**Глава 7. Навигационные панели с помощью CSS**

Навигация присутствует на любом хорошем сайте, даже если он одностраничный. В зависимости от ситуации, навигационные ссылки могут вести на различные разделы сайта либо отправлять к закладке (якорю), расположенной на текущей странице. Грамотное составление навигации, в которой не запутается пользователь, требует определенных знаний и опыта. К дизайну навигационной панели также нужно подходить с умом, и в этом уроке мы расскажем вам, как создать удобное навигационное меню.

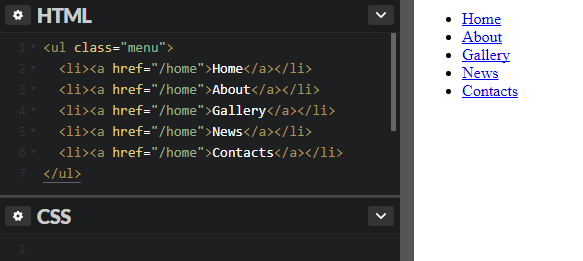
Создание навигации

Что такое навигация? Это набор ссылок, зачастую упорядоченных и сгруппированных по смыслу. Навигационное меню часто создается с использованием HTML-тега списка <ul>, где в каждом пункте <li> содержится одна ссылка <a>. В HTML5 для навигации ввели отдельный тег <nav>, куда можно просто помещать теги ссылок. После создания HTML-каркаса можно переходить к его стилизации с помощью CSS, где вы можете определять, каким будет ваше меню — вертикальным, горизонтальным, выпадающим и т. п.

Допустим, у нас есть простая навигация с пятью ссылками, созданная в HTML на основе маркированного списка:

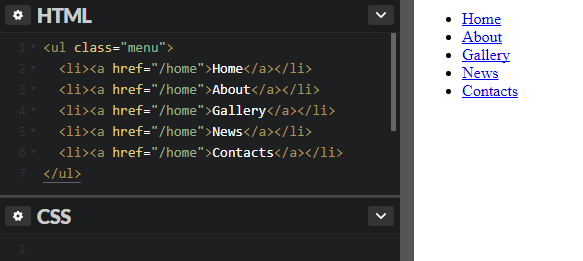
|  |
| --- |
| <ul class="menu">  <li><a href="#">Home</a></li>  <li><a href="#">About</a></li>  <li><a href="#">Gallery</a></li>  <li><a href="#">News</a></li>  <li><a href="#">Contacts</a></li>  </ul> |

Изначально, без стилей наша навигация выглядит, как обычный список:



Чтобы сделать этот элемент более похожим на навигационную панель, необходимо проделать несколько обязательных шагов. Во-первых, возле пунктов меню не нужны маркеры, а во-вторых, стандартные значения margin и padding, заданные браузером для списка, могут мешать, поэтому мы обнуляем их. В итоге получается следующее:

|  |
| --- |
| .menu {  list-style-type: none;  margin: 0;  padding: 0;  } |



По сути, идентичные действия по отношению к спискам делает файл сброса стилей *Reset.css*, поэтому, если вы используете его на вашем сайте, то код выше можно не записывать.

После обнуления стандартных стилей списка можно приступать непосредственно к созданию стилей для навигации. Мы покажем, как делать вертикальные и горизонтальные панели.

**Вертикальное меню**

Создание вертикальной навигационной панели считается более легким занятием, нежели создание горизонтального меню. Главным образом потому, что нет необходимости задавать горизонтальное расположение ссылок. Это просто вертикальный список страниц сайта. И всё же некоторые важные стили применить придется.

Блочные ссылки

Прежде всего нужно сделать все теги <a> блочными элементами:

.menu a { display: block; }

Существует несколько причин для этого:

Вы сможете задавать отступы для ссылок.

Блочный элемент занимает всю ширину контейнера, тогда как ширина строчного элемента зависит от его содержимого. Из-за этого могут появиться проблемы, если вы, например, решите установить для ссылок фоновый цвет:

|  |
| --- |
| .menu {  list-style-type: none;  margin: 0;  padding: 0;  }  .menu a{  background-color:yellow;  } |

Строчные ссылки могут быть разной длины, что негативно отразится на внешнем виде.

