значением Длина линии, нарисованной карандашом, — 35 миль.
- 3 Поменяйте имя файла с изображением с `pencil.gif` на `broken.gif`. На самом деле такого файла не существует, поэтому вы получите «отсутствующее» изображение.
- 4 Обновите страницу в браузере.
- 5 И наконец, загрузите пару других браузеров и протестируйте страницу в них. Получились другие результаты?

Сверьте свои результаты с нашими, при-
веденными в конце главы.

Например, вы можете попробовать
Firefox (<http://www.mozilla.org/>) или
Opera (<http://www.opera.com/>).

Определение размеров изображений

Остался еще один атрибут элемента ``, о котором вы обязательно должны знать. Точнее, это пара атрибутов: `width` и `height`. Вы можете использовать их, чтобы заранее указать браузеру размер изображения на вашей странице.

Рассмотрим, как используются атрибуты `width` и `height`:

```

```

Атрибут `width` указывает браузеру, изображение какой ширины должно появиться на странице.

Атрибут `height` указывает браузеру, какой длины должно быть изображение.

В обоих атрибутах размер задается в пикселях. Если вы не знаете, что такое пиксели, то сможете прочесть о них немного позже в этой главе. Атрибуты `width` и `height` можно задавать для любого изображения. Если не указать их, то браузер автоматически определит размер изображения перед тем, как вывести его на экран.

часть Задаваемые Вопросы

В: Зачем вообще нужны эти атрибуты, если браузер все равно может вычислить размер изображений?

О: Если вы укажете ширину и высоту изображения в HTML-коде, то многие браузеры смогут распланировать структуру страницы перед тем, как ее отобразить. В ином случае браузеру придется менять планировку страницы после того, как он узнает размеры изображения. Помните, что браузер загружает изображения уже после того, как загрузит HTML-файл и начнет отображать страницу. И если вы отдельно не укажете размер изображения, то браузер не будет его знать до тех пор, пока не загрузит рисунок.

Вы также можете указать значения высоты и ширины, которые меньше или больше, чем размеры изображения, и браузер установит пропорции рисунка так, чтобы они соответ-

ствовали новым размерам. Многие люди делают так, когда им нужно вывести в браузере существующее изображение, уменьшив или увеличив его реальный размер. Тем не менее, как вы убедитесь чуть позже, есть множество причин не использовать для этого атрибуты `width` и `height`.

В: Я должен использовать эти атрибуты только в паре? Могу ли я просто указать ширину или высоту?

О: Можете, но, указывая только один размер, вы оставляете для браузера такой же объем работы, как если бы вообще не указали ни ширины, ни высоты (и не так уж часто бывает полезно указывать лишь высоту или лишь ширину, если только вы не хотите масштабировать изображение до определенной высоты или ширины соответственно).

В: Вы много раз говорили, что HTML предполагается использовать для структуризации, а не для дизайна. А эти атрибуты больше похожи на те, что применяются для оформления веб-страниц. Я не прав?

О: В зависимости от того, как вы используете эти атрибуты. Если вы просто устанавливаете нужные размеры для изображения, то это действительно лишь информация. Однако если вы используете `width` и `height`, чтобы изменить размеры изображения в браузере, то можно говорить, что вы применяете их для оформления веб-страницы. В этом случае, вероятно, лучше подумать об использовании CSS, что позволит добиться тех же результатов.

Создание сайта для самых больших фанатов: myPod

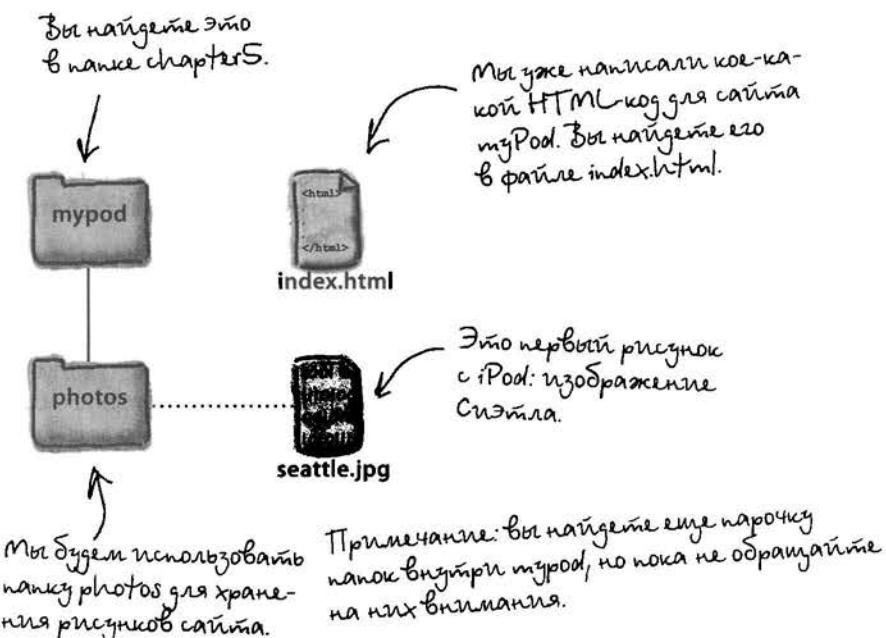
Обладатели iPod очень их любят и берут с собой повсюду. Представьте себе создание нового сайта под названием myPod, который будет содержать фотографии ваших друзей и их iPod в известных местах по всему миру.

Что вам нужно, чтобы приступить к делу? Всего лишь некоторые знания по HTML, парочка изображений и любовь к вашему iPod.

Мы уже написали кое-какой HTML-код для этого сайта, но еще не добавили рисунки. Именно тут вы приступаете к работе. Но перед тем, как вы доберетесь до рисунков, позвольте нам немного вам помочь; найдите папку chapter5 в файлах с примерами для книги. Здесь вы найдете папку myPod, которую нужно открыть. Вот что вы увидите внутри.



Мой iPod в Сиэтле! Здесь видны дождевые облака и башня Спейс Нидл. И не видно 628 кафе



Доработка файла index.html для сайта myPod

Открыв файл index.html, вы увидите, что работа над страницей myPod уже начата. Вот HTML-код, который есть на данный момент:

```
<html>
  <head>
    <title>myPod</title>
    <style type="text/css">
      body { background-color: #eaf3da; }
    </style>
  </head>
  <body>
    <h1>Добро пожаловать в myPod</h1>
    <p>
      Добро пожаловать туда, где можно похвастаться своим iPod и рассказать, где
      вы с ним были. Хотите повеселиться вместе с нами? Все, что вам для этого
      нужно, – это любой iPod: от ранней классической версии до самой современной
      нановерсии – самого маленького iPod с фотографиями лучшего качества
      и с цифровой камерой. Просто сфотографируйте свой iPod в любимом месте, и мы
      будем рады поместить этот снимок на сайте myPod. Итак, чего же вы ждете?
    </p>
    <h2>Сиэтл, Вашингтон</h2>
    <p>
      Я и мой iPod в Сиэтле! Здесь видны дождевые облака и башня Спейс Нидл.
      И не видно 628 кафе.
    </p>
  </body>
</html>
```

А вот как это
выглядит в браузе. Неплохо, но нам
нужны рисунки.



Мы вставили сюда готовый CSS-код. Пока просто оставьте его таким. Все, что он делает, – придает странице светлый, голубой фон. Обещаем, что начнем разбирать CSS через парочку глав!

Этот HTML-код выглядит привычно, так как используется основные строительные блоки: `<h1>`, `<h2>` и `<p>`.

Добро пожаловать туда, где можно похвастаться своим iPod и рассказать, где вы с ним были. Хотите повеселиться вместе с нами? Все, что вам для этого нужно, – это любой iPod: от ранней классической версии до самой современной нановерсии – самого маленького iPod с фотографиями лучшего качества и с цифровой камерой. Просто сфотографируйте свой iPod в любимом месте, и мы будем рады поместить этот снимок на сайте myPod. Итак, чего же вы ждете?



далее ▶

203



Возьми в руку карандаш

Как вы понимаете, уже написана большая часть HTML-кода, чтобы сайт myPod заработал. Все, что вам осталось сделать, — добавить элемент `` для каждой фотографии, которую вы хотите на нем разместить. Пока есть только одна фотография `seattle.jpg`, так что добавьте нужный элемент, чтобы поместить это изображение внизу страницы. Когда вы с этим справитесь, загрузите страницу в браузере и оцените виды Сиэтла.

```

<html>
  <head>
    <title>myPod</title>
    <style type="text/css">
      body { background-color: #eaf3da; }
    </style>
  </head>
  <body>

    <h1>Добро пожаловать в myPod</h1>
    <p>
      Добро пожаловать туда, где можно похвастаться своим iPod и рассказать,
      где вы с ним были. Хотите повеселиться вместе с нами? Вам для этого нужен
      любой iPod: от ранней классической версии до самой современной нановерсии —
      самого маленького iPod с фотографиями лучшего качества и с цифровой
      камерой. Просто сфотографируйте свой iPod в любимом месте, и мы будем рады
      поместить этот снимок на сайте myPod. Итак, чего же вы ждете?
    </p>

    <h2>Сиэтл, Вашингтон</h2>
    <p>
      Я и мой iPod в Сиэтле! Здесь видны дождевые облака и башня Спейс Нидл.
      И не видно 628 кафе.
    </p>

    <p>
    </p>
  
```

↑
Здесь вы должны поместить
свою первую фотографию.

Нам нужно
изображение.

← Важный элемент ``
должен появиться
здесь.

```

    </body>
  </html>

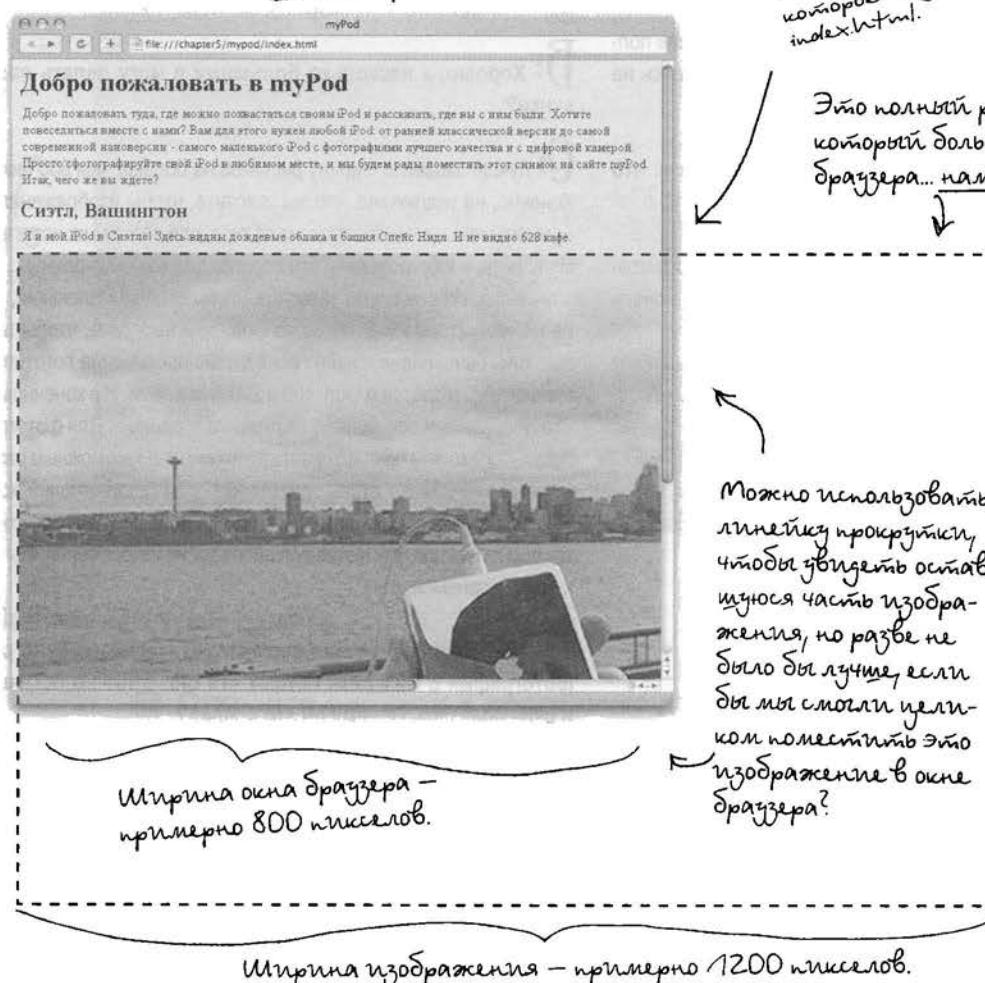
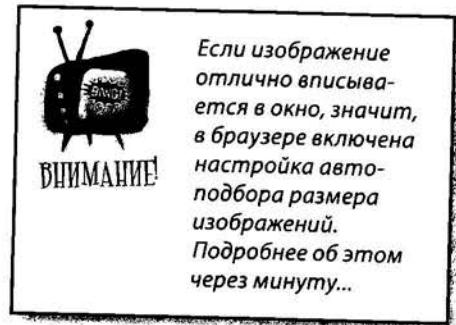
```



Ого! Рисунок слишком большой

Итак, изображение находится как раз там, где должно, но оно слишком большое. И, кстати, почти все фотографии, сделанные современными цифровыми камерами, такие же большие (и даже больше). Может быть, нам оставить изображение таким, какое оно есть, и предположить, что посетители сайта будут использовать линейку прокрутки? Дальше вы увидите, что это плохая идея.

