* [Главная](https://htmlacademy.ru/)
* [Моё обучение](https://htmlacademy.ru/study)
* [**Создание семантической разметки по макету**](https://htmlacademy.ru/skills/semantic-markup)

**Методика семантической разметки**

**Для чего подходит это руководство и как им пользоваться**

HTML-разметка страницы пишется по определённым правилам, которые описываются в специальных документах — спецификациях. Спецификации разрабатываются сообществом разработчиков браузеров и профессионалов отрасли.

Это руководство поможет понять, как правильно написать семантическую HTML-разметку типовой веб-страницы.

Что же такое семантика? Это, собственно, и есть следование правилам, которые описываются в спецификациях. Семантическая разметка — разметка, в которой все элементы используются согласно их назначению.

Изначально веб задумывался как способ обмениваться документами, и большинство существующих типов сайтов (информационные сайты, блоги, новостные сайты, магазины и так далее) можно представить в виде документов — страниц, содержащих информационный контент.

А что, если у меня не текстовый документ, а интерфейс веб-приложения? Ответ [тут](https://htmlacademy.ru/skills/semantic-markup/accessory-01).

Подробно о спецификации языка HTML и работе с ней рассказывается в [отдельной статье](https://htmlacademy.ru/skills/semantic-markup/accessory-02).

В процессе разметки страницы мы будем размечать все содержательные сущности документа подходящими по смыслу тегами. Декоративные (не содержательные) сущности в разметку входить не должны.

Как происходит типичный процесс написания разметки:

1. Получаем графический макет.
2. Пишем по нему разметку в текстовом редакторе.
3. Периодически открываем файл с написанной разметкой в браузере и видим результат.

Далее в этом руководстве мы рассмотрим алгоритм, которому можно следовать при разметке большинства типовых веб-страниц. Для удобства все шаги будем разбирать на практических примерах из реальных макетов.

**Алгоритм написания разметки**

Подготовительный этап

**Шаг 1. Выделяем крупные смысловые блоки на каждой странице сайта.**

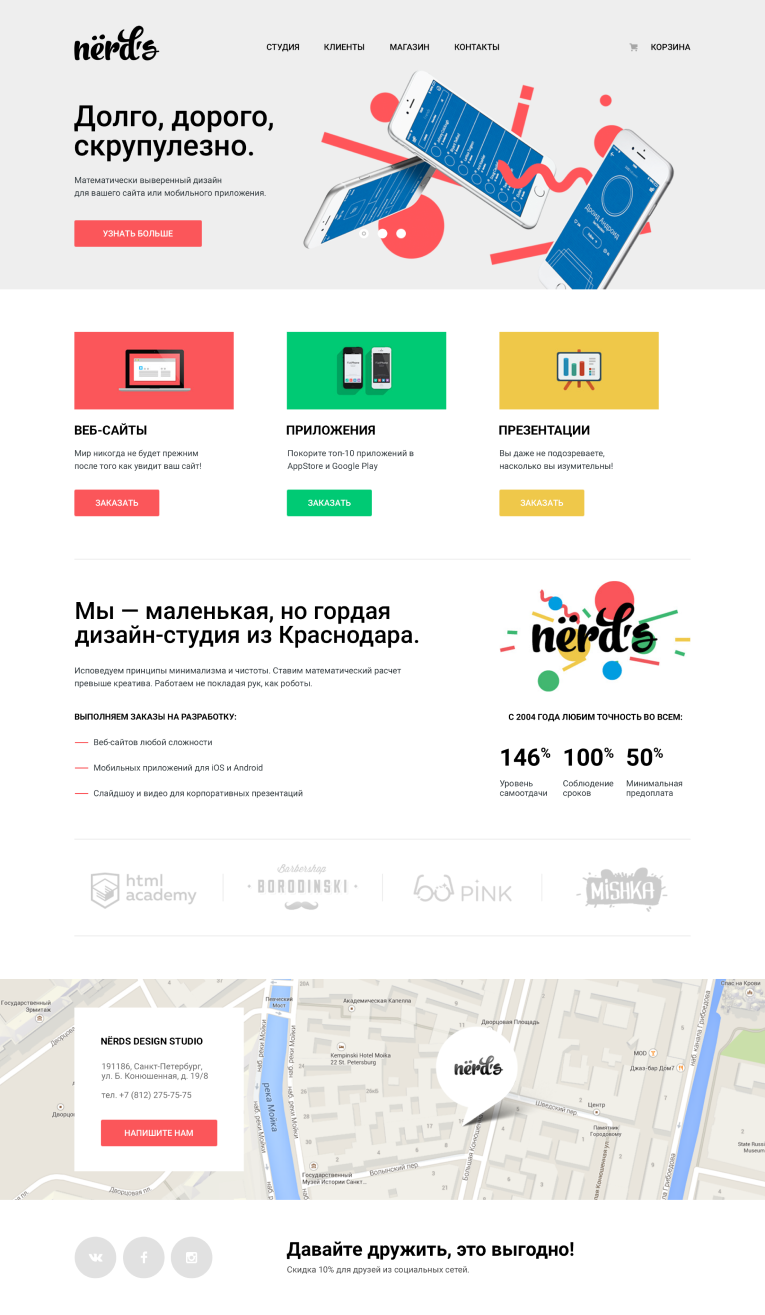
Типичный список блоков: *Шапка*, *Подвал* и *Основное содержимое* или, если перевести это на язык тегов, <header>, <footer> и <main>.

