

Лабораторне заняття №6

з навчальної дисципліни WEB-Дизайн

на тему:

Основи CSS

Навігація по HTML-сторінці

За допомогою текстового редактора в папці **lab_6/** створюємо текстовий документ з назвою **nav_bar.html** з відповідним html-КОДОМ:

На этом практическом занятии мы превратим простой список ссылок во впечатляющую панель навигации с эффектом-ролloverом и выделением нажатой кнопки текущего раздела сайта.

1. Откройте файл веб-страницы `nav_bar.html` из папки `09\nav_bar` в HTML-редакторе.

Как видите, в этом файле содержится совсем короткий программный код. Здесь присутствует внутренняя таблица со сбросом стандартных стилей CSS и одним правилом, определяющим простейшее форматирование тега `<body>` основного содержимого веб-страницы. Код HTML создает маркированный список, состоящий из шести ссылок. Веб-страница имеет вид, показанный на рис. 9.13, 1. Первым шагом будет добавление HTML-кода, обеспечивающего целенаправленное форматирование ссылок списка средствами CSS.

2. Найдите открывающий тег `` и добавьте в него `class = "mainNav"`, чтобы он выглядел следующим образом:

```
<ul class="mainNav">
```

Атрибут `class` назначает этот список главной областью навигации, или панелью управления. Мы будем использовать его для создания селекторов потомков, избирательно форматирующих только ссылки этого списка (а не все ссылки веб-страницы).

3. Добавьте во внутреннюю таблицу после `body` новый стиль:

```
.mainNav {  
  margin: 0;  
  padding: 0;
```

```
list-style: none;
}
```

Этот стиль будет применен лишь к тегу `` с классом `mainNav`. Он удаляет отступ и маркеры, которые браузер добавляет в маркированные списки, как показано на рис. 9.13, 2. Теперь приступим к форматированию ссылок.

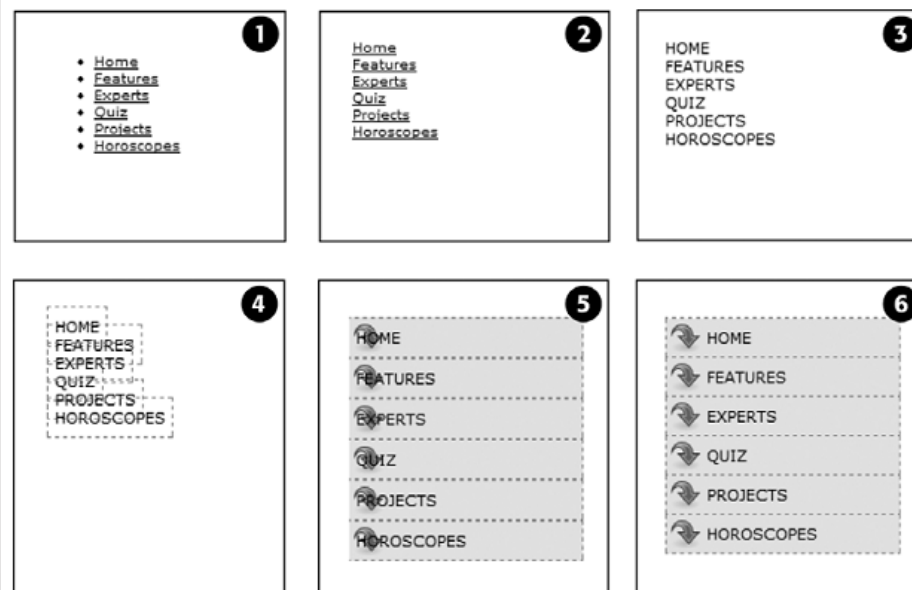


Рис. 9.13. Вам может показаться, что преобразование простого списка в сложную панель навигации требует выполнения объемной работы, но на самом деле нужно всего лишь создать несколько стилей

4. Добавьте производный селектор для форматирования ссылок списка:

```
.mainNav a {
  color: #000;
  font-size: 11px;
  text-transform: uppercase;
  text-decoration: none;
}
```

Этот стиль определяет базовое форматирование текста ссылок: устанавливает цвет и размер шрифта, верхний регистр всех букв, удаляет линию подчеркивания (см. рис. 9.13, 3). Сейчас придадим ссылкам вид кнопок.

5. Добавьте в стиль `.mainNav` свойства `border` и `padding`:

```
border: 1px dashed #999;
padding: 7px 5px;
```

Теперь при просмотре веб-страницы в браузере вы заметите пару проблем (см. рис. 9.13, 4): границы перекрываются между собой, а ссылки имеют различную ширину. Это происходит потому, что тег `<a>` является линейным элементом

и ширина ссылки равна длине ее текста. Кроме того, верхний и нижний отступы не обеспечивают увеличения высоты линейным элементам, поэтому границы накладываются друг на друга (о линейных элементах читайте в разделе «Управление размерами полей и отступов» гл. 7). Решить эти проблемы со ссылками можно путем изменения их способа отображения браузером.