Давайте взглянем на изображение и на браузер и поймем, насколько все плохо выглядит.



часто Задаваемые Вопросы

В: Что плохого в том, чтобы посетитель использовал линейку прокрутки для просмотра всего изображения целиком?

О: Веб-страницами с большими изображениями пользоваться очень неудобно. Дело не только в том, что посетители не могут видеть весь рисунок целиком, — само по себе использование линейки прокрутки обременительно. Кроме того, при работе с большими изображениями сервер и браузер обмениваются большими объемами данных, что занимает много времени и может привести к тому, что ваша страница будет медленно грузиться, особенно у тех, кто использует соединение по телефонной линии и другие медленные подключения.

В: Почему я не могу просто использовать атрибуты `width` и `height`, чтобы поменять размеры изображений на странице?

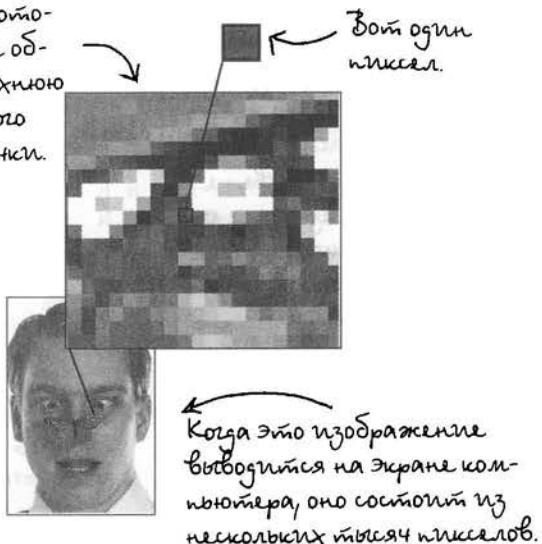
О: Потому что браузеру все-таки нужно извлечь изображение полностью перед тем, как уменьшить его, чтобы оно поместилось на вашей странице.

В: Вы сказали, что ширина окна браузера 800 пикселов. Что это означает?

О: Экран монитора состоит из миллионов точек, называемых пикселями. Вы сможете их увидеть, если посмотрите на монитор с очень близкого расстояния.

В то время как размеры экранов мониторов и их разрешения имеют тенденцию различаться (у некоторых людей мониторы большие, у не-

Задаваемые
вопросы, которые вмес-
то об-
разуют верхнюю
часть правого
крыла бабочки.



которых маленькие), 800 пикселов — это наиболее часто используемая ширина окна браузера, которую люди для себя устанавливают. Итак, 800 пикселов — это та максимальная ширина ваших изображений, которой хорошо бы придерживаться на практике (мы вернемся к этому в следующих главах).

В: Какова зависимость размера изображения на экране от количества пикселов?

О: Примерно такова: 72 пикселя на каждый дюйм, хотя, в зависимости от вашего монитора, может быть и 120 пикселов на дюйм. Допустим, разрешение вашего экрана составляет 72 пикселя на каждый дюйм. Если вы хотите, чтобы изображение было приблизительно 3 дюйма в ширину и высоту, то задайте размер, равный $72 \times 3 = 216$ пикселов, или, округлив, 250×250 пикселов.

В: Хорошо, а насколько большими я могу делать свои рисунки?

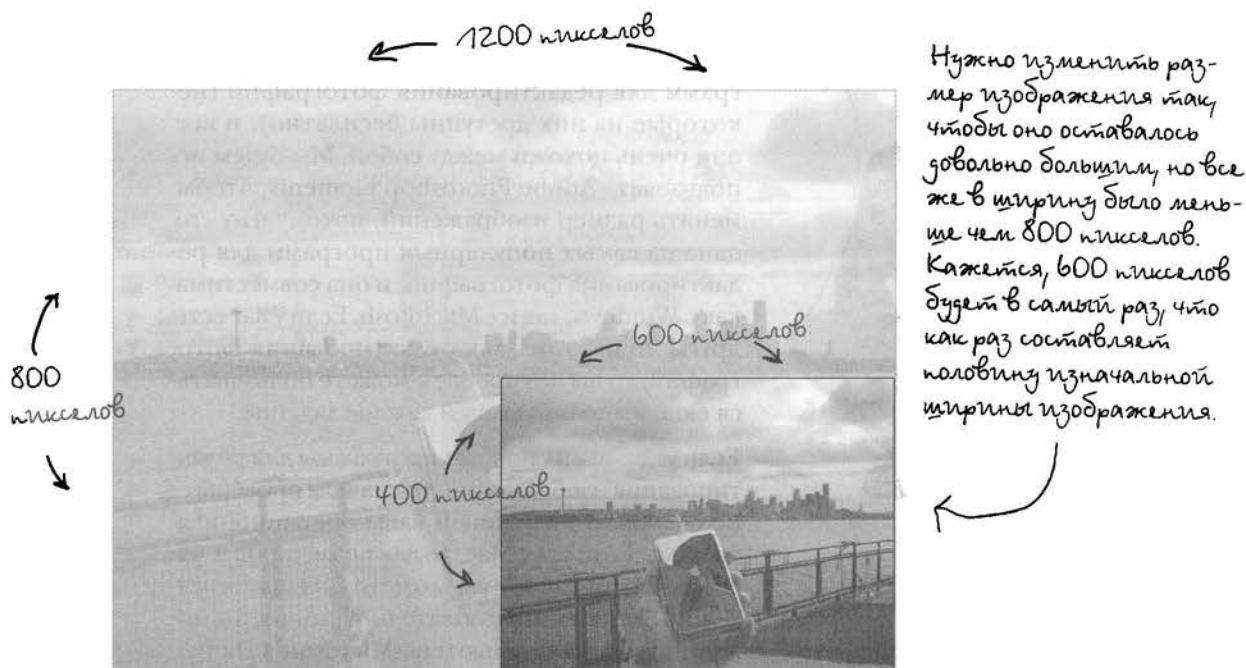
О: Лучше задавать ширину рисунков не больше чем 800 пикселов. Конечно, не исключено, что вы захотите, чтобы изображения были намного меньше, — это зависит от того, для чего вы их используете. Например, изображение — это логотип для вашей страницы. В этом случае вы, скорее всего, захотите, чтобы он был маленький, но тем не менее читаемый. В конце концов, вам не нужно, чтобы ширина логотипа была равна ширине всей страницы. Ширина логотипов, как правило, колеблется между 100 и 200 пикселями. И в конечном счете ответ на ваш вопрос зависит от дизайна страницы. Для фотографий, которые вы хотите просматривать с максимально возможным разрешением, можно создать страницу с миниатюрами изображений, которые быстро загружаются, и дать возможность пользователю щелкнуть на каждом из них для просмотра большой версии изображения. Мы вскоре к этому вернемся.

В: Мне кажется, что браузер автоматически изменил размер фотографии с Сэтлом, потому что она отлично помещается в окне браузера. Почему он это сделал?

О: У некоторых браузеров есть возможность изменения размера изображения, если его ширина больше ширины окна. Но многие браузеры такого не делают, поэтому не нужно на это полагаться. Если бы даже у каждого браузера было такое свойство, все равно браузер и сервер обменивались бы намного большим количеством данных, чем необходимо, а это замедлило бы загрузку страницы и сделало бы ее менее практичной.

Изменение размера изображения

Давайте изменим размер этого рисунка так, чтобы он помещался в окно браузера. Сейчас размеры такие: 1200 пикселов в ширину и 800 пикселов в высоту (через минутку вы узнаете, как это определить). Поскольку мы хотим, чтобы ширина рисунка была не более 800 пикселов, мы должны точно определить, изображение какой ширины подойдет для нашей веб-страницы iPod наилучшим образом. Весь смысл страницы заключается в просмотре фотографий iPod, поэтому мы, скорее всего, захотим, чтобы изображения были достаточно большими. Если мы уменьшим размеры рисунков до 600 пикселов в ширину и 400 пикселов в высоту, то они все равно будут занимать большую часть ширины окна браузера, но появятся отступы по бокам. Звучит хорошо? Давайте изменим размер этого изображения...



Вот что вам нужно сделать.

- 1 Откройте рисунок в программе для редактирования фотографий.**
- 2 Уменьшите размер изображения в два раза (до 600×400 пикселов).**
- 3 Сохраните рисунок под названием seattle_med.jpg.**



Перед тем как приступить к работе, давайте выясним, какую программу для редактирования фотографий использовать, чтобы изменить размер изображения. У меня есть Photoshop Elements. Она подойдет?

Хороший вопрос. Сейчас есть множество программ для редактирования фотографий (несколько из них доступны бесплатно), и все они очень похожи между собой. Мы будем использовать Adobe Photoshop Elements, чтобы менять размер изображений, потому что это одна из самых популярных программ для редактирования фотографий, и она совместима как с Windows, так и с Macintosh. Если у вас есть другая программа для редактирования фотографий, то вы без проблем можете пользоваться ею и выполнять в ней каждое задание.

Если у вас до сих пор нет программы для редактирования изображений, то сначала проверьте, не содержит ли таковой ваша операционная система. Если у вас Mac, то для редактирования своих фотографий вы можете пользоваться iPhoto. Если вы пользователь Windows, то можете найти на компьютере Microsoft's Digital Image Suite. Если же у вас все-таки нет подходящей программы, все равно продолжайте работу и для каждого шага используйте HTML-код и рисунки из папок с примерами.

Если у вас нет Adobe Photoshop Elements, но вы хотите использовать именно эту программу, чтобы продолжить обучаться по этой главе, можете скачать пробную версию и пользоваться ею в течение 30 дней. Этот адрес, где вы сможете ее найти: <http://www.adobe.com/products/triadobe/main.jsp>.