Исходя из второго пункта, можно отметить еще один положительный момент. Когда ссылка является блочным элементом, область ее кликабельности расширяется: даже если кликнуть по месту, где нет текста, она все равно сработает. Это удобно, потому что не надо прицеливаться по маленькой ссылке. А с помощью вертикальных отступов padding можно увеличить область кликабельности еще и по высоте.

**Ширина меню**

Следующее, что нам понадобится сделать — это задать ширину меню. Если у вас заранее не была определена ширина навигационной панели, и она не находится в контейнере, который ограничивает ее, то меню будет растянуто на всю ширину окна (как в нашем примере выше) — ведь блочный элемент по умолчанию ведет себя именно так. Вы можете задать необходимую ширину для элемента .menu, например:

|  |
| --- |
| .menu {  list-style-type: none;  margin: 0;  padding: 0;  width: 300px;  }  .menu a{  background-color:yellow;  display:block;  } |

Вместо этого можно установить ширину для пунктов <li> либо для ссылок <a> — визуально эффект может быть одинаковый, но учтите, что в таком случае ширина элемента .menu всё равно останется 100%.

**Разделение пунктов**

Чтобы визуально разделить пункты меню, можно задать каждому из них нижнюю или верхнюю границу. В зависимости от того, какую из них вы используете, у вас будет не хватать границы сверху или снизу меню. Это можно исправить, добавив еще одну границу к самому блоку .menu:

|  |
| --- |
| .menu a {  border-top: 1px solid blue;  }  .menu {  border-bottom: 1px solid blue;  } |

т.е.

|  |
| --- |
| .menu {  list-style-type: none;  margin: 0;  padding: 0;  width: 300px;  border-bottom: 1px solid blue;  }  .menu a{  background-color:yellow;  border-top: 1px solid blue;  display:block;  } |

**Высота пунктов и вертикальное выравнивание**

Сейчас ссылки меню выглядят слишком низкими, расположенными близко друг к другу. Если текст каждой ссылки не будет занимать более одной строки, то увеличить их высоту и сохранить при этом текст вертикально по центру можно с помощью свойств height и line-height:

|  |
| --- |
| .menu a {  height: 30px;  line-height: 30px;  } |

Т.е.

|  |
| --- |
| .menu {  list-style-type: none;  margin: 0;  padding: 0;  width: 300px;  border-bottom: 1px solid blue;  }  .menu a{  background-color:yellow;  border-top: 1px solid blue;  display:block;  line-height: 30px;  } |

Как видно на анимации, свойство height влияет на высоту ссылки, а межстрочный интервал line-heightизменяет уже высоту самой текстовой строки. Обязательно задавайте одинаковые значения для этих двух свойств, если хотите, чтобы текст ссылки был вертикально отцентрирован.

**Горизонтальное меню**

Основная задача при создании горизонтальной навигации — расположить пункты меню в один ряд. Это можно сделать несколькими способами.

**Способ первый**

Первый вариант — задать пунктам списка свойство display со значением inline либо inline-block:

|  |
| --- |
| .menu li {  display: inline;  } |

|  |
| --- |
| .menu {  list-style-type: none;  margin: 0;  padding: 0;  width: 300px;  border-bottom: 1px solid blue;  }  .menu a{  background-color:yellow;  border-top: 1px solid blue;  line-height: 30px;  text-decoration:none;  }  .menu li{  display:inline;  } |

При этом теги <a> внутри <li> нельзя делать блочными, иначе пункты перестанут выстраиваться горизонтально и снова встанут в вертикальную позицию. Ссылкам можно также задать значение inline или inline-block. Выберите второе, если вы собираетесь в дальнейшем создавать отступы для ссылок.