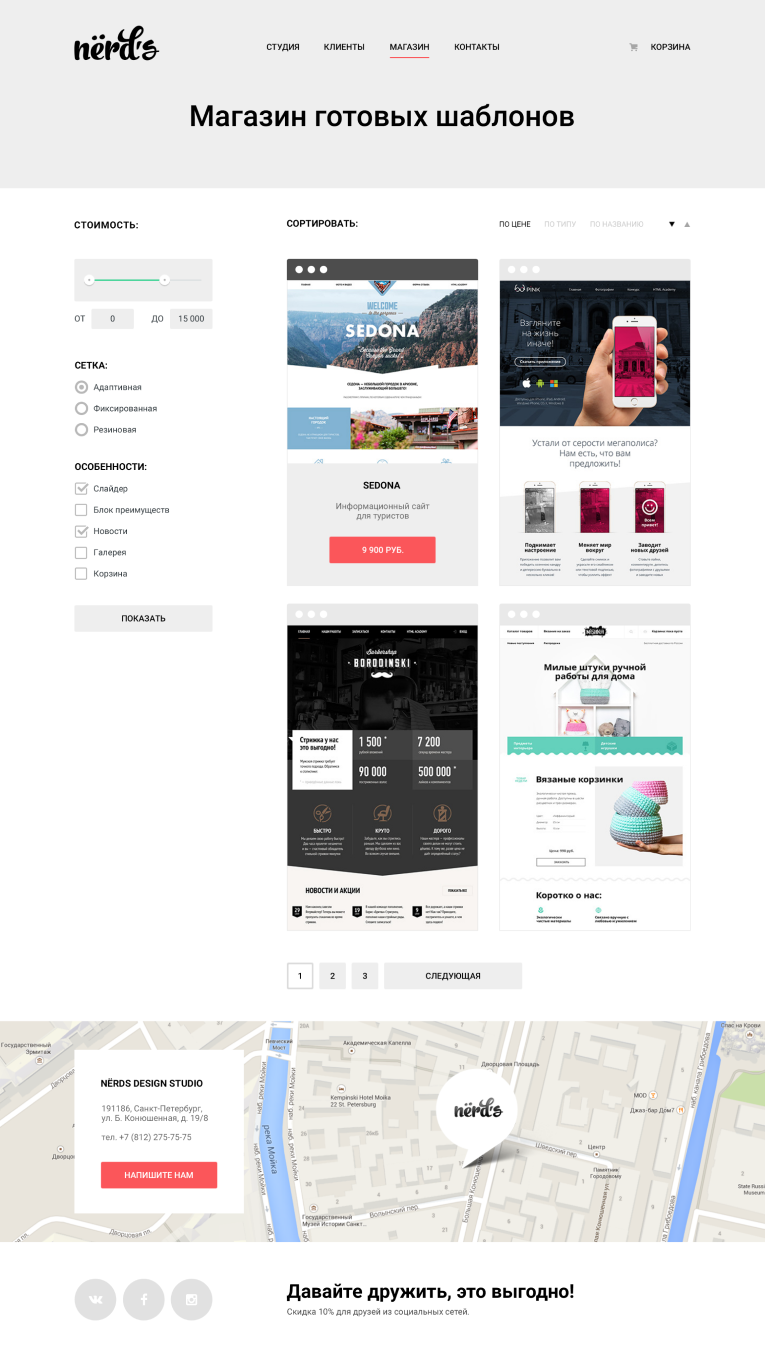
Вспомним назначение каждого из указанных семантических тегов:

