

# 11 Форматирование таблиц и веб-форм

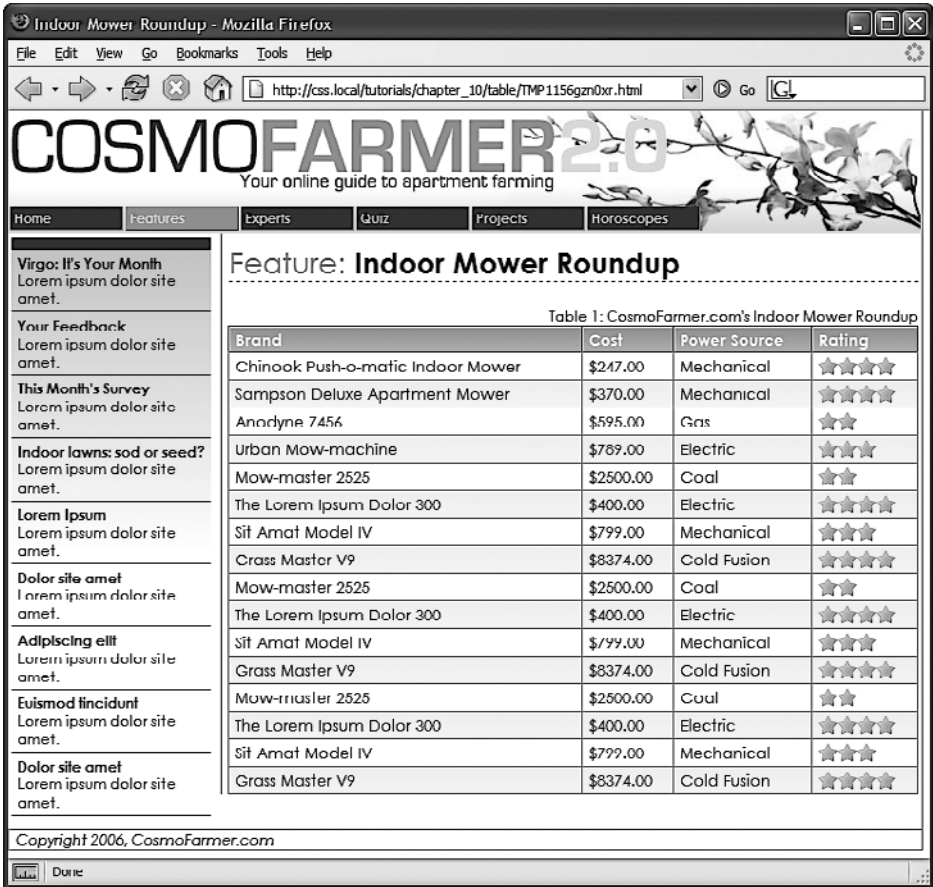
Возможности каскадных таблиц стилей не ограничиваются форматированием текста, изображений и ссылок. Вы можете создавать информационные таблицы с расписаниями, результатами спортивных мероприятий и списки воспроизведения с музыкой, которые становятся намного более читабельными, если к ним добавить границы, фоновые рисунки и другие визуальные улучшения. Ко всему прочему, вы можете применять каскадные таблицы стилей для разработки соответствующих элементов веб-форм, которые помогут вашим посетителям сделать заказ, подписаться на рассылку или воспользоваться вашими самыми новыми веб-приложениями. В этой главе вы узнаете, как вывести на экран таблицы и веб-формы с помощью HTML, а также как их скомпоновать и применить к ним соответствующее форматирование посредством каскадных таблиц стилей. В практикуме в конце главы вы создадите таблицу и веб-форму, используя приемы, которым попутно обучитесь.

## Разумное применение таблиц

За короткую историю существования Всемирной паутины HTML-таблицы получили очень широкое распространение. Изначально созданные для отображения данных в табличном виде, они стали очень популярным инструментом для компоновки макетов. Столкнувшись с некоторыми ограничениями в HTML, дизайнеры творчески подошли к решению этой проблемы и начали применять столбцы и строки таблиц для размещения в них таких элементов страниц, как заголовки и боковые панели. Как вы увидите в части III этой книги, каскадные таблицы стилей намного лучше справляются с компоновкой веб-страниц. Вы можете сконцентрировать все ваше внимание на использовании таблиц по их прямому назначению — для отображения данных (рис. 11.1).

HTML (и XHTML) имеет в своем распоряжении внушительное количество элементов, предназначенных для создания таблиц. С помощью этого фрагмента HTML-кода создается очень простая таблица, которую вы можете видеть на рис. 11.2.

```
<table border="1">
<caption align="bottom">
  Table 1: CosmoFarmer.com's Indoor Mower Roundup
</caption>
```



**Рис. 11.1.** Благодаря CSS в этой таблице, содержащей информацию о газонокосилках, шрифты изменяются на более привлекательные, создаются границы и меняются фоновые цвета, но вся основная структура создана с помощью HTML-кода

```
<colgroup>
  <col class="brand" />
  <col class="price" />
  <col class="power" />
</colgroup>
<thead>
<tr>
  <th scope="col">Brand</th>
  <th scope="col">Price</th>
  <th scope="col">Power Source</th>
</tr>
</thead>
<tbody>
<tr>
  <td>Chinook Push-o-matic Indoor Mower</td>
```

```
<td>$247.00</td>
<td>Mechanical</td>
</tr>
<tr>
<td>Sampson Deluxe Apartment Mower</td>
<td>$370.00</td>
<td>Mechanical</td>
</tr>
</tbody>
</table>
```

Даже имея в своем распоряжении лишь по три строки и столбца, таблица использует девять уникальных HTML-элементов: table, caption, colgroup, col, thead, tbody, tr, th и td. Вообще-то, чем меньше HTML-кода, тем лучше, и все эти элементы вам не нужны: вполне можно обойтись только следующими: table, tr и td (и в большинстве случаев еще th). Но различные табличные элементы предоставляют множество полезных зацепок для привязки к ним каскадных таблиц стилей. Заголовок для каждого из столбцов таблицы, задаваемый элементом th, может иметь различный вид в разных ячейках таблицы, если вы будете использовать при создании стиля элемент th, а элемент colgroup можно применять в качестве средства управления шириной столбца таблицы. Это избавит вас от нудной работы по созданию множества различных классов, таких как .tableHeader, а также от необходимости вручную задавать их для каждой ячейки таблицы. В следующем разделе вы познакомитесь с примерами использования различных элементов и ощутите все их преимущества.

Brand	Price	Power Source	Mini-Review	<ht>
Chinook Push-o-matic Indoor Mower	\$247.00	Mechanical	The latest model of the Chinook mower is a big improvement over last year's model. It's smooth gliding action is perfect for even massively over grown sod. Its handling around corners is superb -- perfect for those tight areas around sofas and coffee tables.	<td> граница
Sampson Deluxe Apartment Mower	\$370.00	Mechanical	In our battery of 7 mowing tests, the Sampson scored 9 or above on each. The fine blades turn even large weeds into tiny cuttings, perfect for composting or salad garnishes.	отступ

Table 1: CosmoFarmer.com's Indoor Mower Roundup

**Рис. 11.2.** Заголовки таблицы с данными, как эта, обычно создаются с помощью элемента th. Заголовки ячеек определяют тип информации, отображаемой в строках и столбцах. Актуальные данные хранятся в таблице в элементах td

# Форматирование таблиц

Вы можете использовать многие свойства каскадных таблиц стилей, о которых уже прочитали, чтобы добавить привлекательности таблицам и их контенту. Свойство

color, например, устанавливает цвет текста в таблице. Тем не менее вы обнаружите некоторые свойства, особо полезные при использовании их в таблицах, а также свойства, предназначенные исключительно для форматирования таблиц. Тот факт, что таблица состоит из нескольких HTML-элементов, помогает определить, к какому из них применить соответствующее свойство CSS. Отступы элемента table никак не влияют на отображение. В следующих нескольких разделах рассматриваются свойства CSS, используемые для форматирования таблиц, а также HTML-элементы, к которым они применяются.