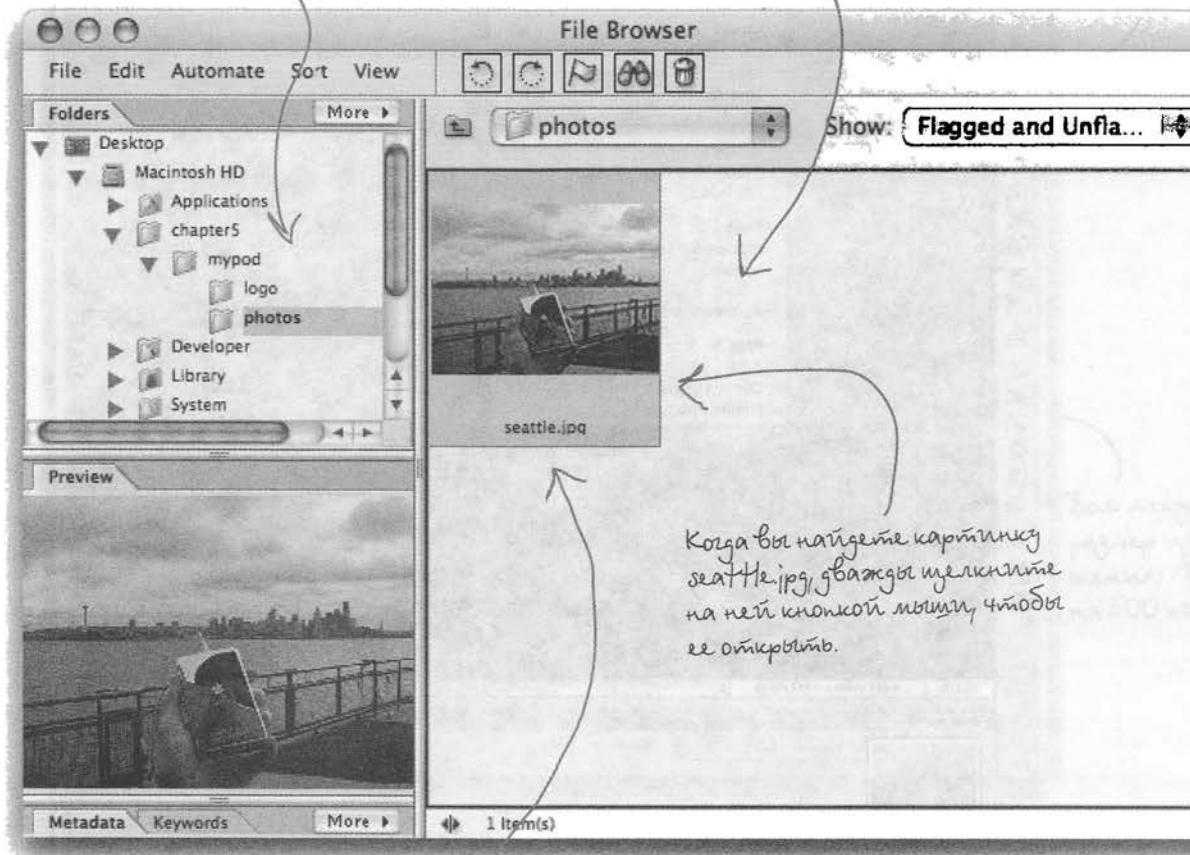
Открытие изображения

Во-первых, запустите программу для редактирования изображений и откройте в ней файл seattle.jpg. В Photoshop Elements вам нужно будет выбрать пункт **Browse Folders** (Открыть) в меню **File** (Файл), после чего откроется окно **File Browser** (Открыть). Вы будете его использовать, чтобы найти файл seattle.jpg в папке chapter5/mypod/photos.

Ах да, если вы работаете в Windows, то сразу используйте меню **File Open** (Файл>Открыть), чтобы открыть картинку seattle.jpg.

Запустите окно **File Browser** (Открыть). Используйте его, чтобы найти картинку seattle.jpg.

Перемещаясь по папкам, вы будете видеть в них эскизы изображений.

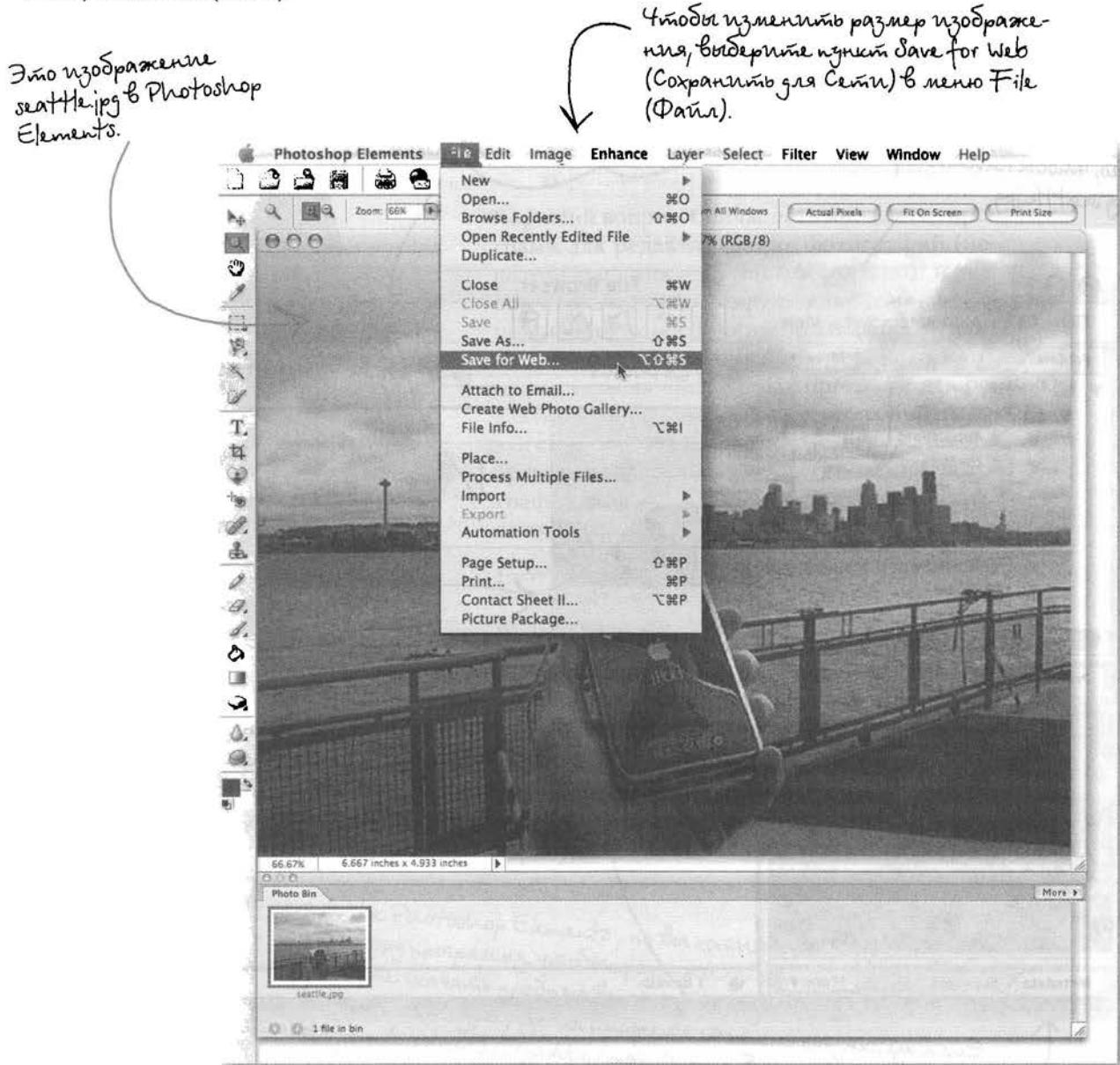


Если щелкнуть кнопкой мыши на эскизе изображения, то появится его большой вариант.

Когда вы найдете картинку seattle.jpg, дважды щелкните на ней кнопкой мыши, чтобы ее открыть.

Изменение размера изображения

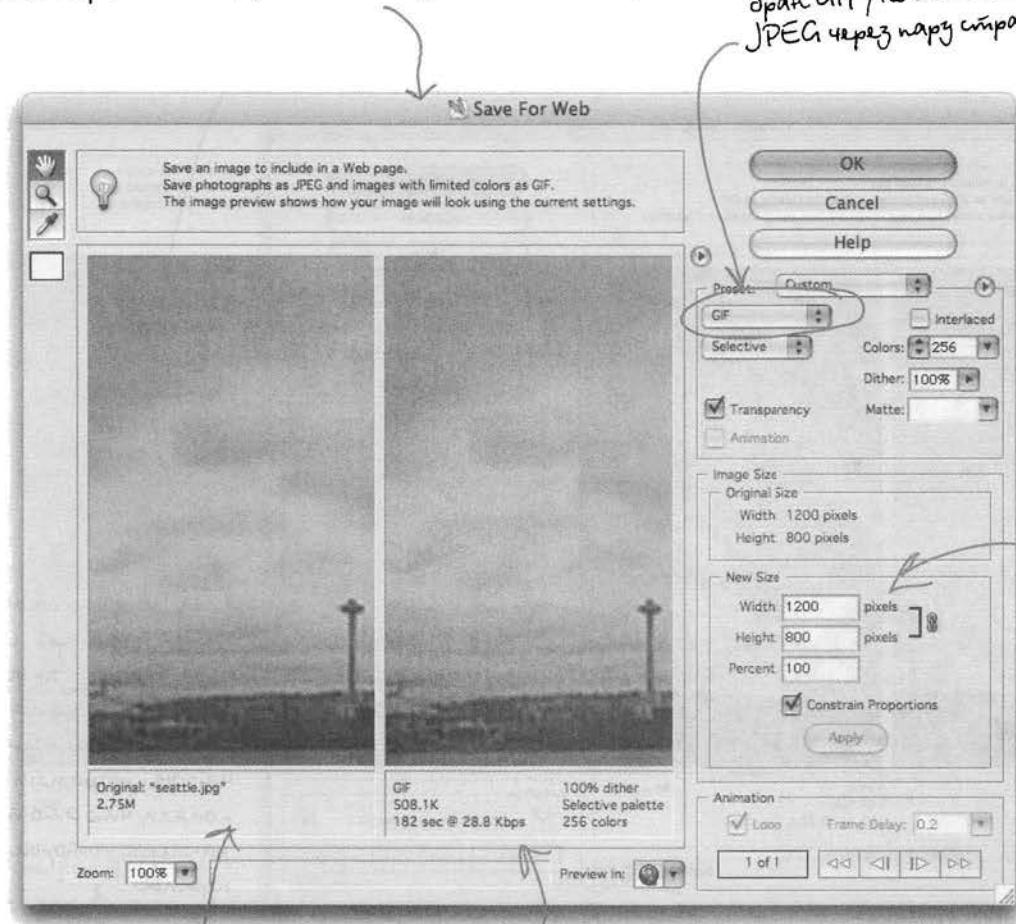
Теперь, когда файл seattle.jpg открыт, для изменения размера рисунка и его сохранения мы будем использовать окно Save for Web (Сохранить для Сети). Чтобы открыть это окно, просто выберите пункт Save for Web (Сохранить для Сети) в меню File (Файл).



Изменение размера изображения, продолжение...

Выбрав пункт меню Save for Web (Сохранить для Сети), вы увидите одноименное окно; давайте познакомимся с ним перед тем, как начать использовать.

В этом окне вы можете делать множество интересных вещей. На данный момент мы сфокусируемся на том, как его использовать для изменения размера изображения и сохранения в формате JPEG для веб-страницы.



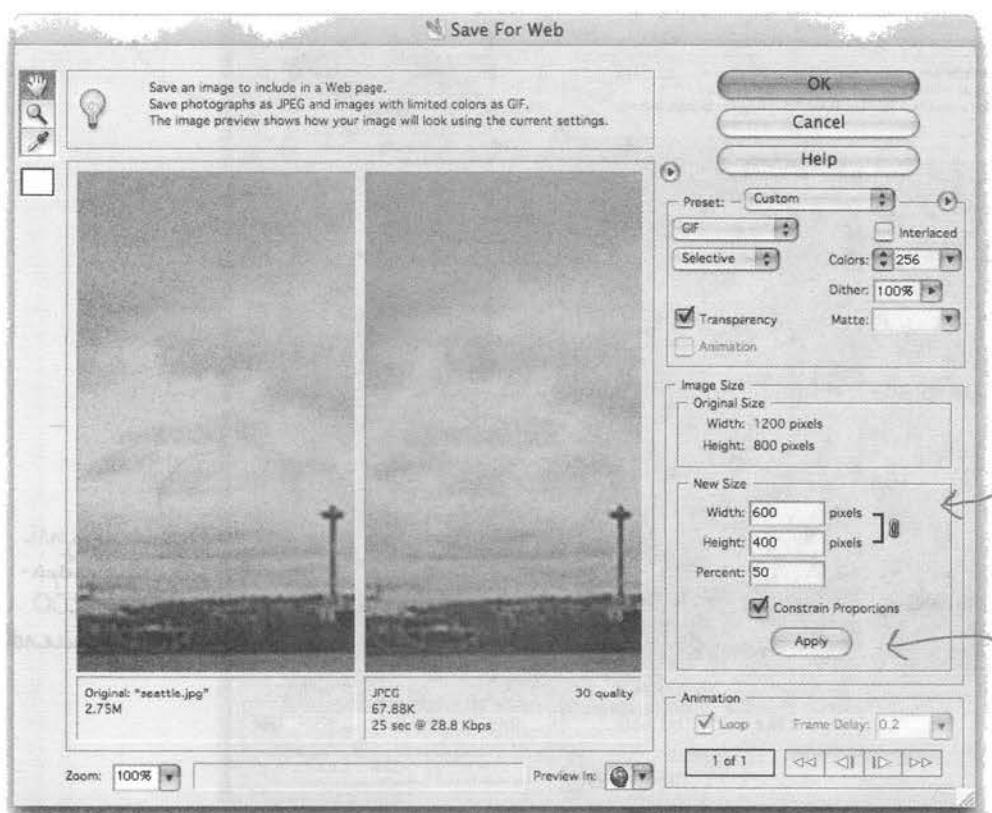
Здесь вы выбирайте формат для сохранения файлов. Сейчас выбран GIF, но мы поменяли его на JPEG через пару страниц.