* <header> — это вводная часть всего документа или одного из его разделов ([подробнее рассказывается в интерактивных курсах](https://htmlacademy.ru/courses/299/run/6));
* <main> — основное содержимое страницы — то, что не повторяется на других страницах, уникальное содержимое каждой страницы;
* <footer> — это заключительная часть всего сайта или смыслового раздела со справочной информацией.

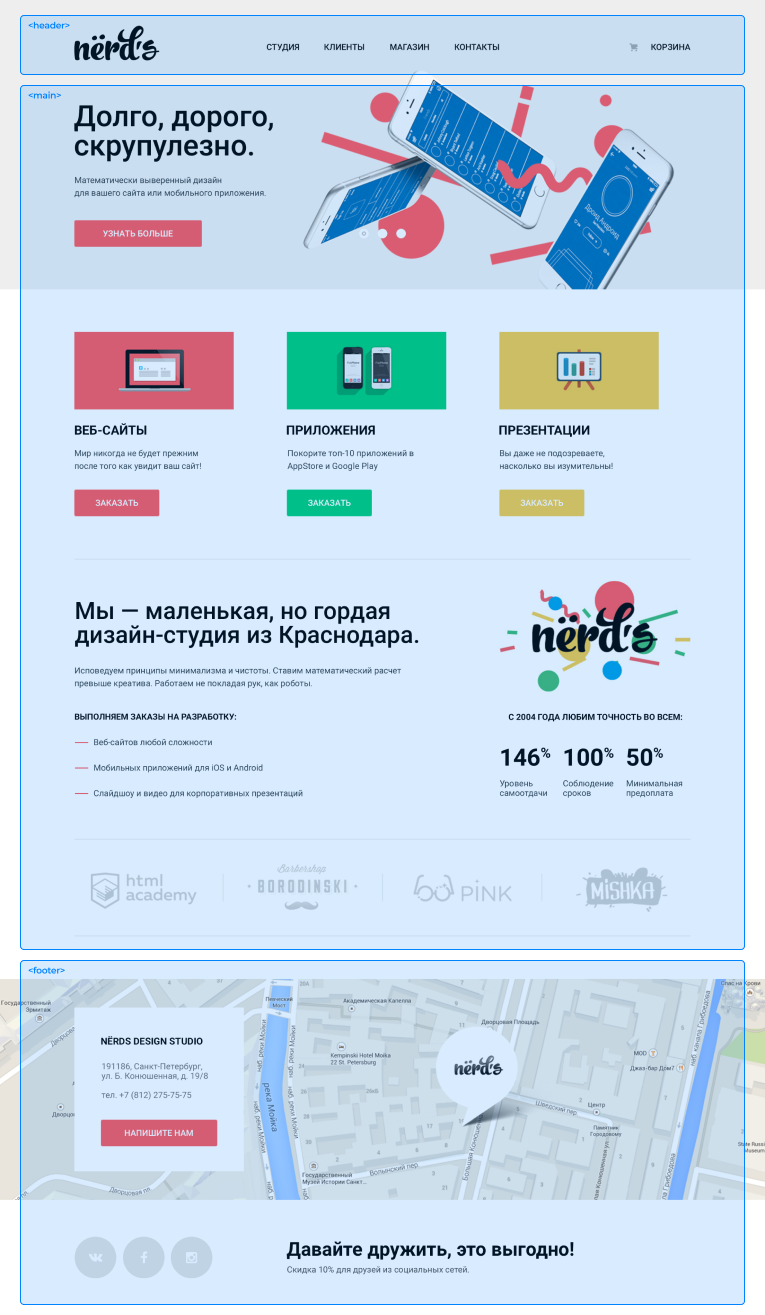
Мы наглядно продемонстрируем процесс определения этих блоков, для этого понадобится, как минимум, две страницы сайта.