## Добавление отступов

Как вы прочитали в разделе «Управление размерами полей и отступов» главы 7, отступ — это расстояние между границей элемента и его содержимым. Вы можете использовать отступы, чтобы допустить немного свободного места между краями абзацев с текстом и их границами. Когда речь идет о таблицах, границы — это *края* ячейки, и благодаря отступам добавляется свободное пространство вокруг содержимого этой ячейки таблицы (см. рис. 11.2). С их помощью можно управлять расстоянием между контентом конкретной ячейки и ее границами.

Отступы применяются либо к заголовкам таблицы, либо к ее ячейкам, но никак не к самому элементу table. Итак, чтобы задать отступ размером 10 пикселей для всех ячеек таблицы, вам следует воспользоваться следующим стилем:

```
td, th { padding: 10px; }
```

Вы можете также контролировать отступы от каждого из четырех краев ячейки. Чтобы добавить 10 пикселей пространства сверху для каждой ячейки с данными в таблице, 3 пикселя снизу и по 5 пикселей слева и справа, создайте такой стиль:

```
td {  
  padding-top: 10px;  
  padding-right: 5px;  
  padding-bottom: 3px;  
  padding-left: 5px;  
}
```

Или используйте сокращенную запись свойства padding:

```
td {  
  padding: 10px 5px 3px 5px;  
}
```

## Настройка горизонтального и вертикального выравнивания

Чтобы настроить расположение содержимого *внутри* самой ячейки, используйте свойства text-align и vertical-align.

Первое свойство применяется для управления горизонтальным выравниванием и может принимать значения left, right, center и justify (рис. 11.3). Это

наследуемое свойство (см. главу 4, чтобы узнать больше о наследовании). Если вы хотите выровнять содержимое всех ячеек таблицы по правому краю, создайте следующий стиль:

```
table { text-align: right; }
```

Это свойство удобно применять к элементам `th`, так как браузеры обычно выравнивают его по центру. Стиль вида `th { text-align: left; }` выравнивает заголовки с ячейками таблицы.

left	center	right	justified
Brand	Price	Power Source	Mini-Review
Chinook Push-o-matic Indoor Mower	\$247.00	Mechanical	The latest model of the Chinook mower is a big improvement over last year's model. It's smooth gliding action is perfect for even massively over grown sod. Its handling around corners is superb -- perfect for those tight areas around sofas and coffee tables.
Sampson Deluxe Apartment Mower	\$370.00	Mechanical	In our battery of 7 mowing tests, the Sampson scored 9 or above on each. The fine blades turn even large weeds into tiny cuttings, perfect for composting or salad garnishes.

Table 1: CosmoFarmer.com's Indoor Mower Roundup

Рис. 11.3. С помощью свойства `text-align` можно указать край ячейки, по которому будет выровнено ее содержимое

У ячеек в таблицах есть также такой параметр, как высота. Обычно браузеры выравнивают содержимое вертикально по центру ячейки (см. значение `middle` на рис. 11.4). Вы можете изменить это с помощью свойства `vertical-align`. Используйте одно из этих четырех значений: `top`, `baseline`, `middle` или `bottom`. Значение `top` помещает содержимое ячейки вверх, `middle` — по центру, а `bottom` — внизу. При указании значения `baseline` выравнивание происходит так же, как и при установке значения `top`, за исключением того, что браузер выравнивает первую строку текста в каждой ячейке заданной строки таблицы (рис. 11.4). Скорее всего, вы даже не заметите тонкостей работы значения `baseline`. В отличие от `text-align` свойство `vertical-align` не наследуется, поэтому вы можете использовать его только в стилях, которые применяются напрямую к элементам `th` и `td`.

СОВЕТ

Итак, изученное форматирование таблиц применяется сразу ко всем вашим таблицам. Если же вы хотите задать для определенной таблицы индивидуальный стиль (или для каждой ее ячейки), измените выбранный селектор. Чтобы использовать особый дизайн определенной таблицы, присвойте ей имя класса — `<table class="stocks">` — и создайте несколько селекторов потомков вида `.stocks td` или `.stocks th`, чтобы применить различное форматирование к отдельным ячейкам. Если вы хотите изменить форматирование определенной ячейки в таблице, примените класс `<td class="subtotal">` к элементу и создайте класс для форматирования этой ячейки.

Brand	Price	Power Source	Mini-Review	
Chinook Push-o-matic Indoor Mower	\$247.00	Mechanical	The latest model of the Chinook mower is a big improvement over last year's model. It's smooth gliding action is perfect for even massively over grown sod. Its handling around corners is superb -- perfect for those tight areas around sofas and coffee tables.	top
Sampson Deluxe Apartment Mower	\$370.00	Mechanical	In our battery of 7 mowing tests, the Sampson scored 9 or above on each. The fine blades turn even large weeds into tiny cuttings, perfect for composting or salad garnishes.	baseline
Sampson Deluxe Apartment Mower	\$370.00	Mechanical	In our battery of 7 mowing tests, the Sampson scored 9 or above on each. The fine blades turn even large weeds into tiny cuttings, perfect for composting or salad garnishes.	middle
Chinook Push-o-matic Indoor Mower	\$247.00	Mechanical	The latest model of the Chinook mower is a big improvement over last year's model. It's smooth gliding action is perfect for even massively over grown sod. Its handling around corners is superb -- perfect for those tight areas around sofas and coffee tables.	bottom

Table 1: CosmoFarmer.com's Indoor Mower Roundup

**Рис. 11.4.** Если в ячейке используются отступы, ее содержимое на самом деле никогда не выравнивается по верхним или нижним границам: всегда есть промежуток, равный размеру отступа

Создание границ

Свойство border (см. раздел «Добавление границ» главы 7) поддерживается таблицами так же хорошо, как и другими элементами, но вам нужно кое-что учитывать. Во-первых, использование границ в стиле, формирующем элемент table, выделяет только таблицу, а не какие-либо определенные ячейки. Во-вторых, при применении границ к ячейкам (td { border: 1px solid black; }) образуется видимый интервал между ячейками сверху, как это показано на рис. 11.5. Чтобы правильно управлять границами, необходимо понять принцип визуализации ячеек таблицы браузерами и разобраться со свойством border-collapse.

- **Управление промежутками между ячейками таблицы.** Если не указано другое, браузеры расставляют в таблице интервалы между ячейками около 2 пикселей. Этот промежуток заметен, когда вы задаете границы для ячеек. В CSS доступно свойство border-spacing, которое можно использовать, чтобы управлять размером этого промежутка. Свойство нужно применять к самой таблице, а если требуется удалить пустое пространство, добавляемое браузером между ячейками, присвойте свойству border-spacing значение 0:

```
table {
  border-spacing: 0;
}
```

Brand	Price	Power Source	Mini Review
Chinook Push-o-matic Indoor Mower	\$247.00	Mechanical	The latest model of the Chinook mower is a big improvement over last year's model. It's smooth gliding action is perfect for even massively over grown sod. Its handling around corners is superb -- perfect for those tight areas around sofas and coffee tables.