6. Добавьте свойство `display: block;` в стиль `.mainNav`.

Мы изменяем параметры отображения тега `<a>` таким образом, чтобы он обрабатывался браузером как абзац текста или другой блочный элемент, с расположением ссылок в виде списка, в точности одна над другой. Осталась единственная проблема — ссылки растянуты по всей ширине окна браузера — слишком много для кнопок. Исправим ситуацию, ограничив ширину тега `` в соответствующем стиле.

7. Во внутренней таблице найдите стиль `.mainNav` и добавьте в него свойство `width: 175px;`.

```
.mainNav {
  margin: 0;
  padding: 0;
  list-style: none;
  width: 175px;
}
```

При ширине списка 175 пикселей ссылки по-прежнему слишком широки, хотя и ограничиваются шириной элемента-контейнера (тег ``). В большинстве случаев список ссылок заключен в какой-то элемент разметки веб-страницы (например, боковая панель), для которого уже определена ширина, поэтому можете пропустить этот шаг (о том, как создавать боковые панели, читайте в части 3).

Сейчас перейдем к самому интересному.

8. Добавьте в стиль `.mainNav` свойство `background` следующим образом:

```
.mainNav a {
  color: #000;
  font-size: 11px;
  text-transform: uppercase;
  text-decoration: none;
  border: 1px dashed #999;
  padding: 7px 5px;
  display: block;
  background-color: #E7E7E7;
  background-image: url(images/nav.png);
  background-repeat: no-repeat;
  background-position: 0 2px;
}
```

Эти строки изменяют цвет фона ссылок на серый и устанавливают одиночное изображение с левой стороны каждой кнопки (см. рис. 9.13, 5). Здесь тоже нужно кое-что исправить: текст ссылки накладывается на значок, а линии границ между кнопками имеют толщину, равную 2 пикселям (теоретически границы имеют толщину 1 пиксел, но сливающиеся воедино линии соседних ссылок образуют линию толщиной 2 пиксела).

9. Удалите верхнюю границу и измените отступ в стиле `.mainNav`, чтобы он выглядел следующим образом:

```
.mainNav a {  
  color: #000;  
  font-size: 11px;  
  text-transform: uppercase;  
  text-decoration: none;  
  border: 1px dashed #999;  
  border-bottom: none;  
  padding: 7px 5px 7px 30px;  
  display: block;  
  background-color: #E7E7E7;  
  background-image: url(images/nav.png);  
  background-repeat: no-repeat;  
  background-position: 0 2px;  
}
```

Выглядит неплохо: текст каждой ссылки отделен от значка, границы выделяются. Однако нижняя граница последней ссылки исчезла. Существует несколько способов разобраться с этим. Один состоит в том, чтобы создать стилевой класс с надлежащим параметром нижней границы `border-bottom`, а затем применить его к этой ссылке. Будет гораздо проще применить нижнюю границу к тегу ``, содержащему список ссылок (поскольку этот тег не имеет отступов, нет промежутка, отделяющего верх `` от верхней стороны этой первой ссылки).

10. Добавьте верхнюю границу в стиль `.mainNav`, чтобы он выглядел следующим образом:

```
.mainNav {  
  margin: 0;  
  padding: 0;  
  list-style: none;  
  width: 175px;  
  border-bottom: 1px dashed #999;  
}
```

Вот и все, мы создали простейшую панель навигации с применением границ, отступов, фонового цвета и изображений (см. рис. 9.13, б).

1. Добавьте в конце таблицы стилей файла `nav_bar.html` следующий стиль:

```
.mainNav a:hover {  
    font-weight: bold;  
    background-color: #B2F511;  
    background-position: 3px 50%;  
}
```

Он определяет внешний вид ссылки-кнопки в состоянии `hover`. Стиль изменяет шрифт текста кнопки с обычного на полужирный, а также цвет фона на ярко-зеленый. Кроме того, он использует метод CSS-спрайтов, упоминавшийся ранее. То же самое изображение применяется в шаге 8 в прошлом разделе — оно в действительности содержит три различных значка (рис. 9.14). В этом случае изображение центрируется внутри кнопки, отображая средний значок файла.