В окне слева показывается ваше первоначальное изображение, а справа - изображение в том формате, в котором вы будете сохранять его для Сети. Сейчас в нем показан формат GIF; далее мы поменяем его на JPEG.

далее ▶

211

Как видите, это окно содержит множество функциональных возможностей. Давайте начнем ими пользоваться. Чтобы изменить размер изображения, нужно изменить ширину на 600 пикселов и высоту на 400 пикселов. Затем следует сохранить изображение в формате JPEG. Начнем с изменения размера изображения...



Это повлияет не на первоначальное изображение, а на файл, который вы хотите сохранить.

Вы должны нажать кнопку **Apply** (Применить), чтобы уменьшить размер изображения; в противном случае оно сохранится с исходной шириной и высотой.

(1) Измените размер изображения так: ширина – 600 и высота – 400. Если у вас установлен флајжок **Constrain Proportions** (Сохранить пропорции), то все, что нужно сделать – ввести ширину 600, а программа автоматически изменяет высоту на 400.

(2) После того как высота и ширина корректно заданы, нажмите кнопку **Apply** (Применить), чтобы программа поняла, что это те размеры, которые вам нужны.

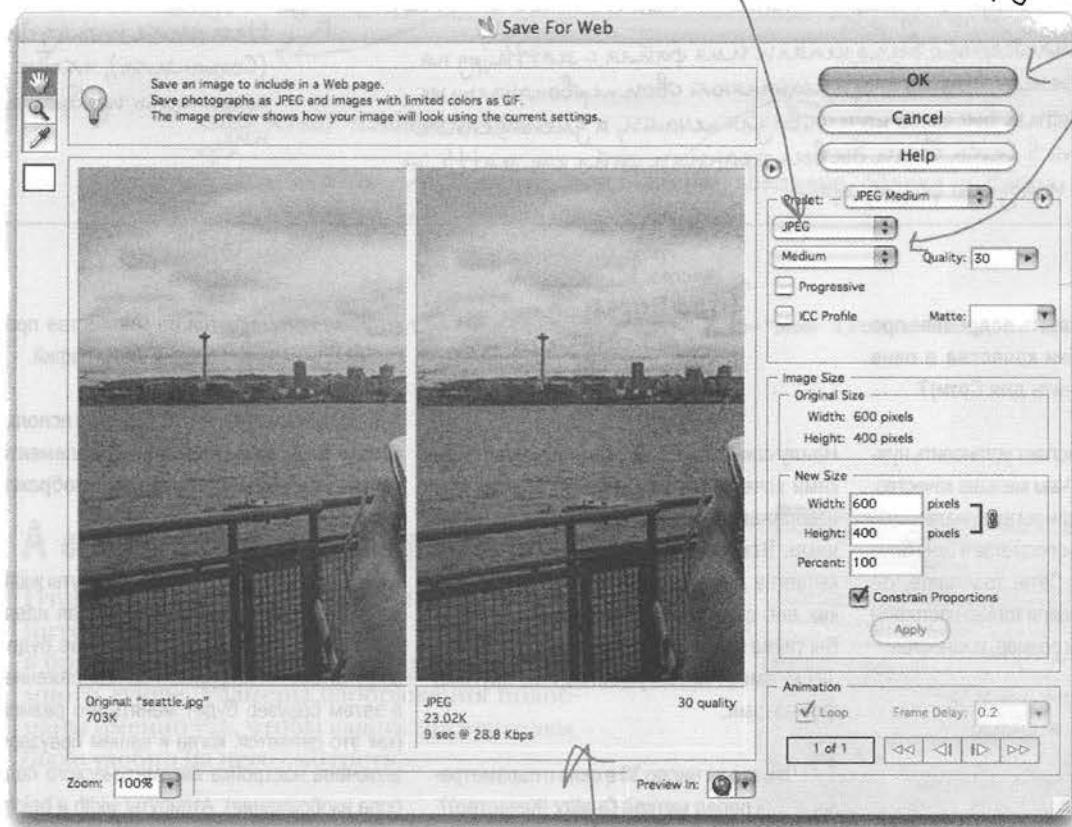
Размеры изменены, теперь сохраняем

Сейчас вам всего лишь нужно сохранить изображение в правильном формате (JPEG). Для этого следует выбрать данный формат и установить качество Medium (Среднее). Подробнее о качестве мы поговорим немного позже.

(1) Теперь, когда установлен размер изображения, остается выбрать для него формат. Сейчас выбран GIF; поменяйте его на JPEG, как мы это здесь сделали.

(2) Установите качество Medium (Среднее).

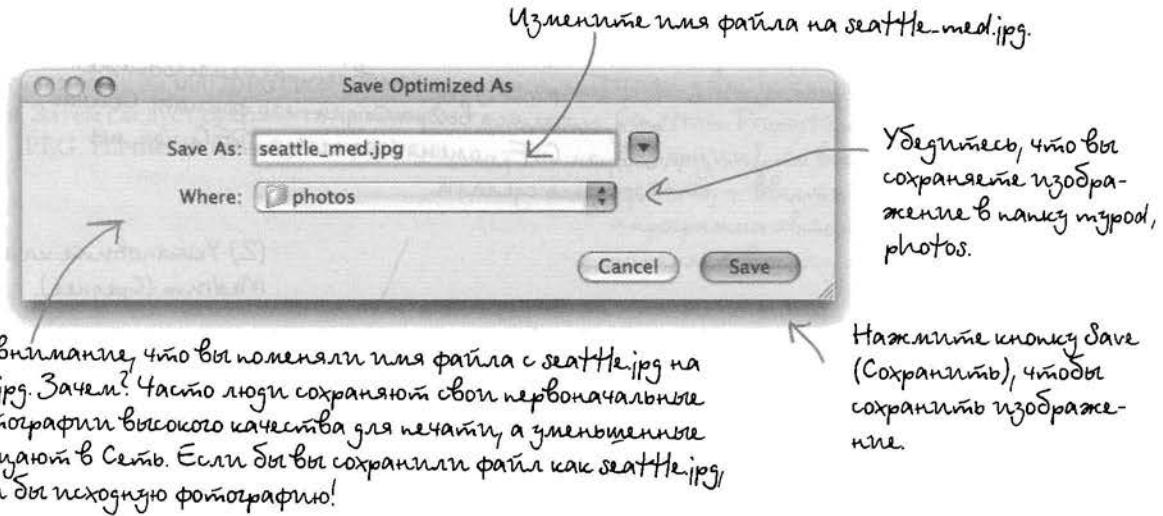
(3) Это все. Нажмите кнопку OK и переходите к следующей странице.



Обратите внимание, что после того, как вы нажали кнопку Apply (Применить) в предыдущем шаге, размер изображения был изменен и оно отобразилось с новыми размерами.

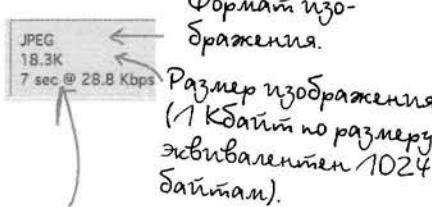
Сохранение изображения

После того как вы нажмете кнопку OK, появится окно для сохранения файла. Сохраните рисунок под именем seattle_med.jpg, чтобы не потерять первоначальную фотографию.



В: Вы можете рассказать подробнее про параметры настройки качества в окне Save for Web (Сохранить для Сети)?

О: Формат JPEG позволяет установить нужный уровень качества. Чем меньше качество, тем меньше размер. Если вы взглянете на панель предварительного просмотра в окне Save for Web (Сохранить для Сети), то увидите, что после того, как вы поменяли только настройки качества, изменились и размер, и качество.



Photoshop Elements даже говорит вам, как много времени займет передача файла браузеру при соединении по телефонной линии.

Задаваемые вопросы

которые применяются во множестве программ для редактирования фотографий.

В: Нельзя ли вместо этого просто использовать атрибуты width и height элемента , чтобы изменить размер изображения?

О: Вы можете использовать атрибуты width и height, но это не совсем хорошая идея. Почему? Потому что в этом случае будет загружаться полноформатное изображение, а затем браузер будет менять его размер (как это делается, когда в вашем браузере включена настройка автоматического подбора изображения). Атрибуты width и height просто помогают браузеру выяснить, как много места нужно оставить для изображения; и если вы их используете, то они должны соответствовать реальным ширине и высоте изображения.

Исправление HTML-кода для myPod

После того как вы сохранили изображение, можете выйти из программы Photoshop Elements. Теперь все, что вам остается сделать, – внести кое-какие изменения в страницу сайта myPod index.html, чтобы она содержала новую версию фотографии: seattle_med.jpg.

Рассмотрим фрагмент файла index.html, содержащий только те части, которые нужно изменить.

```

<html>
  <head>
    <title>myPod</title>
    <style type="text/css">
      body { background-color: #eaf3da; }
    </style>
  </head>
  <body>
    .
    .
    .
    <h2>Сиэтл, Вашингтон</h2>
    <p>
      Я и мой iPod в Сиэтле! Здесь видны дождевые облака и башня Спейс Нидл.
      И не видно 628 кафе.
    </p>
    <p>
      
    </p>
  </body>
</html>

```

Здесь будет основная часть HTML-кода. Она у вас уже есть в файле index.html.

Все, что вам нужно сделать, – поменять имя файла в элементе `` на имя файла, который вы только что создали: seattle_med.jpg.

А сейчас протестируем...

Итак, вперед! Внесите все изменения, сохраните их и обновите страницу index.html в браузере. Теперь все должно выглядеть намного лучше. Размеры изображения подобраны именно так, чтобы вашим посетителям было удобно на него смотреть.



Теперь изображение отлично помещается в окне браузера.

[далее >](#)

215

КАКОЙ ФОРМАТ ИЗОБРАЖЕНИЯ?

Новое задание для вас: откройте файл chapter5/testimage/eye.jpg в Photoshop Elements. Откройте окно Save for Web (Сохранить для Сети) и заполните пропуски (см. ниже), выбирая каждую настройку качества для JPEG (низкое, среднее, высокое и т. д.). Нужную информацию вы найдете в панели предварительного просмотра, которая находится под самим изображением. Когда справитесь с этим, определите, какая настройка лучше всего подходит для данного изображения.

Формат
Размер изображения.
Время пересылки
при соединении по
телефонной линии.

Формат	Качество	Размер	Время	Победитель
JPEG	<u>Максимальное</u>	_____	_____	<input type="checkbox"/>
JPEG	<u>Очень высокое</u>	_____	_____	<input type="checkbox"/>
JPEG	<u>Высокое</u>	_____	_____	<input type="checkbox"/>
JPEG	<u>Среднее</u>	_____	_____	<input type="checkbox"/>
GIF	<u>Не применяется</u>	_____	_____	<input type="checkbox"/>



Еще больше фотографий для myPod

Для myPod поступила новая партия фотографий: по две из Сиэтла и от друга из Великобритании. Размеры фотографий уже были уменьшены так, что их ширина не превышает 800 пикселов. Добавьте для них элементы `` (вы найдете фотографии в папке `photos`).

```
<html>
  <head>
    <title>myPod</title>
    <style type="text/css">
      body { background-color: #eaf3da; }
    </style>
  </head>
  <body>

    <h1>Добро пожаловать в myPod</h1>
    <p>
      Добро пожаловать туда, где можно похвастаться своим iPod и рассказать, где вы с ним были. Хотите повеселиться вместе с нами? Вам для этого нужен любой iPod: от ранней классической версии до самой современной нановерсии – самого маленького iPod с фотографиями лучшего качества и с цифровой камерой. Просто сфотографируйте свой iPod в любимом месте, и мы будем рады поместить этот снимок на сайте myPod. Итак, чего же вы ждете?
    </p>

    <h2>Сиэтл, Вашингтон</h2>
    <p>
      Я и мой iPod в Сиэтле! Здесь видны дождевые облака и башня Спейс Нидл. И не видно 628 кафе.
    </p>

    <p>
      
      
      
    </p>

    <h2>Бирмингем, Англия</h2>
    <p>
      Это несколько фотографий из Бирмингема. Мы повстречали здесь несколько людей, страстно влюбленных в свои iPod. Взгляните на классические британские красные телефонные будки!
    </p>

    <p>
      
      
    </p>
  </body>
</html>
```

Если хотите, можете добавить сюда пару своих фотографий. Только не забудьте сначала изменить их размер.