После этого можно заняться оформлением ссылок. Например, можно сделать такой стиль:

|  |
| --- |
| .menu a {  text-decoration: none;  font-family: sans-serif;  color: #5757a0;  display: inline-block;  padding: 10px 20px;  background-color: lavender;  border-bottom: 5px solid #5757a0;  } |

|  |
| --- |
| .menu {  list-style-type: none;  margin: 0;  padding: 0;  width: 350px;  border-bottom: 1px solid blue;  }  .menu a {  text-decoration: none;  font-family: sans-serif;  color: #5757a0;  display: inline-block;  padding: 10px 10px;  background-color: lavender;  border-bottom: 5px solid #5757a0;  }  .menu li{  display:inline;  } |

Мы добавили фоновый цвет и нижнюю границу для каждой ссылки, а также увеличили ее размеры при помощи внутренних отступов padding. Заметьте, что между пунктами меню есть небольшие промежутки, хоть мы и не добавляли их через стили. Они возникают, потому что браузер так воспринимает символ возврата каретки между закрывающим и открывающим тегами <li>. Эти промежутки можно убрать, записав HTML-код списка в одной строке:

|  |
| --- |
| <ul class="menu">  <li><a href="#">Home</a></li>  <li><a href="#">About</a></li>  <li><a href="#">Gallery</a></li>  ...  </ul> |

Однако это неудобно, ведь эти изменения придется вносить в HTML вручную, да и чтение кода затрудняется. Поэтому попробовать избавиться от промежутков можно с помощью отрицательного значения margin-right:

|  |
| --- |
| .menu li {  margin-right: -5px;  } |

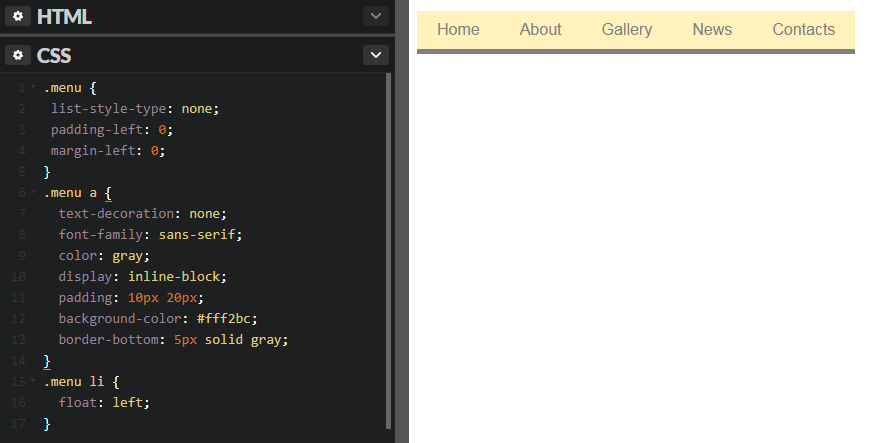
Этот код не убирает промежутки, а прикрывает их за счет сдвига элементов на пять пикселей влево. Данный способ нельзя назвать очень надежным, потому что размер промежутка может варьироваться в зависимости от других стилей.

Если промежутки между пунктами не запланированы в вашем дизайне, лучше воспользоваться вторым методом горизонтального выстраивания элементов.

**Способ второй**

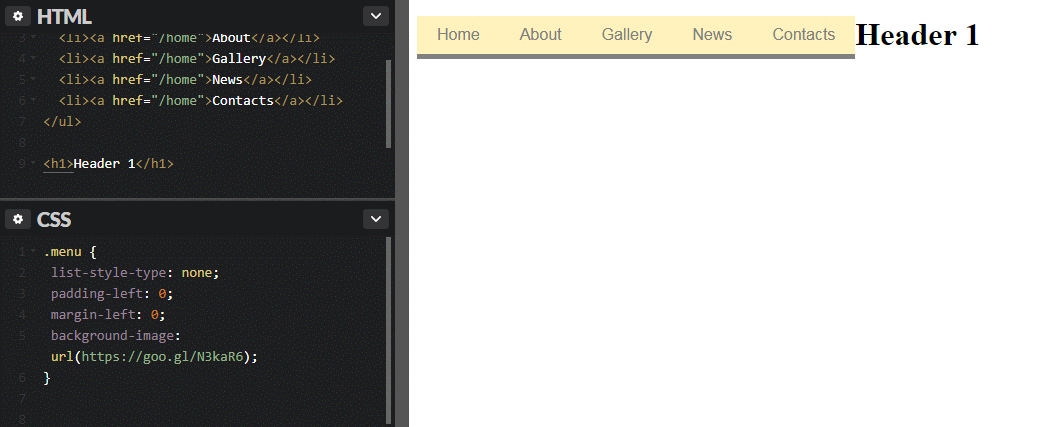
Возможно, вы уже догадались, что второй вариант создания горизонтального меню заключается в использовании свойства float. Для этого добавьте стиль для тегов <li>:

|  |
| --- |
| .menu li {  float: left;  } |

  
 Как видим, промежутков между пунктами уже нет. Все остальные стили работают, как и прежде. Но если вы добавите после навигации другой HTML-элемент (допустим, <h1>), то на него повлияет обтекание и он встанет в один ряд с пунктами меню. Отменить это действие можно, добавив заголовку свойство clear: left. Но при разработке сайта с большим количеством разных страниц может стать сложно следить за тем, нигде ли вы не забыли указать данное свойство. Гораздо легче отменить обтекание со стороны навигации. В этом поможет такая запись:

|  |
| --- |
| .menu {  overflow: hidden;  } |

Это также поможет вернуть нормальную высоту родителя плавающих элементов. Мы уже говорили об этой проблеме в предыдущих уроках, но тогда мы решали ее через специальный хак с использованием псевдоэлемента :after. Пример выше — это еще один способ решить вопрос с исчезнувшей высотой контейнера.



**Примечание:** если вы установите фон для элемента .menu, то вы не увидите его до тех пор, пока не примените к нему overflow: hidden. Логично, ведь как можно увидеть фон элемента, чья высота равна нулю?

|  |
| --- |
| .menu {  list-style-type: none;  margin: 0px;  padding: 0px;  width:550px;  border-bottom: 5px solid red;  overflow:hidden;  }  .menu a {  text-decoration: none;  font-family: sans-serif;  color: #5757a0;  display: inline-block;  padding: 10px 20px;  background-color: lavender;  border-bottom: 5px solid #5757a0;  font-size:25px;  }  .menu li{  float:left;  margin-right:-5px;  }  <html>  <head>  <title>  My first Site  </title>  <link rel="stylesheet" href="style.css">  </head>  <body>  <ul class="menu">  <li><a href="#">Home</a></li>  <li><a href="#">About</a></li>  <li><a href="#">Gallery</a></li>  <li><a href="#">News</a></li>  <li><a href="#">Contacts</a></li>  </ul>  <H1> My Title Site </H1>  </body>  </html> |

**CSS меню**

Если ваш веб-сайт не ограничивается одной веб-страницей, то стоит подумать о добавлении панели навигации (меню). Меню - раздел веб-сайта, предназначенный помогать посетителю перемещаться по сайту. Любое меню представляет собой список ссылок, ведущих на внутренние страницы сайта. Самым простым способом добавить панель навигации на сайт является создание меню с помощью CSS и HTML.

**Вертикальное меню**

Первым шагом создания вертикального меню будет создание маркированного списка. Также нам нужно будет иметь возможность идентифицировать список, поэтому мы добавим к нему атрибут id с идентификатором "navbar". Каждый элемент <li> нашего списка будет содержать по одной ссылке:

|  |
| --- |
| <ul id="navbar">  <li><a href="#">Главная</a></li>  <li><a href="#">Новости</a></li>  <li><a href="#">Контакты</a></li>  <li><a href="#">О нас</a></li>  </ul> |

Наша следующая задача заключается в сбросе стилей списка, установленных по умолчанию. Нам нужно убрать внешние и внутренние отступы у самого списка и маркеры у пунктов списка. Затем зададим нужную ширину:

|  |
| --- |
| #navbar {  margin: 0;  padding: 0;  list-style-type: none;  width: 100px;  } |

Теперь пришло время стилизовать сами ссылки. Мы добавим к ним фоновый цвет, изменим параметры текста: цвет, размер и насыщенность шрифта, уберем подчеркивание, добавим небольшие отступы и переопределим отображение элемента <a> со строчного на блочный. Дополнительно были добавлены левая и нижняя рамки к пунктам списка.