Table 1: CosmoFarmer.com's Indoor Mower Roundup

Brand	Price	Power Source	Mini-Review
Chinook Push-o-matic Indoor Mower	\$247.00	Mechanical	The latest model of the Chinook mower is a big improvement over last year's model. It's smooth gliding action is perfect for even massively over grown sod. Its handling around corners is superb -- perfect for those tight areas around sofas and coffee tables.

Table 1: CosmoFarmer.com's Indoor Mower Roundup

Brand	Price	Power Source	Mini-Review
Chinook Push-o-matic Indoor Mower	\$247.00	Mechanical	The latest model of the Chinook mower is a big improvement over last year's model. It's smooth gliding action is perfect for even massively over grown sod. Its handling around corners is superb -- perfect for those tight areas around sofas and coffee tables.

Table 1: CosmoFarmer.com's Indoor Mower Roundup

**Рис. 11.5.** Браузеры обычно добавляют промежутки между всеми ячейками в таблице (вероятно, вы не заметите дополнительное пространство, пока не добавите границы, как показано в верхнем изображении)

Но если вы предпочитаете оставлять пустое пространство между ячейками, то его можно добавить:

```
table {  
  border-spacing: 2px;  
}
```

- **Удаление двойных границ.** Даже если вы уберете промежутки между ячейками, границы, заданные для ячеек, будут удваиваться. Это происходит потому, что нижняя граница одной ячейки добавляется к верхней границе нижней ячейки, и в итоге образуется линия, которая в два раза толще заданной ширины границы (см. рис. 11.5, *посередине*). Самый лучший способ избавиться от этого (а также от промежутков между ячейками) — использовать свойство `border-collapse`. Ему можно присвоить одно из двух значений — `separate` или `collapse`. Значение `separate` эквивалентно тому, как обычно и отображаются таблицы: с промежутками между ячейками и двойными границами. Значение `collapse` схлопывает границы таблицы, позволяя избавиться от интервалов и удвоения границ (см. рис. 11.5, *внизу*). Используйте значение `collapse` следующим образом:

```
table { border-collapse: collapse; }
```

**ПРИМЕЧАНИЕ**

Если присвоить свойству `border-collapse` значение `collapse`, свойство `border-spacing` работать не будет.

- **Скругление углов.** Чтобы добавить к ячейкам таблиц (но не к самим таблицам) скругленные углы, можно воспользоваться свойством `border-radius`. Например, если нужно добавить ячейкам таблицы контуры и скругленные углы, создайте следующий стиль:

```
td {  
    border: 1px solid black;  
    border-radius: 5px;  
}
```

Учтите, что при присвоении свойству `border-collapse` значения `collapse` браузеры проигнорируют установку для ячеек таблицы свойства `border-radius` и визуализируют обычные прямые углы.

**ПРИМЕЧАНИЕ**

К таблицам и ячейкам можно применять свойство `box-shadow`. Если применить его к таблице, тень появится позади всей таблицы, а если его применить к отдельным ячейкам, то отдельная тень будет у каждой ячейки.

## Форматирование строк и столбцов

Чередование затемненных строк в таблице, как показано на рис. 11.6, — это всего лишь один из обычных способов оформления таблицы. Изменив внешний вид каждой строки в таблице, вы позволите посетителям с комфортом просматривать нужные данные. Добавив ранее рассмотренный селектор `nth-of-type`, можно чередовать фон строк:

```
tr:nth-of-type(odd) { background-color: red; }  
tr:nth-of-type(even) { background-color: blue; }
```

Если применять один и тот же фон к чередующимся строкам всех таблиц не нужно, следует просто добавить к целевой таблице имя класса (например, `products`, для таблицы с информацией о товарах), а затем воспользоваться селектором потомка:

```
.products tr:nth-of-type(odd) { background-color: red; }  
.products tr:nth-of-type(even) { background-color: blue; }
```

Вы также не ограничены исключительно цветом. Можно использовать фоновые изображения (см. раздел «Добавление фоновых изображений» главы 8) или даже линейные градиенты (см. подраздел «Линейные градиенты» раздела «Использование градиентных фонов» главы 8) для создания более привлекательного вида, например тонкого затемнения строки таблицы с заголовками, как на рис. 11.6 (вы проработаете похожий в практикуме этой главы). Вы также можете задавать селектор потомков для каждой ячейки в этой строке. Это очень удобно, когда вы



применяете стили ко всем ячейкам в одном столбце, используя для каждой определенный стиль и вид, например `<td class="price">`.

#### ПРИМЕЧАНИЕ

Селектор `nth-of-type` в Internet Explorer 8 и более ранних версиях браузера не поддерживается.

В процессе форматирования столбцов надо немного схитрить. В языке HTML доступны элементы `colgroup` и `col`, определяющие группу столбцов и отдельный столбец соответственно. Вы можете добавить по одному элементу `col` к каждому столбцу в таблице и далее обозначить их с помощью класса или идентификатора (см. HTML-код в разделе «Разумное применение таблиц» этой главы).

Этими элементами поддерживаются только два набора свойств: `width` и свойства фона (`background-color`, `background-image` и т. д.). Однако и они могут быть очень полезными. Если вы хотите настроить ширину всех строк в столбце, то можете пропустить все атрибуты HTML и форматировать столбцы, применив стиль к элементу `col`. Допустим, используется следующий фрагмент HTML-кода: `<col class="price">`. Вы можете добавить этот стиль в таблицу CSS и установить ширину каждой ячейки в данном столбце равной 200 пикселям:

```
.price { width: 200px; }
```

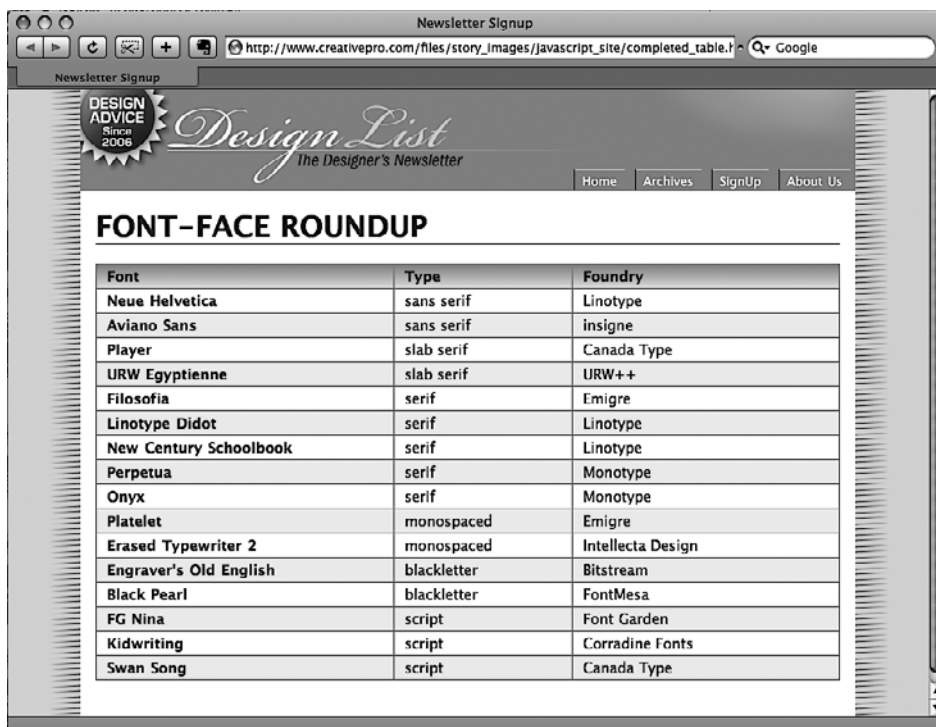


Рис. 11.6. Благодаря чередующемуся фоновому цвету строк намного проще воспринимать данные

Чтобы выделить столбец, вы можете воспользоваться свойствами форматирования фона. И снова считайте, что у вас есть элемент `col`, к которому применен класс `price`:

```
.price { background-color: #F33; }
```

Имейте в виду, что фон в столбцах помещается позади ячеек, поэтому, если вы установите фоновый цвет или изображение в элементах `td` или `th`, фон столбцов не будет виден.