Добавьте будем хранить все фотографии из Сиэтла вместе.

То же самое для фотографий из Бирмингема...

Еще один тест для myPod

Подумайте только, какое значение имеют несколько изображений! Страница начинает совершенно по-другому выглядеть.

Итак, у вас есть много изображений на странице, и вы уже изменили их размеры, но они все равно остаются очень большими. Мало того, что страница будет загружаться все медленнее, потому что вы добавляете больше изображений, но и пользователь должен постоянно прокручивать ее, чтобы увидеть все фотографии. Подумайте, не будет ли лучше, если пользователи смогут сначала видеть маленькое изображение — эскиз — каждой фотографии, а затем щелкать на нем кнопкой мыши, чтобы просмотреть большое изображение?

Вот как страница выглядит теперь.

Это страница со всеми изображениями в их полном размере



Доработка сайта таким образом, чтобы использовались эскизы

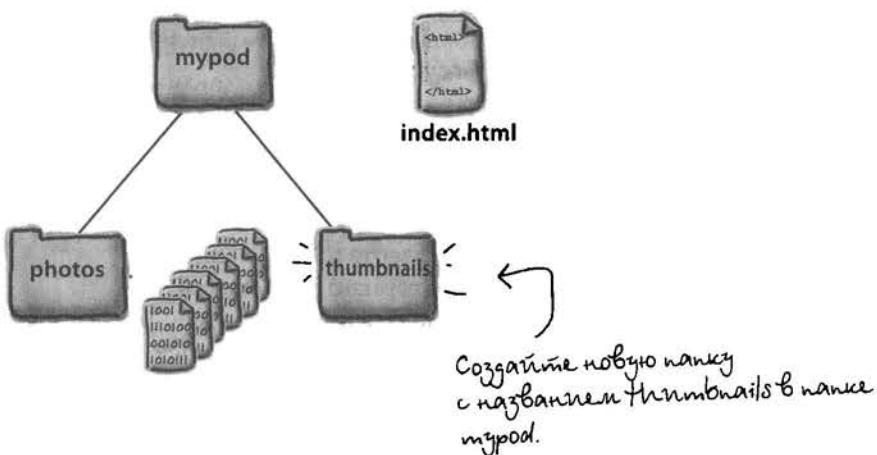
Сейчас вы сделаете вашу страницу более удобной в использовании, заменив имеющиеся фотографии меньшими по размеру (которые мы будем называть эскизами). После этого вы создадите ссылки с каждого эскиза на большую фотографию. Рассмотрим, как это сделать.

- ❶ Создайте новый каталог для малых версий изображений.
- ❷ Измените размер каждой фотографии до 150×100 пикселов и сохраните в папку `thumbnail`.
- ❸ Укажите на эскизы фотографий в атрибутах `src` каждого элемента `` файла `index.html`.
- ❹ Добавьте ссылку с каждого эскиза на новую страницу с большой фотографией.

Создание нового каталога для эскизов

Чтобы содержать файлы в порядке, создайте отдельную папку для уменьшенных версий изображений. В противном случае все закончится тем, что у вас будет папка, в которой в одну кучу свалены и большие фотографии, и их уменьшенные копии. В результате, когда будет добавлено значительное количество фотографий, легко можно будет запутаться в них.

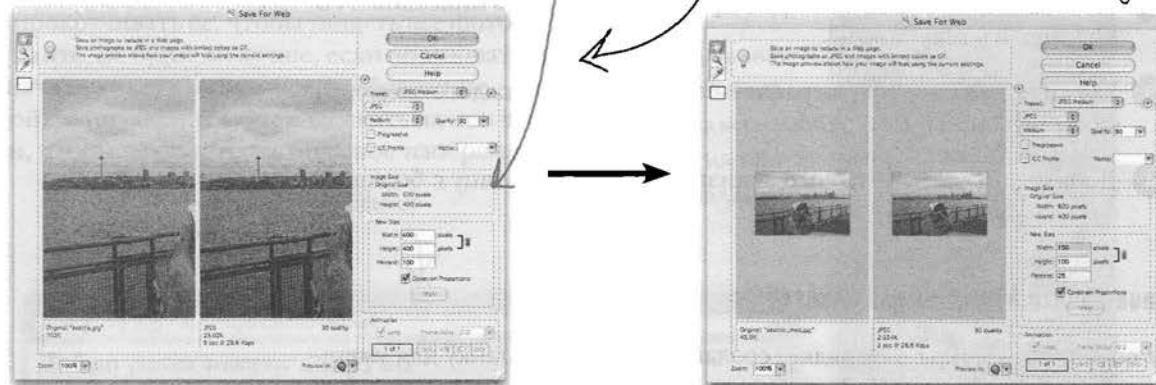
Создайте папку с названием `thumbnails` в папке `typod`. Если вы работаете с файлами, приведенными в качестве примеров к книге, то такая папка там уже есть.



Создание эскизов

Теперь у вас есть место, куда вы будете складывать эскизы, поэтому можно их создать. Начните с открытия файла seattle_med.jpg в программе для редактирования фотографий. Измените размер фотографии до 150×100 пикселов, используя тот же метод, которым уменьшали фотографии до размера 600×400 пикселов.

В Photoshop Elements выберите пункт меню Save for Web (Сохранить для Сети).



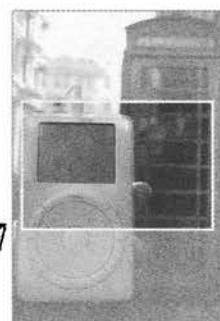
Изменив размер изображения, нажмите OK и сохраните его с таким же именем, но в папке thumbnail. Будьте осторожны: если вы сохраните этот файл в папке photos, то замените им большую фотографию.

Теперь повторите это для каждой фотографии из папки photos.



Хорошее замечание. Поскольку высота изображений больше их ширины, у нас есть два варианта: мы можем поменять размеры местами (и сделать рисунки размером 100×150) или обрезать каждый рисунок и сделать для него малую версию размером 150×100 пикселов. Мы выбираем первый вариант. Если хотите выяснить, как это делается в вашей программе для редактирования фотографий, то можете обрезать рисунки и создать эскизы размером 150×100 пикселов.

Если вы работаете с файлами, приведенными в качестве примеров к книге, то найдите все эскизы фотографий в папке thumbnails и вам не придется всего этого делать (в конце концов, вы читаете HTML, а не занимаетесь обработкой фотографий).



Доработка HTML-кода таким образом, чтобы использовались эскизы

Теперь вам остается лишь поменять HTML-код таким образом, чтобы элементы `` обращались к изображениям из папки `thumbnails`, а не из папки `photos`. Поскольку в коде используются относительные пути, такие как `photos/seattle_med.jpg`, это будет нетрудно. Все, что вам нужно сделать, — заменить в пути папку `photos` папкой `thumbnails` для каждого элемента ``.

```

<html>
  <head>
    <title>myPod</title>
    <style type="text/css">
      body { background-color: #eaf3da; }
    </style>
  </head>
  <body>

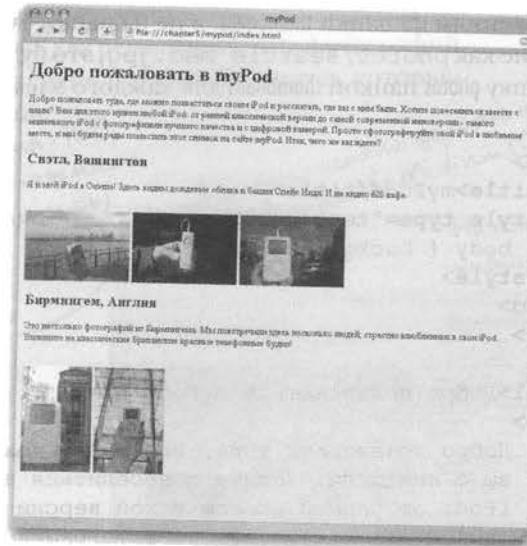
    <h1>Добро пожаловать в myPod</h1>
    <p>
      Добро пожаловать туда, где можно похвастаться своим iPod и рассказать, где
      вы с ним были. Хотите повеселиться вместе с нами? Вам для этого нужен любой
      iPod: от ранней классической версии до самой современной нановерсии —
      самого маленького iPod с фотографиями лучшего качества и с цифровой камерой.
      Просто сфотографируйте свой iPod в любимом месте, и мы будем рады поместить
      этот снимок на сайте myPod. Итак, чего же вы ждете?
    </p>
    <h2>Сиэтл, Вашингтон</h2>
    <p>
      Я и мой iPod в Сиэтле! Здесь видны дождевые облака и башня
      Спейс Нидл. И не видно 628 кафе.
    </p>
    <p>
      
      
      
    </p>
    <h2>Бирмингем, Англия</h2>
    <p>
      Это несколько фотографий из Бирмингема. Мы повстречали здесь несколько
      людей, страстно влюбленных в свои iPod. Взгляните на классические британские
      красные телефонные будки!
    </p>
    <p>
      
      
    </p>
  </body>
</html>

```

Все, что вам нужно сделать, — поменять название папки с `photos` на `thumbnails`.

И снова тест для myPod

Во-о-о-от... намного лучше. Пользователи теперь могут видеть все изображения, имеющиеся в наличии, просто взглянув на страницу. Кроме того, сейчас сразу видно, какая фотография в каком городе была сделана. Теперь нужно найти способ, чтобы создать ссылку с эскизов фотографий на их оригиналы.



Минуточку,
вам не кажется,
что вы пытаетесь нас
одурачить? Раньше рисунки
отображались один под
другим; а теперь — один
за другим.



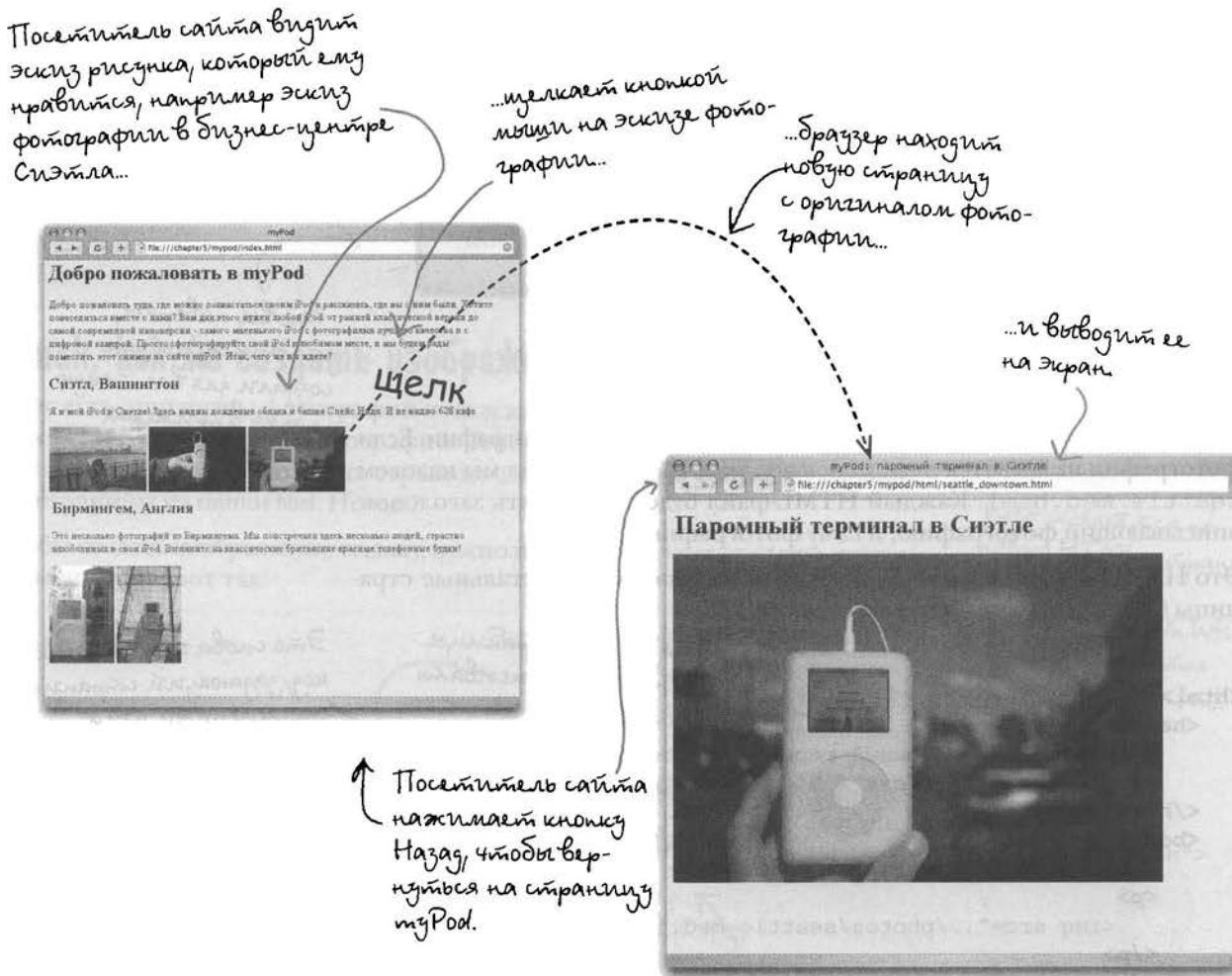
Верно, но элемент `` — это строчный элемент.