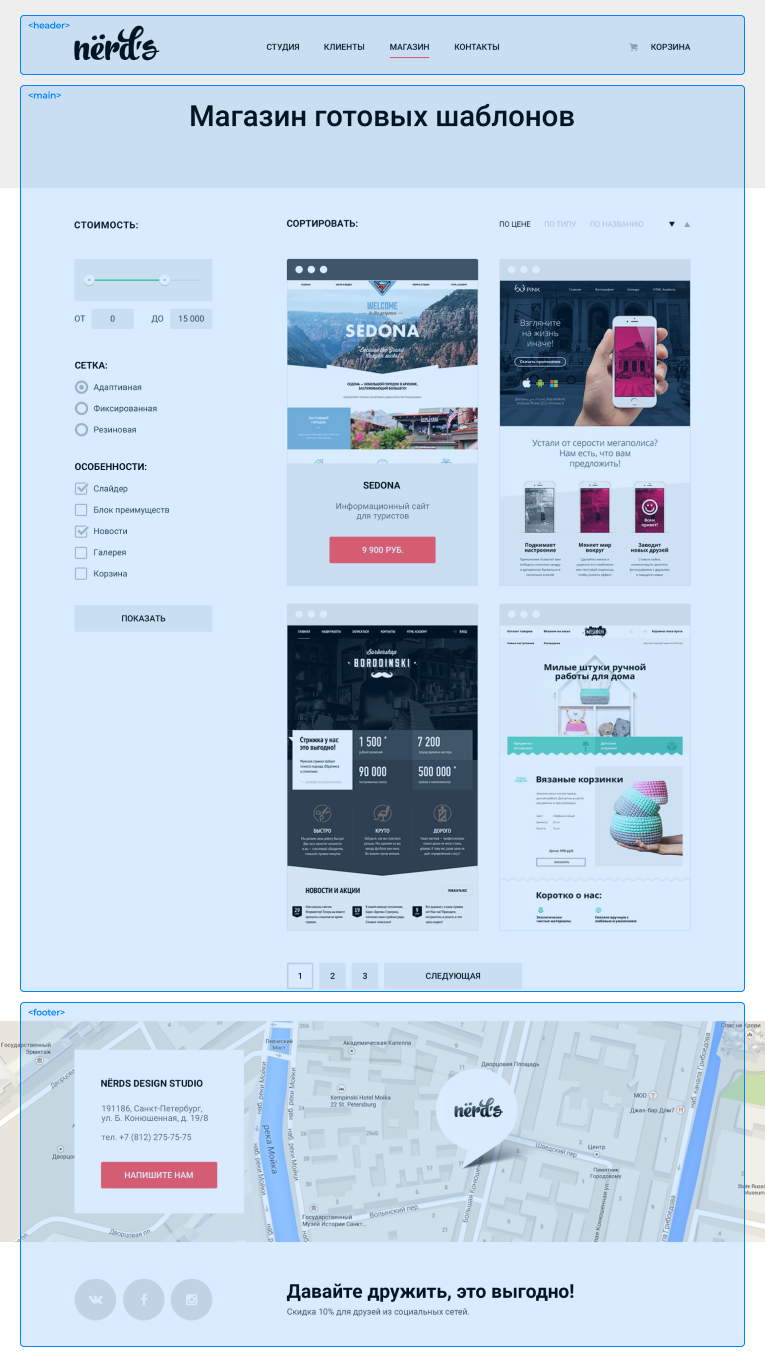
Рассмотрим макет главной страницы и страницы каталога сайта интернет-магазина Nerds:





После анализа макетов легко определить, что станет шапкой сайта, основным содержимым, уникальным для каждой страницы, и подвалом сайта. В итоге получим следующий результат:





Прямоугольниками синего цвета мы выделили крупные смысловые блоки.