#### ПРИМЕЧАНИЕ

Обычно браузеры отображают границы и фоновый цвет каждой пустой ячейки в таблице. Каскадные таблицы стилей позволяют скрыть пустые ячейки. Для этого добавьте свойство `empty-cells: hide` к стилю, примененному к таблице:

```
table {  
  empty-cells: hide;  
}
```

Однако будьте осторожны. Если свойству `border-collapse` присвоить значение `collapse`, браузеры проигнорируют свойство `empty-cells` и отобразят границы и фоновый цвет даже пустых ячеек.

## Форматирование веб-форм

Веб-формы — это основной способ коммуникации пользователя с сайтом. Передавая информацию в веб-форму, вы можете подписаться на рассылку, поискать какие-либо товары в базе данных, ввести изменения в профиль социальной сети или заказать конструктор LEGO *Star Wars*, о котором давно мечтали.

Совсем не обязательно, чтобы ваши веб-формы выглядели так же, как и большинство других форм во Всемирной паутине. Применяя каскадные таблицы стилей, вы можете форматировать элементы веб-формы, чтобы они выглядели так же, как и другие элементы сайта: использовали те же шрифты, фоновые изображения и поля. Нет никаких отдельных свойств каскадных таблиц стилей для веб-форм, и вы можете применять практически любые из описанных в этой книге свойств для форматирования элементов веб-форм.

### В КУРС ДЕЛА!

#### Придерживайтесь одного стиля при оформлении веб-форм

Существует несколько причин, по которым надо с осторожностью подходить к изменению привычных и узнаваемых элементов интерфейса, таких как кнопки отправки данных или раскрывающиеся списки. Большинство пользователей уже привыкли к тому, как выглядят и работают элементы веб-форм. В целом внешний вид кнопки отправки данных не меняется от сайта к сайту. Когда люди видят ее, они уже знают, для чего предназначена эта кнопка и когда ее следует нажимать. Если вы *кардинально* измените внешний вид веб-формы, это может привести к тому, что у пользователей возникнут некоторые затруднения при ее заполнении.

Добавив пунктирную границу к какому-либо элементу веб-формы, вы можете добиться того, что пользователь в результате просто не обратит на него внимания и пропустит. А если этот элемент — текстовое поле, предназначенное для ввода адреса электронной почты, на который вы будете отправлять рассылки, можно в итоге потерять несколько пользователей из-за того, что они его попросту пропустили. И напоследок обязательно удостоверьтесь, что пользователи понимают, что перед ними размещена именно веб-форма для ввода данных, а не что-либо иное.

## СОВЕТ

---

Прекрасные примеры дизайнов веб-форм вы найдете по адресу [tinyurl.com/pjwlt6y](http://tinyurl.com/pjwlt6y).

---

## Элементы веб-форм

Для создания веб-форм предназначен широкий спектр HTML-элементов. Некоторые из элементов веб-форм форматировать проще (например, текстовые поля), другие — сложнее (переключатели). Рассмотрим несколько простых элементов веб-форм, а также типы свойств, которые они поддерживают.

- **Группирование элементов веб-формы.** Элемент `fieldset` предназначен для группировки элементов, связанных друг с другом. Большинство браузеров нормально отображают фоновые цвета, фоновые изображения и границы этого элемента. При использовании отступов появляются промежутки между краями элементов `fieldset` и их содержимым.
- **Заголовки групп элементов веб-формы.** Элемент `legend` следует за `fieldset` и содержит название (текстовую метку) группы элементов веб-формы. Оно отображается в центре верхней границы элемента `fieldset`. Если элемент `fieldset` представляет, например, адрес доставки, можете добавить следующий код: `<legend>Адрес доставки</legend>`. С помощью каскадных таблиц стилей можно изменить свойства шрифта заголовка группы, добавить фоновые цвета и изображения, а также определить границы.
- **Текстовые поля.** Элементы `<input type="text">`, `<input type="password">` и `<textarea>` создают текстовые поля веб-формы. Эти элементы отлично поддерживаются браузерами. Вы можете изменить размер, семейство шрифтов, цвет и другие свойства шрифта текстовых полей, а также добавить границы, фоновые цвета и изображения. Вы можете задать ширину и высоту этих полей с помощью свойств `width` и `height` соответственно.
- **Кнопки.** Кнопки веб-формы, такие как `<input type="submit">`, позволяют пользователям передавать данные на сервер, сбрасывать содержимое веб-формы или предпринимать какие-либо другие действия. Многие браузеры позволяют экспериментировать с форматированием текста, границ, фона, с тенями и скруглением углов. Вы также можете выравнивать текст кнопки по левому или правому краю либо по центру с помощью свойства `text-align`. Особенно хорошо смотрятся на кнопках линейные градиенты.
- **Раскрывающиеся списки.** Списки, созданные с помощью элемента `select`, также можно форматировать, используя каскадные таблицы стилей. Большинство браузеров позволяют настраивать шрифт, размер шрифта, фоновый цвет/изображение и границы. Но фоновые цвета и изображения не всегда добавляются к раскрывающимся спискам, когда они раскрываются для отображения всех вариантов выбора.
- **Флажки и переключатели.** Большинство браузеров не позволяют применять форматирование к этим элементам, поэтому без лишней необходимости не стоит этого делать.

**ИНФОРМАЦИЯ ДЛЯ ОПЫТНЫХ ПОЛЬЗОВАТЕЛЕЙ****Атрибут: основной селектор элементов веб-формы**

Когда заходит речь об оформлении веб-форм, селекторы тегов — это не единственный способ. В конце концов, текстовые поля, переключатели, флажки, поля ввода паролей и кнопки используют один и тот же HTML-элемент `input`. Хотя ширина 200 пикселей вполне подходит для текстового поля, скорее всего, вам не захочется, чтобы элемент флажка был таким же большим, поэтому вы не сможете использовать элемент `input` для изменения ширины. Самым лучшим, учитывающим кросс-браузерность способом форматирования исключительно текстовых полей ввода будет создание отдельных классов для каждого из них, например `<input type="text" class="textfield" name="email">`, и стиля для форматирования.

Однако вы можете воспользоваться преимуществами более совершенного селектора атрибутов, позволяющего настроить внешний вид веб-формы, не обращаясь при

этом к классам. Он отбирает HTML-элементы на основе одного из атрибутов. Атрибут `type` определяет, какой тип элемента веб-формы создается с помощью `input`. Так, значение атрибута `type` для текстового поля — `text`. Чтобы создать стиль, который будет изменять фоновый цвет всех однострочных полей ввода на синий, необходимо создать следующий селектор и стиль:

```
input[type="text"] {  
    background-color: blue;  
}
```

Если вы измените значение `text` на `submit`, то создадите стиль исключительно для кнопок.

Селекторы атрибутов поддерживают все современные браузеры, даже Internet Explorer 7, поэтому применять их для форматирования веб-форм можно без опасений.