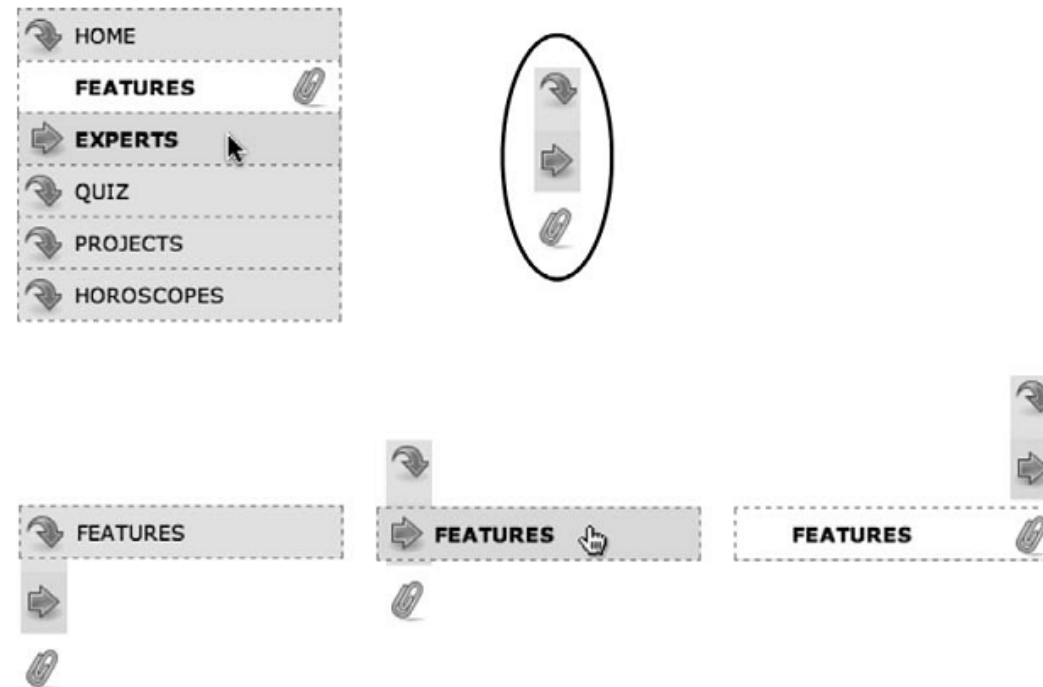


Рис. 9.14. С помощью несложного CSS-кода можно создать интерактивный эффект ролловера для кнопок панели навигации, а также автоматически подсветить раздел сайта текущей страницы

Теперь при наведении указателя мыши на любую кнопку и его перемещении на ней кнопка моментально изменяет свой вид (откройте веб-страницу в своем браузере и посмотрите сами).

2. Найдите тег `<body>` и добавьте в него класс `class="home"`:

```
<body class="home">
```

Теперь, когда мы знаем, какому разделу сайта принадлежит текущая открытая страница, можно использовать селектор потомка для создания конкретного CSS-стиля, который будет применен к веб-странице раздела **Features (Особенности)**. Далее мы должны маркировать разделы, на которые указывает каждая ссылка-кнопка, что достигается путем добавления классов.

3. Найдите в HTML-коде панели навигации ссылку **Home (Домой)** и добавьте в нее класс `class="homeLink"`, чтобы тег выглядел следующим образом:

```
<a href="/index.html" class="homeLink">Home</a>
```

Этот класс однозначно определяет ссылку, предоставляя возможность создания стиля, который будет применен только к ней.

Естественно, нам потребуется добавить классы для всех ссылок панели навигации.

4. Повторите шаг 3 для каждой из ссылок, используя следующие классы: `featureLink`, `expertLink`, `quizLink`, `projectLink` и `horoscopeLink`.

С HTML-частью закончили. Настало время создать CSS-код. Теперь у нас есть маркированные страница и ссылка, и мы можем элементарно создать производный селектор, выделяющий кнопку-ссылку **Features (Особенности)**.

5. Добавьте в таблицу стилей веб-страницы еще один стиль:

```
.home .homeLink {  
  background-color: #FFFFFF;  
  background-position: 97% 100%;  
  padding-right: 15px;  
  padding-left: 30px;  
  font-weight: bold;  
}
```

Мы уже рассматривали эти свойства прежде. Мы снова используем метод `CSS Styles` для регулировки позиции фонового изображения. На этот раз изображение отодвигается на 97 % вправо, а его нижняя часть помещается в нижнюю часть кнопки. Другими словами, значок будет отображаться внизу изображения (см. рис. 9.14). О том, как процентные значения работают с фоновыми изображениями, мы говорили в гл. 8.

6. Отредактируйте селектор только что созданного стиля:

```
.home .homeLink,  
.feature .featureLink,  
.expert .experLink,  
.quiz .quizLink,  
.project .projectLink,  
.horoscope .horoscopeLink  
{  
  background-color: #FFFFFF;  
  background-position: 97% 100%;  
  padding-right: 15px;  
  padding-left: 30px;  
  font-weight: bold;  
}
```

7. Измените атрибут class тега <body> на feature следующим образом:

```
<body class="feature">
```

1. Убедитесь в том, что вы выполнили все шаги по созданию вертикальной панели навигации, и откройте файл `nav_bar.html` в HTML-редакторе.