Структура страницы в разметке на этом этапе такая:

<!DOCTYPE html>

<html lang="ru">

<head>

<meta charset="utf-8">

<title>Заголовок страницы</title>

</head>

<body>

<header class="main-header">

<!-- Шапка сайта -->

</header>

<main>

<!-- Основное содержимое страницы -->

</main>

<footer class="main-footer">

<!-- Подвал сайта -->

</footer>

</body>

</html>

Замечание

**Шаг 2. Размечаем в блоках крупные смысловые разделы.**

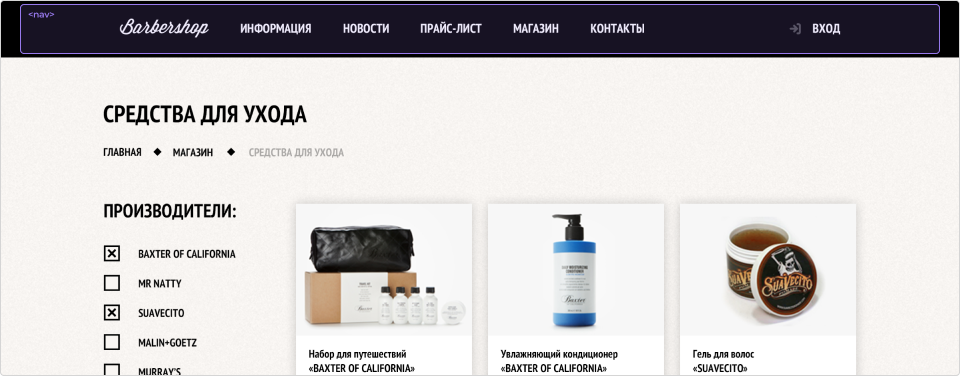
На этом этапе выделяются такие элементы, как:

* <nav> — главная навигация;
* <section> — смысловой раздел, который нельзя отделить от конкретной страницы или документа;
* <article> — отделяемый от сайта смысловой раздел;
* <aside> — дополнительное содержимое.

Начнём с примера тега <nav> — главной навигации сайта. К такому элементу, как правило, относится навигация по сайту, а также логотип — изображение и ссылка на главную страницу сайта.