## Компоновка веб-форм с помощью каскадных таблиц стилей

Все, что надо для создания веб-формы, — добавить несколько текстовых элементов на веб-страницу. Однако зачастую это выглядит хаотично (рис. 11.7, *слева*). Веб-формы обычно выглядят лучше, когда их метки и элементы выровнены по столбцам (см. рис. 11.7, *справа*).

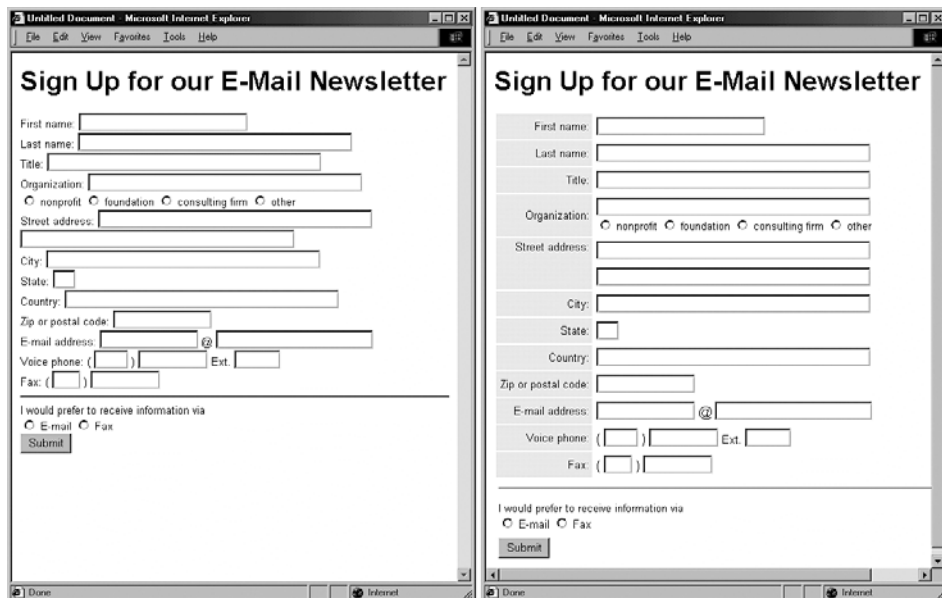
Вы можете добиться этого несколькими путями. Наиболее простой способ — использовать HTML-таблицы. Хотя элементы и метки веб-форм — это, по сути, не табличные данные, они хорошо приспособляются к расположению в формате столбцов/строк. Нужно лишь расположить текстовые метки («Имя», «Номер телефона» и т. д.) в одном столбце, а элементы веб-формы — в другом.

Используя каскадные таблицы стилей, вы также можете создать двухколоночную веб-форму, показанную на рис. 11.7 (преимущество — меньше HTML-кода). Рассмотрим основы этого подхода.

### 1. Поместите каждую текстовую метку в элемент.

Очевидным выбором для этого будет элемент `label`, так как он создан именно для меток веб-формы. Но вы не можете *всегда* использовать только его. Рядом с переключателями обычно располагаются вопросы типа «Какой ваш любимый цвет?», а для каждого положения переключателя используется метка в элементе `label`. Какой же элемент вы будете использовать для применения к тексту вопроса? На помощь придет элемент `span`: `<span>Какой ваш любимый цвет?</span>`. Затем добавьте класс к каждому из этих элементов: `<span class="label">` и на-

значьте класс только тем элементам `label`, которые следует отобразить в левом столбце (в веб-форме, показанной на рис. 11.7, это были бы метки `First name`, `Last name` и т. д., но не элементы `label` для переключателей).



**Рис. 11.7.** Зачастую элементы веб-формы не очень хорошо компонуются с текстом, что приводит к тому, что они располагаются хаотично и ухудшают читабельность веб-формы (слева). Решением данной проблемы может стать организация элементов веб-формы в столбцы (справа) с использованием HTML-таблиц или CSS

2. Присвойте свойству `display` значение `inline-block` и укажите ширину.

По умолчанию элементы `label` и `span` — строчные, игнорирующие многие настройки, доступные блочным элементам, включая свойства `width`, `height` и `text-align`. Но если превратить метки в блочные элементы (с помощью значения `inline-block`), они по-прежнему будут располагаться рядом с элементами веб-формы (оправдывая свое строчное происхождение). Значение `width` предоставит достаточно пространства, позволяя уместить весь текст метки в одну строку, если это возможно. Можно создать класс, код которого будет иметь следующий вид:

```
.label {
  display: inline-block;
  width: 20em;
}
```

Настройки `width` и `inline-block` превращают метки с классом `label` в небольшие равные по размеру блоки и позволяют выровнять все элементы веб-формы по левому краю.

3. Отформатируйте метки и элементы веб-формы.

Есть еще несколько улучшений, которые вы можете использовать. Вам нужно присвоить значение `top` свойству `vertical-align`, чтобы верхний край текста метки

был выровнен с верхним краем элемента веб-формы. Нужно также выровнять текст метки по правому краю, чтобы каждая метка отображалась рядом с соответствующим элементом веб-формы. И наконец, немного увеличив поле справа, можно создать небольшой промежуток между метками и элементами веб-формы.

```
.label {
  float: left;
  width: 20em;
  vertical-align: top;
  text-align: right;
  margin-right: 15px;
}
```

В итоге получится аккуратная веб-форма. Можно добавить и другие улучшения, выделив, например, метки полужирным шрифтом и изменив их цвет. Во втором практикуме этой главы приведен пример, в котором пошагово расписаны все необходимые для этого действия.

## ИНФОРМАЦИЯ ДЛЯ ОПЫТНЫХ ПОЛЬЗОВАТЕЛЕЙ

### Псевдоклассы веб-формы

Существует несколько псевдоклассов, предназначенных для форматирования веб-форм. Псевдокласс `:focus` позволяет создать селектор, изменяющий внешний вид текстового поля при щелчке на нем кнопкой мыши или при переходе на него нажатием клавиши Tab (это называется *фокусировкой* на элементе веб-формы). Этим можно воспользоваться для изменения размера, цвета фона, шрифта и других применяемых к полю свойств каскадных таблиц стилей. Практический пример приведен в практикуме.

Псевдокласс `:checked` поддерживается переключателями и флажками. Он предназначен для форматирования этих элементов, но обычно браузеры в отношении оформления этих элементов веб-формы ведут себя «сдержанно» и большинство свойств к ним не применяют. Но, если хотите внести какие-то изменения, можно создать стиль с именем `:checked` и добавить какие-нибудь свойства CSS. Все указанное в этом стиле будет применяться только к переключателям и флажкам веб-формы. Если нужно отформатировать конкретный флажок при его установке, можно создать следующий класс:

```
.special:checked
```

А затем применить его только к данному флажку:

```
<input type="checkbox" class="special"
```

Элементы веб-формы можно делать активными и неактивными. Неактивный элемент веб-формы изменить невозможно: например, нельзя ввести текст в неактивное текстовое поле или установить/снять неактивный флажок. Изменить состояние элемента веб-формы можно только с помощью JavaScript-сценария, поэтому для использования псевдоклассов `:enabled` и `:disabled` нужно изучить JavaScript. Тем не менее эти псевдоклассы могут оказаться весьма полезными.

Предположим, у вас есть веб-форма заказа с адресными полями «выставление счета» и «доставка». Если рядом с адресом доставки указать флажок «адрес доставки идентичен адресу выставления счета», посетитель может установить этот флажок. Используя JavaScript-сценарий, можно сделать неактивным поле адреса доставки, чтобы в него нельзя было случайно что-нибудь ввести. Для наглядности цвет элемента веб-формы можно изменить, к примеру, на серый:

```
:disabled {
  background-color: #333;
}
```

Существуют и другие псевдоклассы, предназначенные для работы с такими специальными возможностями HTML5-форм, как встроенная проверка правильности заполнения веб-формы. Чтобы узнать о них, посетите сайт [tinyurl.com/7hg9rpt](http://tinyurl.com/7hg9rpt).