Сейчас вы увидите, как просто изменить ориентацию панели. Сначала подчистим кое-что из того, что мы уже сделали. Нужно удалить ширину, которую вы установили для тега `` ранее. Эта ширина препятствовала тому, чтобы навигационные кнопки охватывали всю длину страницы. Но поскольку тегу `` необходимо расширяться, чтобы вмещать набор горизонтальных кнопок, эта ширина подойдет нам без изменений.

2. Найдите стиль `.mainNav` и удалите в нем свойство `width: 175px;`.

Теперь раскрываю сам секрет преобразования вертикальной панели навигации в горизонтальную.

3. Добавьте в таблицу новый стиль (сразу после `.mainNav`):

```
.mainNav li {  
  float: left;  
  width: 12em;  
}
```

Этот стиль применяется к тегу `` (представляющему собой элементы списка, каждый из которых содержит свою ссылку). Первая команда устанавливает для тега выравнивание по левому краю. Таким образом, каждый последующий тег `` располагается с правой стороны предыдущего (такой же эффект вы могли наблюдать в обучающем уроке по созданию фотогалереи в гл. 8). Кроме того, устанавливая ширину тега ``, мы одновременно определяем ширину для всех кнопок панели навигации. В данном случае значение `12em` обеспечивает достаточный размер, чтобы вместить самый длинный текст названия ссылки. Если будет необходимо увеличить длину, вы должны будете увеличить это значение.

При просмотре веб-страницы в браузере вы увидите, что с панелью навигации в основном все в порядке. Требуются только небольшие корректировки (смотрите обведенные кружком области на рис. 9.15, 1). Нижняя граница, созданная нами в шаге 10 выше, занимает всю ширину тега ``, более широкого, чем кнопки, непосредственно образующие панель навигации. Еще более странно то, что нижняя граница расположена не на заднем плане, а поверх кнопок. Кроме того, поскольку последние размещены рядом друг с другом, их левые и правые границы сливаются между собой, образуя линии толщиной 2 пиксела. Сейчас мы решим эти проблемы.

4. В стиле `.mainNav` а измените свойство `border-bottom: none;` на `border-left: none;`.

Это действие приводит к удалению левых границ кнопок, чтобы они не удваивались, и в то же время к нижнему краю добавляется граница. Однако нижняя граница тега `` по-прежнему отображается поверх кнопок, и с левой стороны панели навигации отсутствует граница на крайней левой кнопке (смотрите обведенные кружком области на рис. 9.15, 2). Никаких проблем — просто измените границы ``.

5. Найдите стиль `.mainNav` и измените в нем свойство `border-bottom: 1px dashed #999999;` на `border-left: 1px dashed #999999;`.

Сейчас при просмотре веб-страницы в браузере вы заметите, что граница над кнопками исчезла, но с левой стороны панели навигации так и не появилась (см. рис. 9.15, 3). Это еще одно свидетельство сложности использования плавающих элементов. Они вышли из обычного потока содержимого документа, и браузер больше не видит их в качестве составной части тега ``. Тег уменьшился до минимальных размеров, именно поэтому его нижняя граница отображается поверх остального содержимого (это может показаться слишком запутанным, но так оно и есть, именно поэтому работе с плавающими элементами посвящена отдельная глава книги — см. гл. 13).

К счастью, несмотря на то, что проблема кажется достаточно сложной, ее решение совсем простое. Добавим одно CSS-свойство к маркированному списку.

6. Добавьте два свойства в конце стиля `.mainNav`:

```
.mainNav {  
  margin: 0;  
  padding: 0;  
  list-style: none;  
  border-left: 1px dashed #999999;  
  overflow: hidden;  
}
```

Код `overflow: hidden` заставляет расширяться неупорядоченный список. Почему свойство работает? Об этом вы узнаете из гл. 12. Наконец, значок скрепки, выровненный по правому краю нажатой кнопки Quiz, выглядит неплохо (см. рис. 9.15, 4), но лучше позиционировать его по левому краю.

7. Найдите стиль выбранной кнопки, созданный в шаге 6 (это стиль с длинным селектором). Измените координаты его свойства `background` с `97% 100%` на `3px 100%`. Теперь стиль должен выглядеть следующим образом:

```
.home .homeLink,  
.feature .featureLink,  
.expert .experLink,  
.quiz .quizLink,  
.project .projectLink,  
.horoscope .horoscopeLink  
{  
  background-color: #FFFFFF;  
  background-position: 3px 100%;  
  padding-right: 15px;  
  padding-left: 30px;  
  font-weight: bold;  
}
```

Лекцію закінчено.
Дякую за увагу!