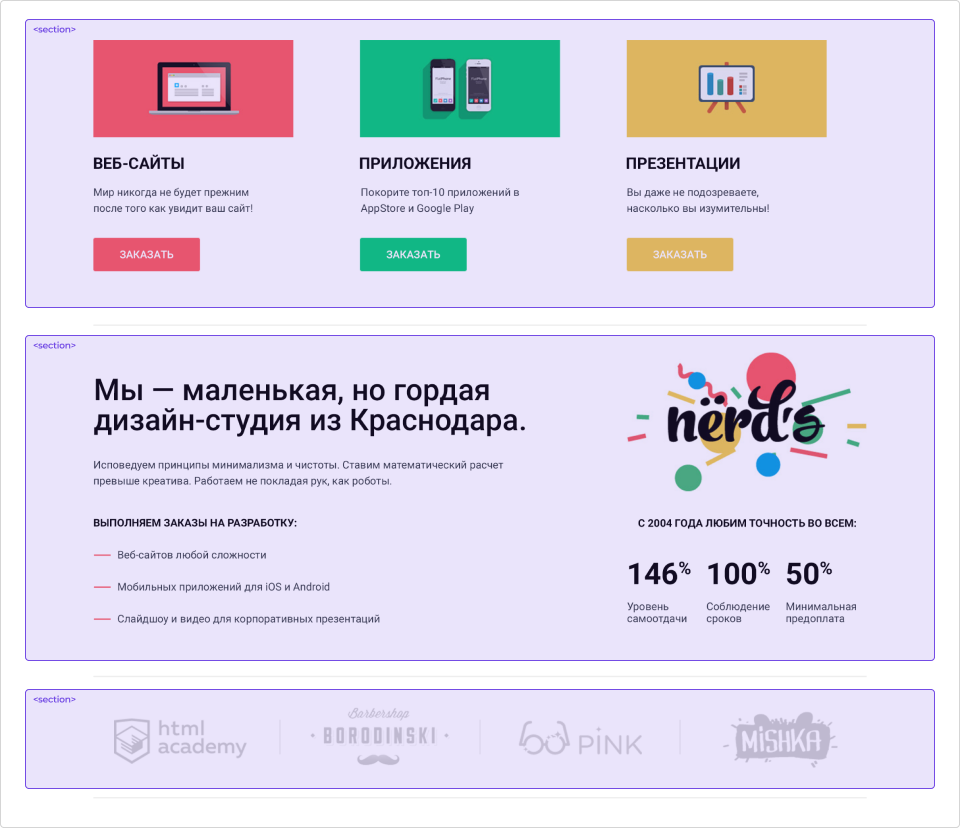
Пример тега <nav> на макете

Помимо навигации по сайту и логотипа, в тег <nav> может включаться и пользовательское меню или ссылка для входа на страницу авторизации пользователя.

Пример тега <nav> на макете

Замечание

Разберём ещё примеры.

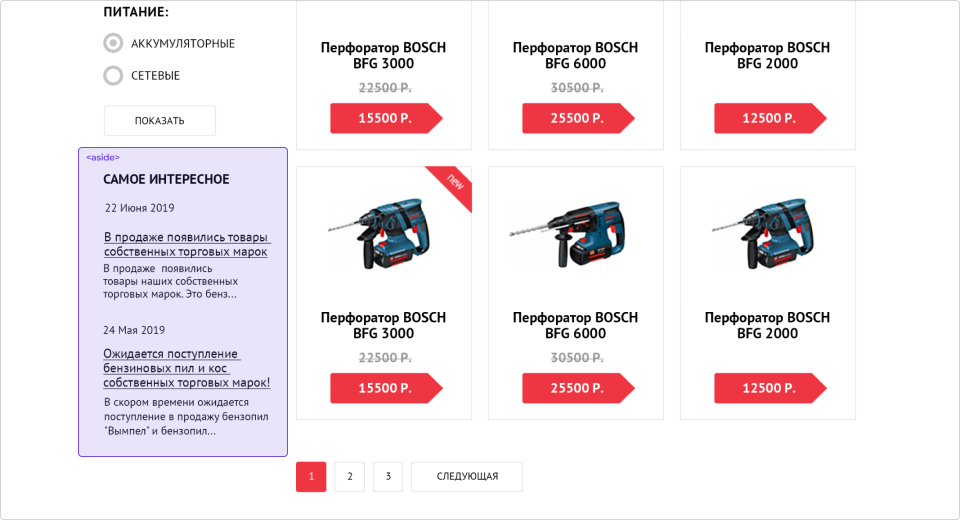
Структура страницы сайта

В макете главной страницы Nerds можно выделить смысловой блок с перечнем услуг, раздел «О нас» и список логотипов компаний. Все эти разделы связаны с тематикой сайта и вне этого сайта не имеют смысла, поэтому их следует разметить тегом <section>.

На втором примере выделен блок, который вполне может существовать и отдельно от сайта. Поэтому его можно разметить тегом <article>.