## Практикум: форматирование таблиц

Язык HTML отлично подходит для создания таблиц, но вам необходим также и CSS для создания стилей. Как вы узнали из раздела «Разумное применение таблиц» этой главы, для создания HTML-таблицы достаточно довольно простого кода. Загрузив примеры к книге, вы найдете уже готовую HTML-таблицу, на которой можете потренироваться в использовании каскадных таблиц стилей. В этом практикуме вы отформатируете строки, столбцы и ячейки таблицы, замените установленный шрифт более привлекательным, а также зададите фоновый цвет.

Чтобы начать обучение, вы должны иметь в распоряжении файлы примеров. Для этого нужно загрузить файлы для выполнения заданий практикума, расположенные по адресу [github.com/mrightman/css\\_4e](https://github.com/mrightman/css_4e). Перейдите по ссылке и загрузите ZIP-архив (нажав кнопку **Download ZIP** в правом нижнем углу страницы). Файлы текущего практикума находятся в папке 11.

1. Загрузите браузер и откройте в нем файл `table.html`, расположенный в папке `11/table`.

Эта страница содержит простую HTML-таблицу. У нее есть название, строка заголовков и девять строк ячеек с данными (рис. 11.8). Кроме того, три раза используется элемент `col` для определения трех столбцов с данными. Как вы скоро увидите, элемент `col` весьма полезен при форматировании, поскольку позволяет устанавливать ширину всех ячеек в столбце.

2. Откройте файл `main.css`, расположенный в папке `11/table/css`, в редакторе HTML-кода.

Эта внешняя таблица стилей, связанная с файлом `table.html`. Для начала создадим стиль, который задает ширину таблицы и шрифт текста. К этой таблице применен класс `inventory`, поэтому вы можете использовать селектор класса для форматирования только этой таблицы.

---

### ПРИМЕЧАНИЕ

Браузеры кэшируют файлы, поэтому, если вы внесли изменения в файл `main.css`, но не видите никаких изменений на странице, необходимо обновить кэш браузера. Чтобы узнать, как это сделать, перейдите по ссылке [tinyurl.com/brfbqmx](https://tinyurl.com/brfbqmx).

---

3. Добавьте следующий стиль в нижней части файла `main.css`:

```
.inventory {  
  font-family: "Trebuchet MS", Arial, Helvetica, sans-serif;  
  width: 100%;  
  margin-top: 25px;  
}
```

Если вы не установите ширину таблицы, она растянется по размеру контента внутри нее. В данном случае мы установим ширину, равную 100 %, чтобы таблица растянулась по всей ширине ее контейнера (здесь это элемент `body` с установленной шириной и автоматическими значениями полей слева и справа для центровки в окне браузера). При изменении шрифта в элементе `table` используется наследование, чтобы изменился шрифт и во всех остальных элементах

<table id="inventory">

Table 1: Current Inventory

<caption>

Product

Price

Rating

<th>

Vitae Quam Lorem

\$19.95

★★★★★

<td>

In Tempus Velit

\$14.55

★★★★

(ячейки таблицы)

Lorem Ipsum Dolor Sat

Priceless

★★★★★

Quis Felis Fringilla

\$29.95

★★

Nunc Sem Pharetra

\$75.99

★★★

Vel Faucibus Elit

\$82.00

★

Non Adipiscing Vitae

\$1.95

★★★★★

Aenean Orci Ante

\$17.95

★★★★★

Venenatis Non Adipiscing

\$44.00

★★★★★

<col id="product">

<col id="price">

<col id="rating">

Table 1: Current Inventory

PRODUCT	PRICE	RATING
Vitae Quam Lorem	\$19.95	★★★★★
In Tempus Velit	\$14.55	★★★★
Lorem Ipsum Dolor Sat	Priceless	★★★★★
Quis Felis Fringilla	\$29.95	★★
Nunc Sem Pharetra	\$75.99	★★★
Vel Faucibus Elit	\$82.00	★
Non Adipiscing Vitae	\$1.95	★★★★★
Aenean Orci Ante	\$17.95	★★★★★
Venenatis Non Adipiscing	\$44.00	★★★★★

**Рис. 11.8.** Отформатировав таблицу с помощью границ, фоновых цветов и других свойств каскадных таблиц стилей, можно не только разукрасить скучные HTML-таблицы (вверху), но и расположить данные в таблице в более читабельной форме (внизу)

внутри страницы — caption, th, td и т. д. И наконец, свойство margin-top смещает таблицу вниз от находящегося выше ее заголовка.

Далее создадим стиль для заголовка таблицы.

4. Добавьте еще один стиль после только что созданного:

```
.inventory caption {
  text-align: right;
  font-size: .85em;
  margin-bottom: 10px;
}
```

Этот селектор потомков влияет только на элемент caption, который появляется внутри другого элемента с классом inventory (элемент table в нашем примере). Элемент caption сообщает о содержимом таблицы. В нашем случае он не должен быть в центре внимания, поэтому мы оставили шрифт небольшого размера и выровняли текст по правому краю. Свойство margin-bottom добавляет небольшое пространство между заголовком и таблицей.



Когда вы читаете данные в таблице, очень легко сбиться. Необходим хороший визуальный ориентир. Если мы добавим границы вокруг ячеек, они визуально выделят информацию.

5. Добавьте еще один групповой стиль:

```
.inventory td, .inventory th {  
    font-size: 1.1em;  
    border: 1px solid #DDB575;  
}
```

Групповой селектор предназначен для форматирования элементов заголовков таблицы (th) и ячеек (td), он уменьшает размер шрифта и рисует границу вокруг каждого заголовка и каждой ячейки. Как правило, браузеры добавляют промежутки между всеми ячейками, поэтому вокруг границ и появляются небольшие промежутки (рис. 11.9). Из-за них вся таблица выглядит какой-то разрозненной. Сейчас мы это исправим.

6. Добавьте свойство border-collapse в стиль, который вы создали на шаге 3. Теперь его содержимое будет таким:

```
.inventory {  
    font-family: "Trebuchet MS", Arial, Helvetica, sans-serif;  
    width: 100%;  
    margin-top: 25px;  
    border-collapse: collapse;  
}
```

Свойство border-collapse убирает промежутки между ячейками. Оно также способствует тому, что соприкасающиеся границы ячеек схлопываются, что избавляет таблицу от толстых, некрасивых границ. Если не воспользоваться свойством border-collapse, то нижняя граница заголовка таблицы соединится с верхней границей ячеек таблицы, что приведет к удвоению границы, и ее толщина станет равна 2 пикселям.

Если вы сейчас взглянете на таблицу, то увидите, что данные в ней выглядят намного лучше, но информация в каждой из ячеек немного сжата. Добавим небольшой отступ, чтобы избавиться от этого.

7. Добавьте отступы с помощью группового селектора, созданного в шаге 5:

```
.inventory td, .inventory th {  
    font-size: 1.1em;  
    border: 1px solid #DDB575;  
    padding: 3px 7px 2px 7px;  
}
```

Хотя верхняя строка таблицы с заголовками и выделяется из-за использования в ней полужирного шрифта, вы можете сделать ее более наглядной.