Самой важной частью наших изменений является переопределение строчных элементов на блочные. Теперь наши ссылки занимают все доступное пространство пунктов списка, то есть для перехода по ссылке нам больше не нужно наводить курсор точно на текст.

|  |
| --- |
| #navbar a {  background-color: #949494;  color: #fff;  padding: 5px;  text-decoration: none;  font-weight: bold;  border-left: 5px solid #33ADFF;  display: block;  }  #navbar li {  border-left: 10px solid #666;  border-bottom: 1px solid #666;  } |

Мы объединили весь код, описанный выше, в один пример, теперь, нажав на кнопку попробовать, вы можете перейти на страницу с примером и увидеть результат:

|  |
| --- |
| <!DOCTYPE html>  <html>  <head>  <meta charset="utf-8">  <title>Название документа</title>  <style>  #navbar {  margin: 0;  padding: 0;  list-style-type: none;  width: 100px;  }  #navbar li {  border-left: 10px solid #666;  border-bottom: 1px solid #666;  }  #navbar a {  background-color: #949494;  color: #fff;  padding: 5px;  text-decoration: none;  font-weight: bold;  border-left: 5px solid #33ADFF;  display: block;  }  </style>  </head>    <body>    <ul id="navbar">  <li><a href="#">Главная</a></li>  <li><a href="#">Новости</a></li>  <li><a href="#">Контакты</a></li>  <li><a href="#">О нас</a></li>  </ul>    </body>  </html> |

При наведении курсора мыши на пункт меню его внешний вид может изменяться, привлекая к себе внимание пользователя. Создать такой эффект можно с помощью псевдо-класса :hover.

Вернемся к рассмотренному ранее примеру вертикального меню и добавим в таблицу стилей следующее правило:

|  |
| --- |
| #navbar a:hover {  background-color: #666;  border-left: 5px solid #3333FF;  } |

Польный код:

|  |
| --- |
| #navbar {  margin: 0;  padding: 0;  list-style-type: none;  width: 100px;  }  #navbar a {  background-color: #949494;  color: #fff;  padding: 5px;  text-decoration: none;  font-weight: bold;  border-left: 5px solid #33ADFF;  display: block;  }  #navbar li {  border-left: 10px solid #666;  border-bottom: 1px solid #666;  }  #navbar a:hover {  background-color: #666;  border-left: 5px solid #3333FF;  }  <html>  <head>  <title>  My first Site  </title>  <link rel="stylesheet" href="style.css">  </head>  <body>  <ul id="navbar">  <li><a href="#">Главная</a></li>  <li><a href="#">Новости</a></li>  <li><a href="#">Контакты</a></li>  <li><a href="#">О нас</a></li>  </ul>  </body>  </html> |

**Горизонтальное меню**

В предыдущем примере мы рассмотрели вертикальную панель навигации, которую чаще всего можно встретить на сайтах слева или справа от области с основным контентом. Однако меню с навигационными ссылками также часто располагается и по горизонтали в верхней части веб-страницы.

Горизонтальное меню можно создать путем стилизации обычного списка. Свойству display для элементов <li> нужно присвоить значение inline, чтобы пункты списка располагались друг за другом.

Для размещения пунктов меню по горизонтали, сначала создадим маркированный список с ссылками:

|  |
| --- |
| <ul id="navbar">  <li><a href="#">Главная</a></li>  <li><a href="#">Новости</a></li>  <li><a href="#">Контакты</a></li>  <li><a href="#">О нас</a></li>  </ul> |

Напишем для нашего списка пару правил, сбрасывающих стиль используемый для списков по умолчанию, и переопределим пункты списка с блочных на строчные:

|  |
| --- |
| #navbar {  margin: 0;  padding: 0;  list-style-type: none;  }  #navbar li { display: inline; } |

Теперь нам осталось лишь определить стилевое оформление для нашего горизонтального меню:

|  |
| --- |
| #navbar {  margin: 0;  padding: 0;  list-style-type: none;  border: 2px solid #0066FF;  border-radius: 20px 5px;  width: 550px;  text-align: center;  background-color: #33ADFF;  }  #navbar a {  color: #fff;  padding: 5px 10px;  text-decoration: none;  font-weight: bold;  display: inline-block;  width: 100px;  }  #navbar a:hover {  border-radius: 20px 5px;  background-color: #0066FF;  } |