Пример тега <article> на макете

Для определения на макете элемента <aside> ищем блок, который напрямую не связан с главным содержимым сайта. Это, как правило, анонсы, реклама и подобное. Вот пример блока <aside>:

Пример тега <aside> на макете

**Шаг 3. Выделяем заголовок всего документа и заголовки смысловых разделов.**

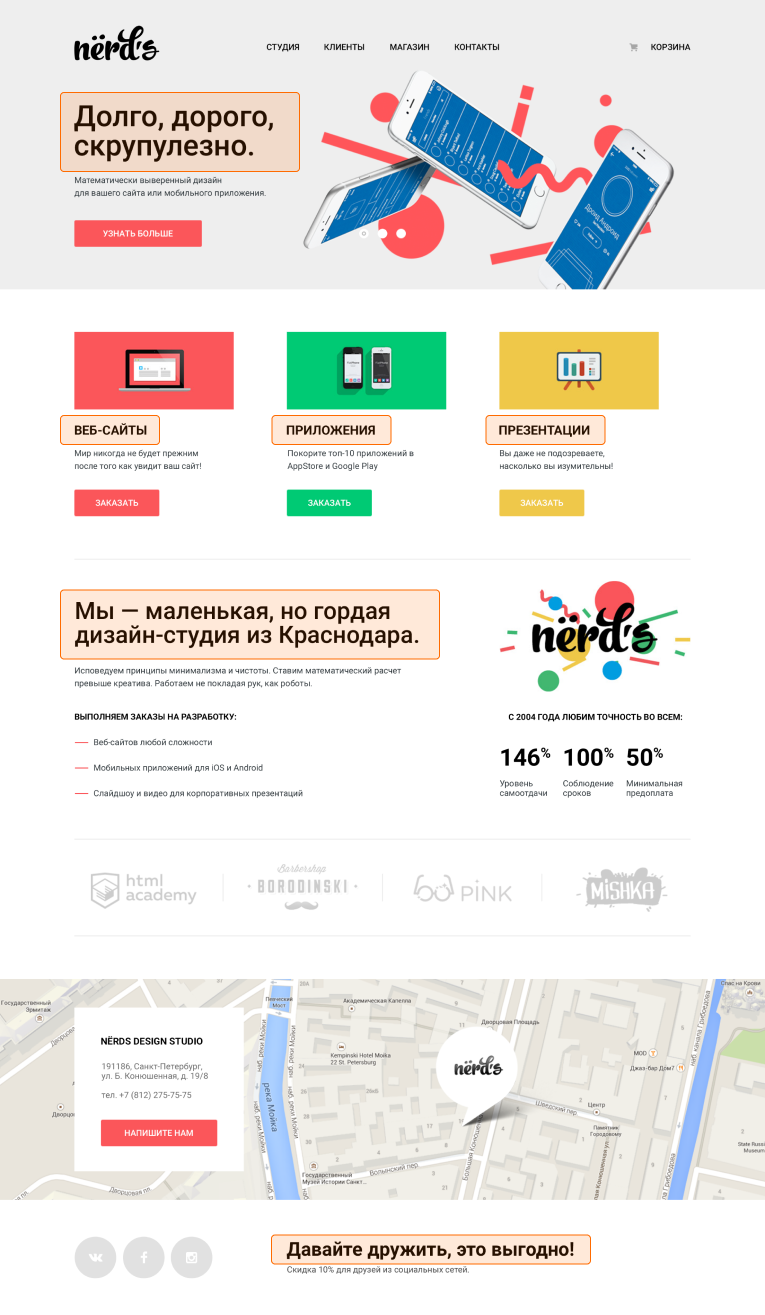
Следующий этап после выделения крупных смысловых блоков — разметка заголовков.