8. Создайте новый стиль после стиля .inventory td, .inventory th:. Он будет использоваться только для форматирования заголовка таблицы:

```
.inventory th {  
    text-transform: uppercase;
```

```
text-align: left;
padding-top: 5px;
padding-bottom: 4px;
}
```

Этот стиль — отличный пример каскадности. Групповой селектор `td, th` определяет общие свойства форматирования для двух типов ячеек. Указав стиль, предназначенный *только* для элемента `th`, вы в дальнейшем сможете использовать его для изменения внешнего вида заголовков таблицы. Например, свойства `padding-top` и `padding-bottom` здесь заменяют аналогичные настройки, определенные в селекторе в шаге 7. Но, поскольку вы не заместили настройки левого и правого отступа, элементы `th` сохраняют 7 пикселей каждого отступа, определенного в шаге 7. Этот стиль также делает все буквы прописными и выравнивает текст по левому краю ячейки.

Заголовки таблицы по-прежнему недостаточно привлекательны, и, кроме того, складывается впечатление, что таблица размещена на фоне страницы. Чтобы исправить положение, можно добавить фоновое изображение.

Product	Price	Rating
Vitae Quam Lorem	\$19.95	★★★★★
In Tempus Velit	\$14.55	★★★
Lorem Ipsum Dolor Sat	Priceless	★★★★★
Quis Felis Fringilla	\$29.95	★★
Nunc Sem Pharetra	\$75.99	★★★
Vel Faucibus Elit	\$82.00	★
Non Adipiscing Vitae	\$1.95	★★★★★
Aenean Orci Ante	\$17.95	★★★★★
Venenatis Non Adipiscing	\$44.00	★★★★★

Product	Price	Rating
Vitae Quam Lorem	\$19.95	★★★★★
In Tempus Velit	\$14.55	★★★
Lorem Ipsum Dolor Sat	Priceless	★★★★★
Quis Felis Fringilla	\$29.95	★★
Nunc Sem Pharetra	\$75.99	★★★
Vel Faucibus Elit	\$82.00	★
Non Adipiscing Vitae	\$1.95	★★★★★
Aenean Orci Ante	\$17.95	★★★★★
Venenatis Non Adipiscing	\$44.00	★★★★★

**Рис. 11.9.** Обычно браузеры добавляют промежутки между ячейками таблицы (вверху). В позиции их соприкосновения граница удваивается. Свойство `border-collapse` с присвоенным значением `collapse` решает обе проблемы (внизу)

9. Измените стиль `th`, добавив линейный градиент и изменив цвет шрифта:

```
.inventory th {
  text-transform: uppercase;
  text-align: left;
  padding-top: 5px;
  padding-bottom: 4px;
  background: rgb(229,76,16);
  background: linear-gradient(to bottom, rgb(229,76,16), rgb(173,54,8));
  color: white;
}
```

Сначала в качестве фона добавляется сплошной RGB-цвет, благодаря этому Internet Explorer 9 и более ранние версии (которые не поддерживают градиенты) отобразят по крайней мере цвет. В следующей строке добавляется линейный градиент с использованием официального синтаксиса. И наконец, цвет текста меняется на белый.

Когда в таблице используется много данных, расположенных во множестве строк и столбцов, иногда бывает трудно определить, к какой строке какие данные относятся. Для повышения читабельности таблицы можно использовать чередующиеся строки, применив селектор `nth-of-type`.

10. Добавьте еще два стиля в файл `main.css`:

```
.inventory tr:nth-of-type(even){
  background-color: rgba(255,255,255,.1);
}
.inventory tr:nth-of-type(odd){
  background-color: rgba(229,76,16,.1);
}
```

Как уже говорилось, селектор `nth-of-type` используется для выбора дочерних элементов, отвечающих той или иной числовой схеме, например каждый пятый абзац. В данном случае первый стиль выбирает каждый четный элемент `tr`, а второй — каждый нечетный элемент `tr`.

И наконец, нужно подогнать ширину тех ячеек, которые находятся ниже в столбцах «Цена» и «Рейтинг». Один из способов заключается в методичном добавлении к этим ячейкам имен классов и создании стиля класса с установленной шириной ячейки. Но лучше будет воспользоваться элементом `col`, что позволит назначить класс или идентификатор всей совокупности ячеек столбца. На рис. 11.9, показано, что эти два столбца имеют идентификаторы `price` и `rating`. С помощью одного группового селектора можно упростить задание ширины для этих двух столбцов.

11. Добавьте еще один стиль в файл `main.css`:

```
#price, #rating {
  width: 15%;
}
```

Сохраните таблицу стилей и откройте файл `table.html` в браузере. Она должна выглядеть так, как показано на рис. 11.9, *внизу*. Готовый пример находится в папке `11_finished\table`.

## Практикум: форматирование веб-форм

В этом уроке вы узнаете, как с помощью каскадных таблиц стилей привести веб-форму в порядок, а также сделать ее более привлекательной. Если вы откроете в браузере страницу `form.html` из папки `11\form`, то увидите, что на ней опубликована простая веб-форма для регистрации подписчика на вымышленном сайте (рис. 11.10). Форма задает несколько вопросов, и в ней используются несколько элементов ввода, таких как текстовые поля, переключатели и раскрывающиеся списки. Если рассматривать ее в качестве веб-формы для подписки, выглядит она вполне нормально, но слегка мрачновато. На следующих страницах мы отформатируем текст, выделим вопросы и элементы веб-формы, а также добавим несколько других улучшений.

1. Откройте файл `global.css`, расположенный в папке `11\form`, в редакторе HTML-кода.

Сейчас вы добавите новый стиль к внешней таблице стилей, связанной с файлом `form.html`. Начнем с увеличения размера шрифта, используемого в веб-форме, чтобы сделать ее более читабельной.

2. Добавьте следующий стиль в нижней части таблицы стилей:

```
.subform {  
    font-size: 1.2em;  
    color: white;  
    font-family:Tahoma, Geneva, sans-serif;  
}
```

Этой веб-форме присвоен класс `subForm`, поэтому новый стиль изменяет шрифт, его размер и цвет для всего текста между открывающим и закрывающим тегами элемента `form`.

Пришло время поработать над макетом. Чтобы удачнее скомпоновать элементы веб-формы, воспользуемся двумя столбцами, в одном из которых будут находиться вопросы (текстовые метки), а в другом — ответы (элементы веб-формы).

3. Добавьте следующий код в таблицу стилей:

```
.subform .label {  
    display: inline-block;  
    width: 200px;  
    vertical-align: top;  
}
```

Этот селектор потомков используется для всех элементов класса `.label`, расположенных в веб-форме. Стиль превращает метки из строчных элементов (которые не поддерживают изменение ширины) в блочные. Свойство `width` устанавливает для области меток значение ширины, равное 200 пикселям, а свойство `vertical-align: top` выравнивает текст меток по верхнему краю соответствующих элементов веб-формы. В результате после применения этого стиля к каждому вопросу в веб-форме вы получите выровненный по ширине столбец. Чтобы оценить полученный результат, вам необходимо применить созданный класс к соответствующим элементам веб-формы.

**Рис. 11.10.** Элемент `table` обычно используется для размещения вопросов в веб-форме. Можно также задействовать CSS, чтобы превратить хаотичный набор текстовых меток и элементов веб-формы (наподобие той, что вы видите на рисунке) в организованную и привлекательную структуру

- Откройте файл `form.html`. Найдите следующий код `<label for="name">` и добавьте класс `class="label"` таким образом:

```
<label for="name" class="label">
```

Вы должны повторить то же самое для всех вопросов на веб-форме.

- Повторите шаг 4 для следующих фрагментов HTML-кода: `<label for="email">`, `<label for="refer">` и `<label for="comments">`.