Другими словами, мы никого не пытаемся «одурачить». Поскольку элемент `` строчный, то предполагается, что перед ним и после него нет разрыва строки. Итак, если в вашем HTML-коде идет несколько элементов `` подряд, то браузер будет отображать их рисунки в своем окне один за другим в одной строке, если это позволяет ширина окна браузера.

Причина того, почему большие фотографии не отображались одна за другую, состоит в том, что в окне браузера для этого не хватало места. Вместо этого они отображались одна под другой. Браузер всегда оставляет свободное место перед блочным элементом и после него, а если вы снова посмотрите в окно браузера, то увидите, что фотографии расположены непосредственно одна под другой, без отступов между ними. Это еще один признак того, что `` — строчный элемент.

Превращение эскизов в ссылки

Вы уже почти все сделали. Теперь осталось только создать ссылки с каждого эскиза на его оригинал. Вот как это будет выглядеть.



Вам нужно сделать две вещи.

- 1 Страницы с изображением каждой фотографии и с названием, которое описывает ее содержание.**
- 2 Ссылки с каждого эскиза на странице index.html на соответствующие оригиналы фотографий.**

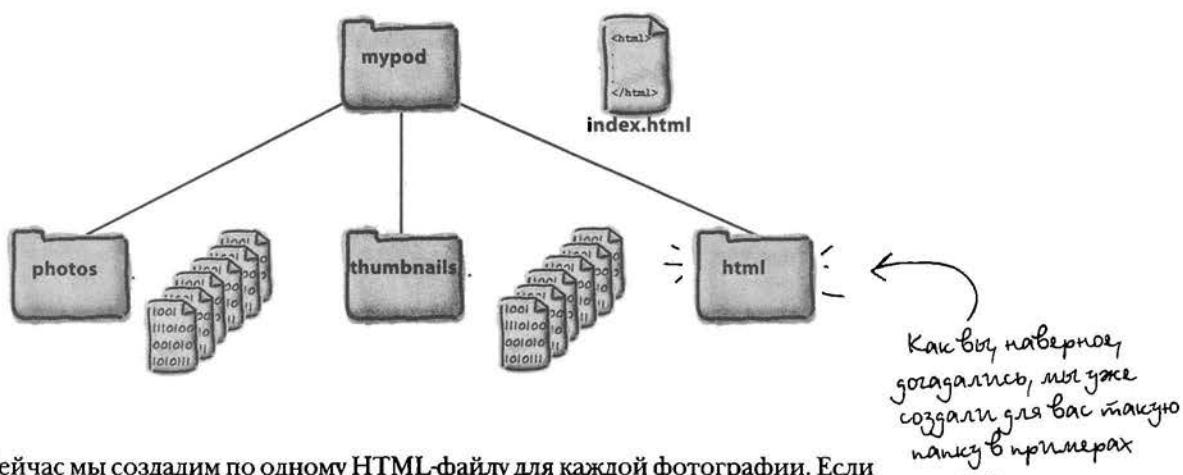
Сначала создадим страницы, а затем вернемся и сделаем изображения-ссылки.

[далее ▶](#)

223

Создание индивидуальных страниц для фотографий

Сначала создайте новую папку html в папке myPod для хранения этих индивидуальных страниц.



Сейчас мы создадим по одному HTML-файлу для каждой фотографии. Если фотография называется `seattle_med.jpg`, то HTML-файл мы назовем `seattle_med.html`. Каждый HTML-файл будет содержать заголовок, описывающий фотографию, и саму фотографию.

Это HTML-код для первой фотографии из Сиэтла. Все остальные страницы будут иметь аналогичную структуру:

```
<html>
  <head>
    <title>myPod: паромный терминал в Сиэтле</title>
    <style type="text/css"> body { background-color: #eaf3da; } </style>
  </head>
  <body>
    <h1>Паромный терминал в Сиэтле</h1>
    <p>
      
    </p>
  </body>
</html>
```

Название страницы. Оно будет описывать фотографию.

Это снова готовый CSS-код, задающий странице определенный цвет.

Здесь мы даем странице описательный заголовок.

Это элемент `img`, который указывает на оригинал фотографии `seattle-med.jpg`. Добавим в элемент `img` описательный атрибут `alt`.

Обратите внимание, что нам нужно использовать символы «..» в относительном пути, потому что папка `html` — это «сестра» папки `photos`, значит, сначала нужно подняться на одну папку вверх, а затем опуститься в папку `photos`.



Упражнение

Если вы посмотрите в папку html в приложении с примерами файлов к этой главе, то найдете там все индивидуальные страницы для фотографий, кроме страницы для seattle_downtown.jpg. Создайте страницу с названием seattle_downtown.html в папке html и протестируйте ее. Перед тем как продолжать дальше, проверьте, чтобы она работала. Если у вас будут какие-то проблемы с выполнением этого упражнения, то можете посмотреть правильное решение в конце этой главы.

Как же создать изображения-ссылки?

У вас есть оригиналы фотографий и их эскизы, а также набор индивидуальных HTML-страниц для фотографий. Теперь вам нужно все это соединить и создать ссылки с эскизов на странице index.html на соответствующие им страницы из папки html. Но как?

Чтобы создать изображение-ссылку, нужно поместить элемент `` внутрь элемента `<a>`, вот так:

Это элемент `(img)` для эскиза seattle-downtown.jpg, как он определен в файле index.html.

```

<a href="html/seattle_downtown.html">
  
</a>

```

Это открывающий тег `(a)` непосредственно перед элементом `(img)`.

Элемент `(img)` полностью вложен в элемент `(a)`.

`href` ссылается на новую HTML-страницу с фотографией seattle-downtown.html, которая находится в директории html.

Как только вы поместили элемент `` в элемент `<a>`, браузер начинает воспринимать изображение как ссылку, на которой можно щелкнуть. Когда вы щелкаете кнопкой мыши на таком изображении, браузер извлекает страницу, указанную в значении атрибута `href`.

Добавление изображений-ссылок в файл index.html

Это последний шаг. Вам осталось вложить все элементы **** для эскизов в файле index.html в элементы **<a>**. Помните, что атрибут href каждого элемента **<a>** должен ссылаться на страницу с оригиналом фотографии из папки **html**. Убедитесь, что все ваши ссылки, эскизы и страницы с оригиналами фотографий соответствуют друг другу.

Здесь мы приводим весь файл index.html. Все, что вам нужно сделать, — добавить HTML-код, выделенный серым цветом.

```
<html>
  <head>
    <title>myPod</title>
    <style type="text/css">
      body { background-color: #eaf3da; }
    </style>
  </head>
  <body>

    <h1>Добро пожаловать в myPod</h1>
    <p>
      Добро пожаловать туда, где можно похвастаться своим iPod и рассказать, где
      вы с ним были. Хотите повеселиться вместе с нами? Вам для этого нужен любой
      iPod: от ранней классической версии до самой современной нановерсии — самого
      маленького iPod с фотографиями лучшего качества и с цифровой камерой. Просто
      сфотографируйте свой iPod в любимом месте, и мы будем рады поместить этот снимок
      на сайте myPod. Итак, чего же вы ждете?
    </p>

    <h2>Сиэтл, Вашингтон</h2>
    <p>
      Я и мой iPod в Сиэтле! Здесь видны дождевые облака и башня Спейс Нидл.
      И не видно 628 кафе.
    </p>

    <p>
      <a href="html/seattle_med.html">
        
      </a>
      <a href="html/seattle_shuffle.html">
        
      </a>
      <a href="html/seattle_downtown.html">
        
      </a>
    </p>

    <h2>Бирмингем, Англия</h2>
    <p>
```

Это несколько фотографий из Бирмингема. Мы повстречали здесь несколько людей, страстно влюбленных в свои iPod. Взгляните на классические британские красные телефонные будки!

```
</p>
<p>
<a href="html/britain.html">
    
</a>
<a href="html/applestore.html">
    
</a>
</p>
</body>
</html>
```

Каждый эскиз вложен в элемент <a>. Только будьте аккуратны и используйте правильное значение атрибута href для каждой ссылки!

Добавьте эти элементы `<a>` в свой файл `index.html`. Сохраните его, обновите в браузере и проверьте, как работает myPod!

Часто
Задаваемые
Вопросы

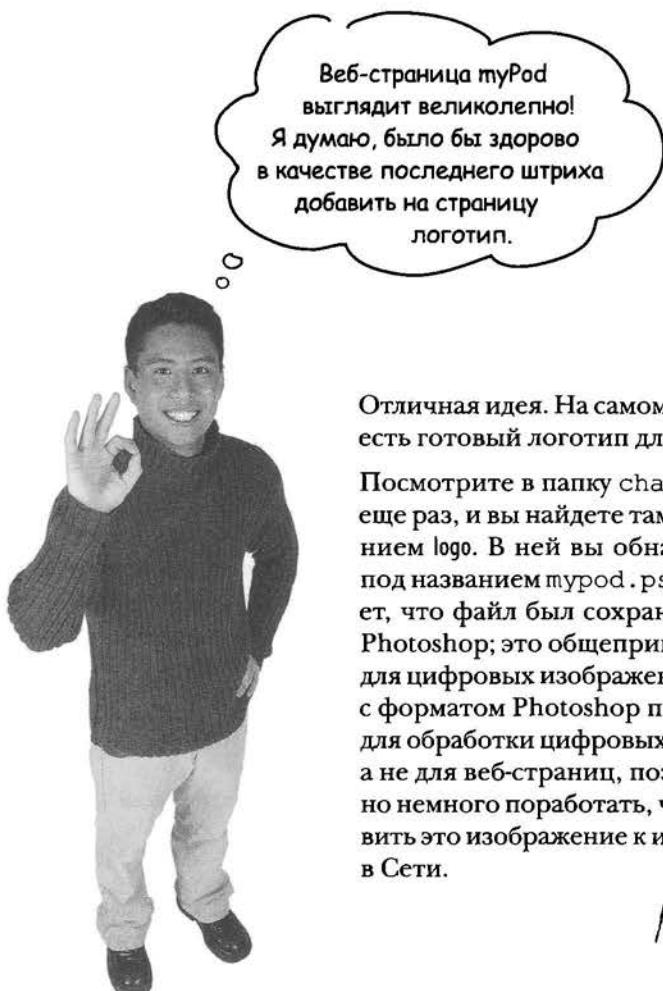
В: Если внутри элемента `<a>` находится текст, то на странице он отображается подчеркнутым. Почему нет ничего подобного для изображений?

О: На самом деле большинство браузеров выделяют изображение границей вокруг него, чтобы показать, что оно является ссылкой (наш браузер Safari — один из тех, кто этого не делает). Если браузер рисует границу вокруг изображений-ссылок, а вам это не нравится, продолжайте читать книгу, и через несколько глав мы расскажем, как убрать эту границу с помощью CSS. Обратите также внимание, что, когда вы наводите указатель мыши на изображение-ссылку, вид указателя меняется так, чтобы подсказать вам, что на рисунке можно щелкнуть кнопкой мыши. В большинстве случаев пользователям понятно,

что изображение является ссылкой, из контекста или по виду указателя мыши, даже если вокруг него нет границы.