Полбный код:

|  |
| --- |
| #navbar {  margin: 0;  padding: 0;  list-style-type: none;  border: 2px solid #0066FF;  border-radius: 20px 5px;  width: 550px;  text-align: center;  background-color: #33ADFF;    }  #navbar a {  color: #fff;  padding: 5px 10px;  text-decoration: none;  font-weight: bold;  display: inline-block;  width: 100px;  }  #navbar a:hover {  border-radius: 20px 5px;  background-color: #0066FF;  }  #navbar li{  display: inline;  }  <html>  <head>  <title>  My first Site  </title>  <link rel="stylesheet" href="style.css">  </head>  <body>  <ul id="navbar">  <li><a href="#">Главная</a></li>  <li><a href="#">Новости</a></li>  <li><a href="#">Контакты</a></li>  <li><a href="#">О нас</a></li>  </ul>  </body>  </html> |

**Выпадающее меню**

Меню, которое мы будем создавать, будет иметь основные навигационные ссылки, расположенные в горизонтальной панели навигации, и подпункты, которые будут отображаться только после наведения курсора мыши на тот пункт меню, к которому эти подпункты относятся.

Сначала нам нужно создать HTML-структуру нашего меню. Основные навигационные ссылки мы поместим в маркированный список:

|  |
| --- |
| <ul id="navbar">  <li><a href="#">Главная</a></li>  <li><a href="#">Новости</a></li>  <li><a href="#">Контакты</a></li>  <li><a href="#">О нас</a></li>  </ul> |

Подпункты мы разместим в отдельном списке, вложив его в элемент <li>, который содержит родительскую ссылку относительно подпунктов. Теперь мы имеем четкую структуру нашей будущей панели навигации:

|  |
| --- |
| <ul id="navbar">  <li><a href="#">Главная</a></li>  <li><a href="#">Новости</a></li>  <li><a href="#">Контакты</a>  <ul>  <li><a href="#">Адрес</a></li>  <li><a href="#">Телефон</a></li>  <li><a href="#">Email</a></li>  </ul>  </li>  <li><a href="#">О нас</a></li>  </ul> |

Теперь приступим к написанию CSS кода. Для начала необходимо скрыть список с подпунктами с помощью объявления display: none;, чтобы они не отображались на веб-странице все время. Для отображения подпунктов нам нужно чтобы при наведении на элемент <li> список снова был преобразован в блочный элемент:

|  |
| --- |
| #navbar ul { display: none; }  #navbar li:hover ul { display: block; } |

Убираем у обоих списков отступы и маркеры, установленные по умолчанию. Элементы списка с навигационными ссылками делаем плавающими, формируя горизонтальное меню, но для элементов списка, содержащих подпункты задаем float: none;, чтобы они отображались друг под другом.

|  |
| --- |
| #navbar, #navbar ul {  margin: 0;  padding: 0;  list-style-type: none;  }  #navbar li { float: left; }  #navbar ul li { float: none; } |

Затем нам нужно сделать так, чтобы наше выпадающее подменю не смещало контент, расположенный под панелью навигации, вниз. Для этого мы зададим пунктам списка position: relative;, а списку, содержащему подпункты position: absolute; и добавим свойство top со значением 100%, чтобы абсолютно позиционированное подменю отображалось точно под ссылкой.

|  |
| --- |
| #navbar ul {  display: none;  position: absolute;  top: 100%;  }  #navbar li {  float: left;  position: relative;  }  #navbar { height: 30px; } |

Высота для родительского списка была добавлена специально, так как браузеры не учитывают в качестве содержимого элемента плавающий контент, то без добавления высоты наш список будет проигнорирован браузером и контент, следующий за списком, будет обтекать наше меню.

Теперь нам осталось стилизовать оба наших списка и выпадающее меню будет готово:

|  |
| --- |
| #navbar ul {  display: none;  background-color: #f90;  position: absolute;  top: 100%;  }  #navbar li:hover ul { display: block; }  #navbar, #navbar ul {  margin: 0;  padding: 0;  list-style-type: none;  }  #navbar {  height: 30px;  background-color: #666;  padding-left: 25px;  min-width: 470px;  }  #navbar li {  float: left;  position: relative;  height: 100%;  }  #navbar li a {  display: block;  padding: 6px;  width: 100px;  color: #fff;  text-decoration: none;  text-align: center;  }  #navbar ul li { float: none; }  #navbar li:hover { background-color: #f90; }  #navbar ul li:hover { background-color: #666; } |