В веб-форме есть еще один дополнительный вопрос — **Rate your apartment farming skills**. Он не размещен в элементе `label`, так как для ответа на него используется одно из положений переключателя, каждое из которых имеет отдельную метку. Вам нужно добавить элемент `span` к тексту, чтобы можно было воспользоваться стилем `label`.

- Найдите текст **Rate your apartment farming skills** и поместите его в элемент `span`, использующий класс `label`:

```
<span class="label"> Rate your apartment farming skills</span>
```

Теперь вопросы размещены в отдельном столбце (рис. 11.11, *вверху*). Но они бы выглядели лучше, если бы были немного смещены и выделены соответствующим образом.

- В файле `global.css` измените стиль `#subForm .label`, который вы создали в шаге 3, следующим образом:

```
.subform .label {
  display: inline-block;
  width: 200px;
  vertical-align: top;
  text-align: right;
```

Регистрация подписчика

Как вас зовут?

Ваш адрес электронной почты

Навыки квартирного ремонта ☐ Новичок ☐ Опытный ☐ Профи

Откуда вы узнали о нас?

Дополнительные комментарии

Регистрация подписчика

Как вас зовут?

Ваш адрес электронной почты

Навыки квартирного ремонта ☐ Новичок ☐ Опытный ☐ Профи

Откуда вы узнали о нас?

Дополнительные комментарии

**Рис. 11.11.** Иногда небольшие и едва заметные изменения могут сделать веб-форму более читабельной

```
margin-right: 10px;  
font-weight: bold;  
color: rgba(255,255,255,.5);  
}
```

Просмотрите страницу в браузере. Веб-форма должна выглядеть так, как показано на рис. 11.11, *внизу*.

Веб-форма выглядит уже намного лучше, но кнопка **Subscribe**, размещенная у левого края, смотрится не на своем месте. Выровняем ее, как и остальные элементы веб-формы.

8. Добавьте еще один стиль в файл `global.css`.

```
.subform input[type="submit"] {  
    margin-left: 220px;  
}
```

Поскольку кнопки отправки данных создаются путем добавления к элементу `input` атрибута `type="submit"`, выбрать их можно с помощью селектора атрибута. Тогда не придется создавать класс и применять его к кнопкам отправки данных.

9. Измените только что созданный стиль кнопки **Subscribe**, добавив дополнительные свойства:

```
.subform input[type="submit"] {  
    margin-left: 220px;  
    padding: 10px 25px;  
    font-size: 1em;  
    color: white;  
}
```

Свойство `padding` добавляет пространство между текстом и краями кнопки, а свойства `font-size` и `color` форматируют шрифт текста кнопки.

Теперь можно проявить изобретательность и добавить к кнопке градиент.

10. Отредактируйте стиль кнопки **Subscribe** следующим образом:

```
.subform input[type="submit"] {  
    margin-left: 220px;  
    padding: 10px 25px;  
    font-size: 1em;  
    color: white;  
    background: rgb(0,102,153);  
    background: linear-gradient(to bottom, rgba(255,255,255,.1) 40%,  
    rgba(255,255,255,.5));  
}
```

С помощью данного кода установлен стандартный цвет фона (для Internet Explorer 9 и более ранних версий), а затем — линейный градиент. В нем заданы два цветовых узла. Первый цвет простирается от верхнего края до 40 % внутрь кнопки, затем начинается плавный переход цвета (градиенты и цветовые узлы рассматриваются в разделе «Использование градиентных фонов» главы 8).

И наконец, еще немного скорректируйте внешний вид. Удалите стандартную рамку, скруглите углы и задайте эффект свечения вокруг блока.

11. Отредактируйте стиль кнопки **Subscribe** в последний раз (изменения выделены полужирным шрифтом):

```
.subform input[type="submit"] {  
    margin-left: 220px;  
    padding: 10px 25px;  
    font-size: 1em;  
    color: white;  
    background: rgb(0,102,153);  
    background: linear-gradient(to bottom, rgba(255,255,255,.1) 40%,  
    rgba(255,255,255,.5));
```

```
border-radius: 5px;
box-shadow: 0 0 4px white;
}
```

Присвоение свойству `border` значения `none` удаляет рамку, которую браузер обычно визуализирует вокруг кнопки, а свойство `border-radius` скругляет углы кнопки. И наконец, добавление тени без горизонтального или вертикального смещения (имеется в виду часть `0 0`) позволяет получить эффект свечения элемента, которое выглядит как легкая белая подсветка, исходящая из-под кнопки.

#### ПРИМЕЧАНИЕ

Усовершенствование дизайна кнопки можно продолжить. Создайте эффект ролловера — `.subform input[type="submit"]:hover` — и измените фоновый цвет. Используя материал предыдущей главы, вы можете этот переход даже анимировать!

Теперь текстовые метки и кнопка **Subscribe** смотрятся великолепно, но зачем на этом останавливаться? Настало время приукрасить элементы веб-формы. Начнем с изменения их шрифта и фоновой окраски.

#### 12. Отформатируйте раскрывающийся список:

```
.subform select {
  font-size: 1.2em;
}
```

Этот стиль немного укрупняет текст. Можно выбрать шрифт, добавить цвет фона и внести другие изменения. Но некоторые браузеры (например, Safari) не позволяют вносить существенные изменения в форматирование раскрывающихся списков, поэтому все внесенные в них изменения стилей нужно тестировать.

Теперь пришла пора внести изменения в текстовые поля.

#### 13. Создайте новый групповой селектор для трех текстовых полей:

```
.subform input[type="text"], .subform textarea {
  border-radius: 5px;
  border: none;
  background-color: rgba(255,255,255,.5);
  color: rgba(255,255,255,1);
  font-size: 1.2em;
  box-shadow: inset 0 0 10px rgba(255,255,255,.75);
}
```

Этот групповой стиль выбирает все элементы `input`, имеющие тип `text`, а также многострочные текстовые области (элемент `textarea`). В нем применяются различные свойства, с которыми вы уже должны быть знакомы, такие как `border-radius`, `background-color`, `font-size` и `box-shadow`. Текстовые поля выглядят мелко, поэтому для них устанавливается ширина и к ним добавляются небольшие отступы.

#### 14. Отредактируйте только что созданный стиль, изменив ширину и применив отступы (изменения выделены полужирным шрифтом):



```
.subform input[type="text"], .subform textarea {
  border-radius: 5px;
  border: none;
  background-color: rgba(255,255,255,.5);
  color: rgba(255,255,255,1);
  font-size: 1.2em;
  box-shadow: inset 0 0 10px rgba(255,255,255,.75);
  width: 500px;
  padding: 5px;
}
```

Вы можете сделать форму более удобной для пользователей, выделив активные элементы с помощью специального псевдокласса `:focus` (см. раздел «Выборка форматировемых ссылок» главы 9). Мы добавим его на следующем шаге.

15. В конце внутренней таблицы стилей добавьте стиль для раскрывающегося списка и трех текстовых полей:

```
.subform input[type="text"]:focus, .subform textarea:focus {
  background-color: white;
  color: black;
}
```

Просмотрите страницу в браузере. Сейчас она должна выглядеть так, как показано на рис. 11.12. Готовую версию созданной в этом уроке веб-формы вы можете найти в папке `11_finished\form`.

**Рис. 11.12.** Используя псевдокласс `:focus`, вы можете сделать веб-форму более удобной для пользователя: будет подсвечиваться активный элемент веб-формы. Здесь можно увидеть, что все готово к выбору пункта из раскрывающегося списка, поскольку он имеет белый цвет фона