В: Нельзя ли создать ссылки непосредственно на JPEG-изображения без всех этих HTML-страниц? Я думал, что браузеры достаточно умны для того, чтобы отображать рисунки просто так.

О: Вы правы, можно создать ссылку непосредственно на изображение: ` ... `. Если вы так сделаете, а потом воспользуетесь такой ссылкой, то браузер сам выведет рисунок в пустом окне. Но вообще считается плохой манерой создавать ссылки непосредственно на изображения, потому что обычно рисунку дают какое-либо описание.



Веб-страница myPod
выглядит великолепно!
Я думаю, было бы здорово
в качестве последнего штриха
добавить на страницу
логотип.

Отличная идея. На самом деле у нас уже есть готовый логотип для myPod.

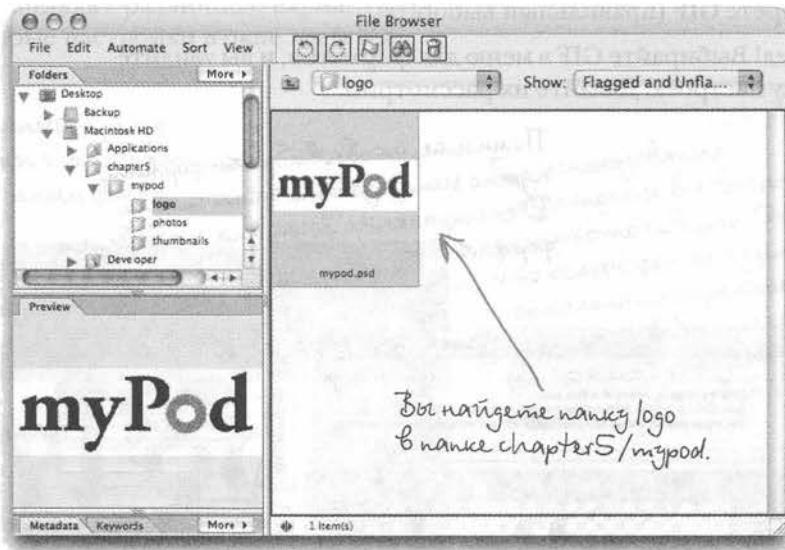
Посмотрите в папку chapter5/myPod еще раз, и вы найдете там папку с названием *logo*. В ней вы обнаружите файл под названием *myPod.psd*. PSD означает, что файл был сохранен в формате Photoshop; это общепринятый формат для цифровых изображений. Но файлы с форматом Photoshop предназначены для обработки цифровых изображений, а не для веб-страниц, поэтому нам нужно немного поработать, чтобы подготовить это изображение к использованию в Сети.

Многие программы для редактирования фотографий работают с файлами формата PSD, поэтому, даже если у вас нет Photoshop Elements, продолжайте читать дальше и делайте все аналогично в своем редакторе. Если же в вашей программе нельзя открыть PSD-файл, вы можете воспользоваться готовыми логотипами из папки *logo*.

Открытие логотипа myPod

Давайте поработаем с логотипом myPod: откройте файл mypod.psd из папки chapter5/mypod/logo в программе Photoshop Elements.

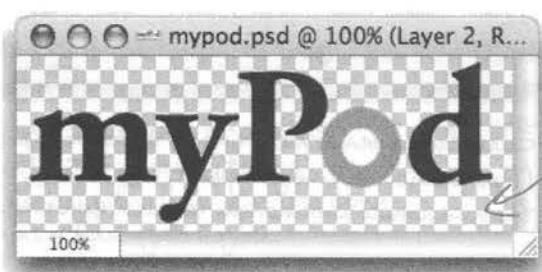
Если в вашей программе для редактирования фотографий файл не открывается, все равно работайте с другим форматом по аналогии.



Подробнее...

Логотип сделан удачно; в нем текст сочетается с двумя кругами: серым и белым (что явно ассоциируется с функцией «click wheel» iPod).

Но что это за узор в клеточку на заднем плане? Это способ, которым большинство программ для редактирования изображений отображают прозрачные области. Вам нужно это помнить, когда будете выбирать графический формат для логотипа...



[далее ↗](#)

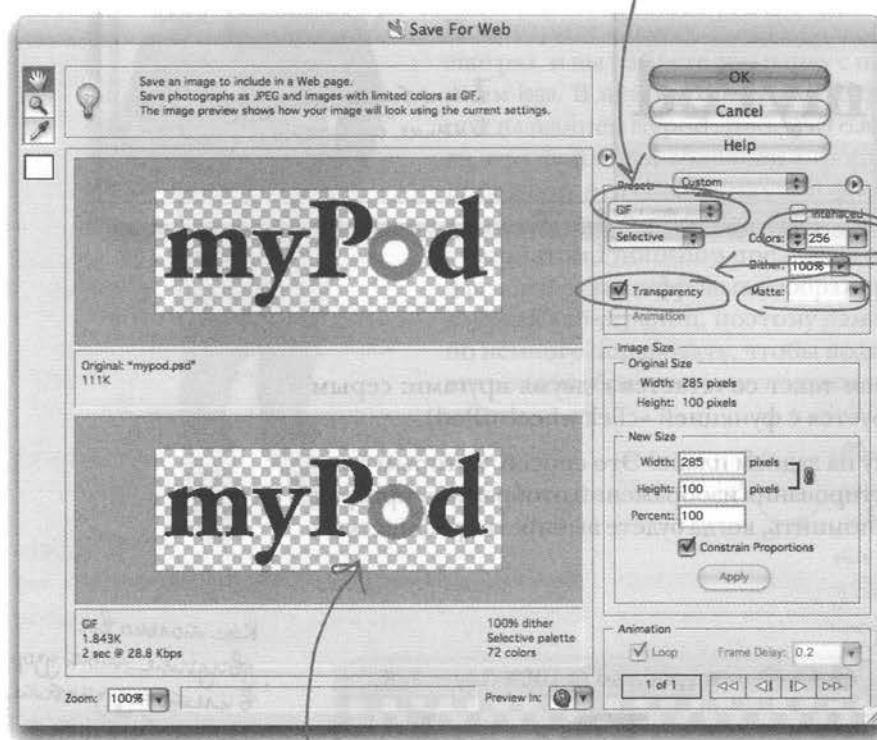
229

Какой формат использовать?

Вы уже знаете, что есть несколько вариантов сохранения изображения: можно выбрать формат JPEG или GIF. В этом логотипе используется всего три цвета, текст и несколько геометрических фигур. Руководствуясь знаниями, которые у вас уже есть, вы, скорее всего, выберете GIF (правильный выбор!).

Итак, вперед! Выбирайте GIF в меню для форматов, и вы увидите еще парочку настроек. Давайте их рассмотрим...

Помните, чтобы выбрать формат, нужно использовать этот список. Для сохранения логотипа мы выберем формат GIF.



Здесь Photoshop Elements показывает вам количество цветов, использованное при сохранении рисунка в формате GIF. Уже установлено максимальное значение – 256. Мы не будем его менять.

Когда вы выбираете формат GIF, появляется флагок Transparency (Прозрачность). По умолчанию он установлен. Нам нужен прозрачный задний план?

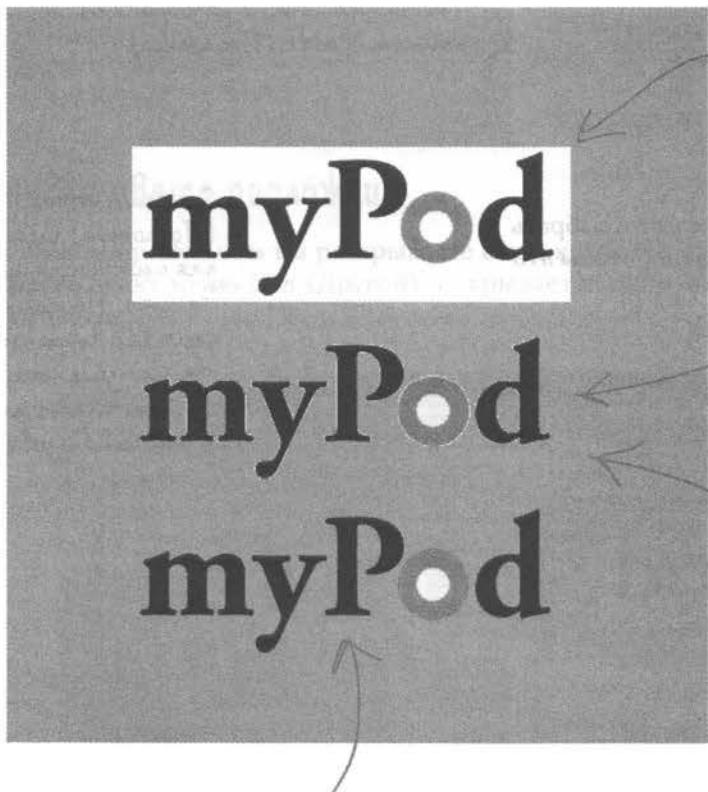
Обратите также внимание на список Matte (Подложка). Он тоже имеет отношение к прозрачности, как вы совсем скоро поймете.

Если вы попытаетесь снять флагок Transparency (Прозрачность), то увидите, что в GIF для предварительного просмотра задний фон стал белым.

Использовать прозрачность или нет?

Логотип myPod будет помещен на светло-зеленый фон. Итак, вам уже нравится идея использовать прозрачность? Хорошо, давайте сравним, как выглядят различные варианты логотипа, используя некоторые настройки в окне Save for Web (Сохранить для Сети).

Это логотип, сохраненный тем самым различными способами и отображенный на веб-странице с зеленым фоном.



О, теперь все выглядит просто здорово. На этот раз мы скажем Photoshop Elements создать подложку вокруг текста с использованием зеленого фона. Как? Покажем вам чуть позже.

Без использования прозрачности все выглядит достаточно исквернно. Очевидно, что белый фон не будет нормально смотреться на зеленой веб-странице (он может выглядеть хорошо только на белой веб-странице).

А это то, что мы получаем, если устанавливаем флагок Transparency (Прозрачность) и сохраняем. Уже лучше, но что это за белый «ореол» вокруг букв в логотипе?

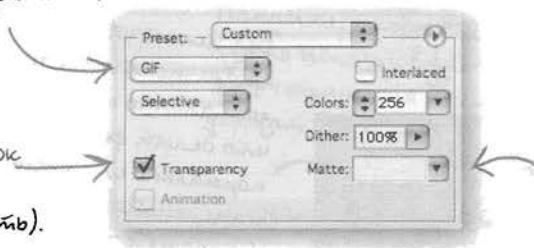
Эффект «ореола» возникает, потому что программа для редактирования фотографий создает подложку, смягчая края текста на выбранном цвете фона. Делая это для нашего логотипа, программа «предполагала», что смягчает края на белом фоне.

Сохранение прозрачного GIF-изображения

Вы знаете, что вам нужна прозрачная версия логотипа в формате GIF, вы также знаете, что нужно использовать подложку, чтобы избежать «ореолов» вокруг текста. Давайте посмотрим на область с настройками GIF в окне Save for Web (Сохранить для Сети).

Вы уже знаете что
нужно выбрать GIF.

И установ-
ить флагок
Transparency
(Прозрачность).

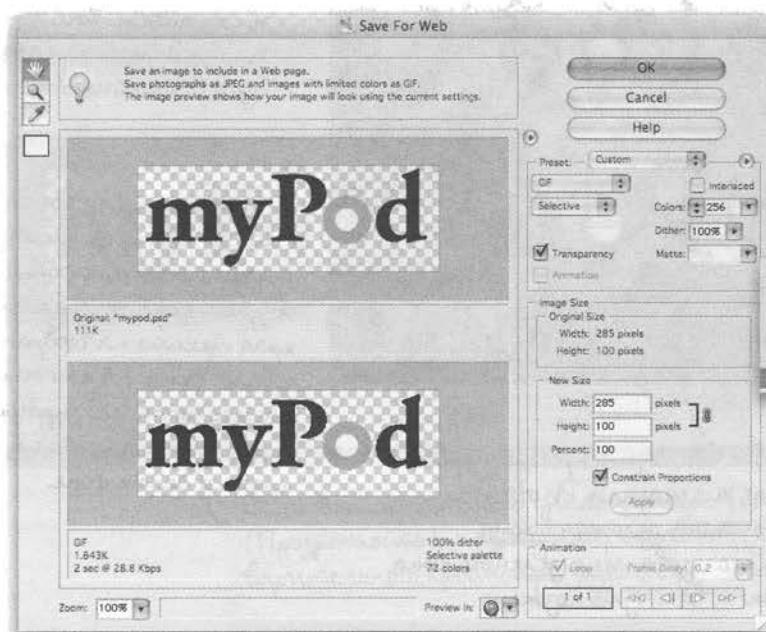


Сейчас нам нужно разобраться со
списком Matte (Подложка).

С помощью списка Matte (Подложка) вы можете выбирать цвет подложки вокруг текста. Желательно, чтобы это был цвет фона веб-страницы.

Список Matte

(Подложка) задает цвет
для смягчения краев
текста. Поскольку цвет
фона веб-страницы
светло-зеленый, мы
хотим, чтобы этот же
цвет использовался для
подложки.



Выберите
пункт Other
(Другой), так
как нуж-
ного цвета
в списке нет.

Минуточку, а как узнать цвет фона Веб-страницы?

Помните тот готовый CSS-код в файле index.html? Он устанавливает светло-зеленый цвет для фона страницы. И вот откуда мы можем взять значение цвета:

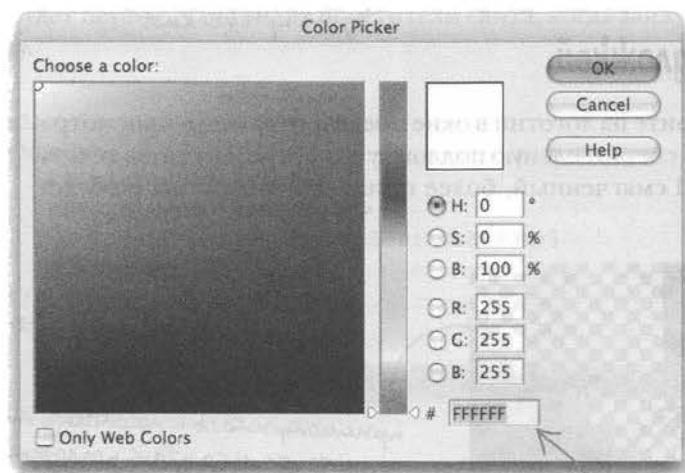
```
<style type="text/css">
  body { background-color: #eaf3da; }
</style>
```

Это и есть цвет фона.

Что? Вы бы никогда не сказали, что это светло-зеленый? Пока поверите нам на слово, а через несколько глав мы к этому вернемся и подробно расскажем вам о цветах.

Установка цвета подложки

Когда в Photoshop Elements вы раскрываете список Matte (Подложка) и выбираете пункт меню Other (Другой), открывается окно Color Picker (Палитра цветов).

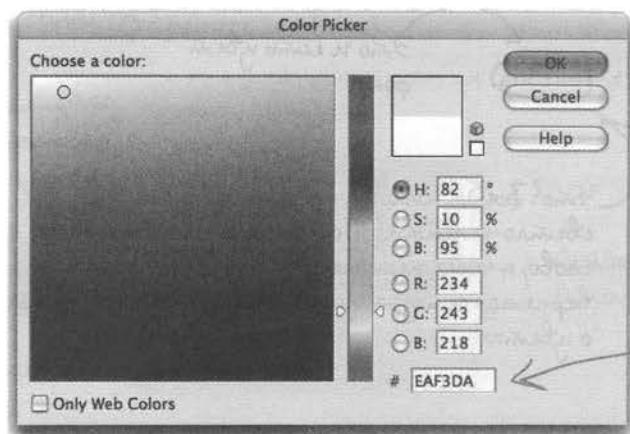


Вот еще множество способов выбрать цвет подложки. Мы просто хотели выбрать цвет фона страницы, уже зная это значение — #eaf3da...

... и он должен быть установлен здесь.

Установка цвета подложки, продолжение

Итак, продолжим! Задайте значение цвета eaf3da в окне Color Picker (Палитра цветов). Вы увидите, что он стал таким же, как цвет фона страницы myPod.

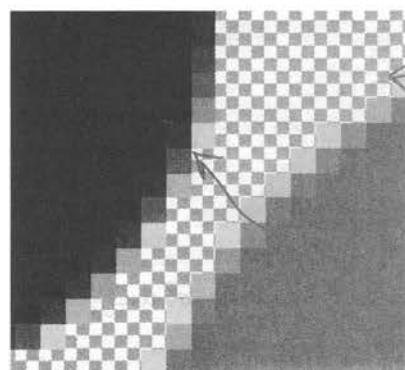


Введите эти буквы прямо сюда. Это поле предназначено специально для указания цвета в веб-формате. Можно вводить либо прописные, либо строчные буквы.

После того как вы ввели цвет в окне Color Picker (Палитра цветов), нажмите кнопку OK и изменения в логотипе будут применены.

Рассмотрим логотип с подложкой

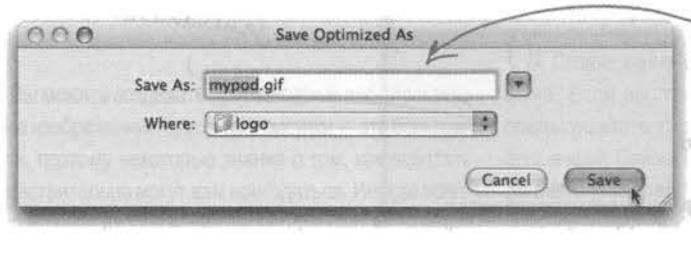
Теперь еще раз внимательно посмотрите на логотип в окне предварительного просмотра. Вы увидите, что программа добавила светло-зеленую подложку вокруг резких углов текста. Это придаст тексту логотипа myPod смягченный, более приятный вид, когда он будет помещен на веб-страницу.



Если сейчас вы внимательно посмотрите на логотип, то заметите, что цвет подложки совпадает с цветом фона веб-страницы mypod.html.

Сохранение логотипа

Отлично, вы задали все необходимые настройки в окне Save for Web (Сохранить для Сети), теперь нажмите кнопку OK, чтобы сохранить изображение с именем mypod.gif.



Программа автоматически изменяет расширение в имени файла на GIF. Сохраните изображение под названием mypod.gif в папку logo.

Добавление логотипа на Веб-страницу myPod

Теперь все, что вам осталось сделать, — добавить логотип на веб-страницу myPod. Мы поместим его в самом верху страницы, чтобы он отображался над описанием сайта и фотографиями iPod. В таком случае это будет первое, что увидят посетители сайта, когда зайдут на страницу.

```
<html>
  <head>
    <title>myPod</title>
    <style type="text/css">
      body { background-color: #eaf3da; }
    </style>
  </head>
  <body>
    <p>
      
    </p>

    <h1>Добро пожаловать в myPod</h1>
    .
    .
  </body>
</html>
```

Добавьте изображение логотипа на самый верх веб-страницы myPod. Укаживайте правильный относительный путь к логотипу, находящемуся в папке logo, и используйте атрибут alt, который опишет изображение.

Здесь будет оставаться часть HTML-кода из файла index.html.

А теперь последний тест

Давайте протестируем эту страницу! Обновите ее в браузере и посмотрите, как выглядит myPod-логотип в формате GIF с использованием прозрачности.



Затем труды сполна
осуществились. Теперь на
бес-странную myPod есть
замечательный логотип.

Отличная
работа. Логотип
выглядит великолепно.
У вас получился
сногшибательный сайт
myPod!



часто Задаваемые Вопросы

В: Обязательно ли мне все это знать о форматах изображений, чтобы создавать хорошие веб-страницы?

О: Нет. Вы можете создавать превосходные веб-страницы вообще без изображений. Но все-таки рисунки — это большая часть Сети, поэтому некоторые знания о том, как работать с ними, действительно могут вам пригодиться. Иногда всего лишь одно или два изображения переводят сайт из разряда хороших в отличные. Есть еще много всего, что можно узнать об изображениях, главное, только захотеть.

В: Зачем вообще сглаживать края текста?

О: Сравните два варианта логотипа для myPod:



Вы видите, что у логотипа, расположенного выше, очень резкие, зубчатые границы текста, что ухудшает его читаемость. Так текст отображается на экране компьютера по умолчанию. Во второй версии логотипа края улучшены благодаря использованию метода, называемого сглаживанием. На экране компьютера текст, обработанный таким методом, лучше читается и более приятен для глаз.

В: Когда вступает в действие подложка?

О: Сглаживание смягчает границы относительно цвета фона. Если вы поместите нижнюю версию логотипа (из предыдущего вопроса) на цветной фон, то увидите в нем белые края. Список Matte (Подложка) в программе Photoshop Elements позволяет установить цвет фона, на который будет помещен текст, поэтому края текста смягчаются с учетом этого цвета.

В: Этот метод работает только для текста?

О: Нет, он работает для любых линий в вашей графике, на которых могут появляться «зазубрины». Кругу в логотипе myPod тоже был применен этот метод.

В: Почему нельзя просто сделать цвет фона в логотипе чистым и совпадающим с цветом фона веб-страницы?

О: Вы можете сделать и так, но в этом есть один недостаток: если на вашей веб-странице имеются другие элементы, которые будут видны через прозрачные области логотипа, то они не будут отображаться в чистом цвете. Пока вы не видели подобных примеров, но еще увидите, когда будете изучать CSS.

В: Что, если я поменяю цвет фона после того, как создам версию логотипа с использованием подложки?

О: Скорее всего, незначительное изменение цвета фона не будет заметно. Однако, если вы значительно поменяете цвет фона страницы, вам придется исправить GIF, используя новый цвет подложки.

ПОВТОРИМ выученное

- Используйте элемент ``, чтобы поместить изображение на веб-страницу.
- Браузеры обращаются с элементами `` немного иначе, чем с другими элементами HTML. После прочтения HTML-страницы браузер извлекает каждое изображение с веб-сервера и отображает его.
- Если у вас много больших картинок на веб-странице, то вы можете сделать ее более удобной в использовании и уменьшить время ее загрузки. Для этого создайте эскизы изображений, на которых пользователь сможет щелкнуть, чтобы посмотреть оригиналы изображений.
- Элемент `` — строчный элемент. Это означает, что браузер не добавляет разрывы строки перед изображениями и после них.
- С помощью атрибута `src` вы указываете месторасположение файла с рисунком. Вы можете брать изображения со своего сайта, используя относительные пути в атрибуте `src`, или изображения с других сайтов, используя URL-адрес.
- Атрибут `alt` элемента `` позволяет задать содержательное описание изображения. Оно выводится в некоторых браузерах, если само изображение не может быть найдено, а также используется в экранных дикторах, чтобы можно было описать изображение людям со слабым зрением.
- Лучше всего на веб-страницах применять изображения шириной не больше 800 пикселов. Многие фотографии, созданные цифровыми камерами, имеют ширину, большую, чем у веб-страниц, поэтому необходимо менять их размер.
- Photoshop Elements — это одна из множества программ для редактирования фотографий, которую вы можете использовать, чтобы поменять размер изображения.
- Слишком большие изображения для веб-страниц создают неудобства в использовании этих страниц и увеличивают время их загрузки и отображения.
- Пиксель — самая маленькая точка, которая может быть изображена на экране. Каждый рисунок состоит из множества тысяч пикселов. В зависимости от монитора количество пикселов на дюйм колеблется от 72 до 120.
- JPEG и GIF — это два формата изображений, широко используемые в браузерах.
- Формат JPEG лучше всего подходит для фотографий и других сложных изображений.
- Формат GIF больше всего подходит для логотипов и другой простой графики с чистыми цветами, линиями или текстом.
- Изображения в формате JPEG могут быть скжаты до различного уровня качества, и вы можете выбрать наилучшее для себя соотношение размера и качества.
- Формат GIF позволяет создавать изображения с прозрачным фоном. Если вы поместите такое изображение на веб-страницу, то все, что окажется под ним, в том числе и фон самой веб-страницы, будет видно сквозь прозрачные части изображения.
- В Photoshop Elements для выбора нужного цвета при смягчении краев в вашем прозрачном GIF-изображении используйте список Matte (Подложка) из окна Save for Web (Сохранить для Сети).
- Изображения могут быть использованы в качестве ссылок на другие веб-страницы. Чтобы создать изображение-ссылку, используйте элемент ``, вложив его в элемент `<a>`, и поместите ссылку в атрибут `href` элемента `<a>`.