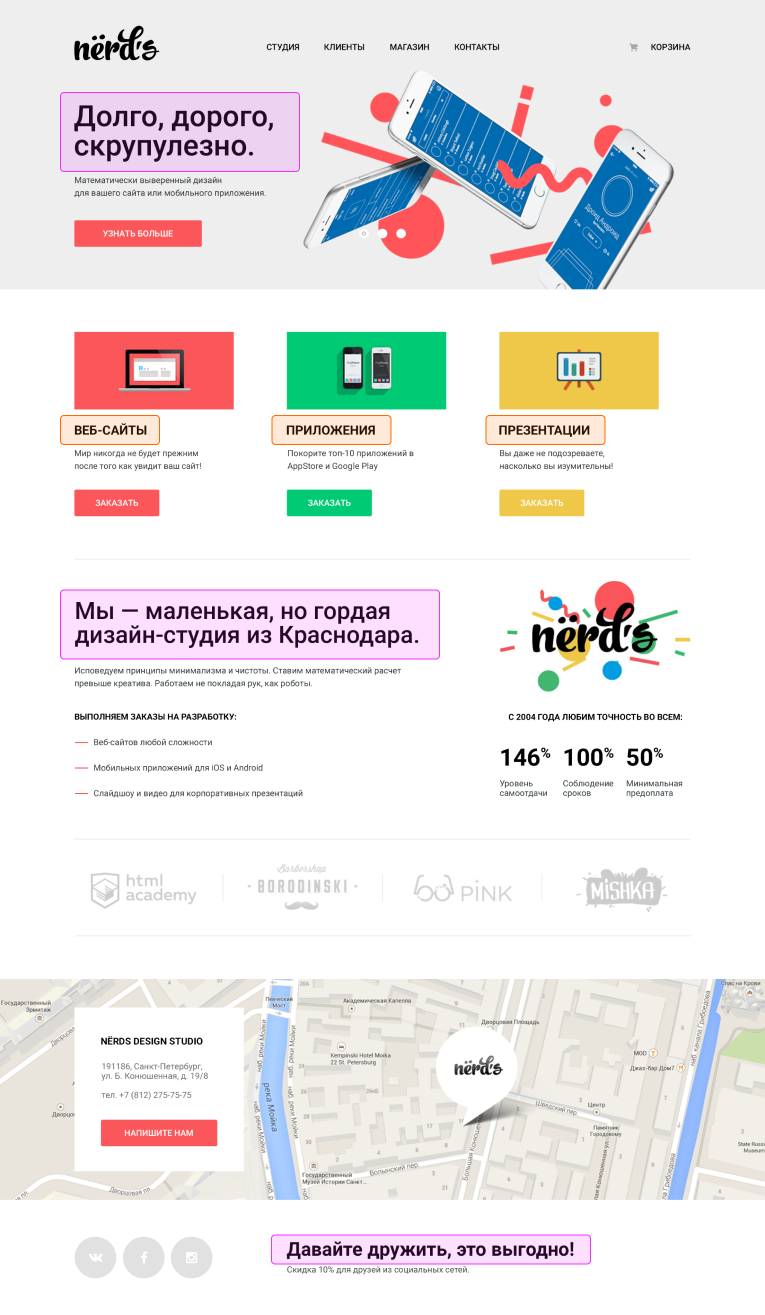
Подходящие по смыслу заголовки лучше добавлять в разметку страницы, даже если в макете заголовок документа или смыслового блока отсутствует. Так документ будет более структурированным. При этом, чтобы свёрстанный сайт визуально полностью соответствовал макету, такие «невидимые» заголовки скрываются с помощью специального класса в CSS. Таким образом «визуально скрытый» заголовок остаётся видимым для поисковиков и скринридеров. Подробно узнать о том, как правильно прятать, можно в HTML-шорте «[Как прятать](https://htmlacademy.ru/blog/video/shorts/short-12)».

Заголовок — это обманчиво простой тег. Главная проблема с заголовками такая: не всегда то, что кажется заголовком, им является. Зачастую вы видите очень крупный текст на макете и думаете, что это заголовок, но это не всегда так. Прежде всего важен не внешний вид заголовков, а их смысл.

Чтобы легко работать с заголовками, нужно понять их основную функцию в тексте. Основная роль заголовка — резюмирующая, он очень сжато передаёт содержание последующего текста. Прочитав заголовок, вы понимаете, что будет после него. Если резюмирующая роль выполняется, это точно заголовок.

Давайте отработаем навык определения на макете элементов, соответствующих заголовочным тегам (<h1>-<h6>). Рассмотрим макет и разберём часто встречающиеся ошибки при определении заголовков в документе.

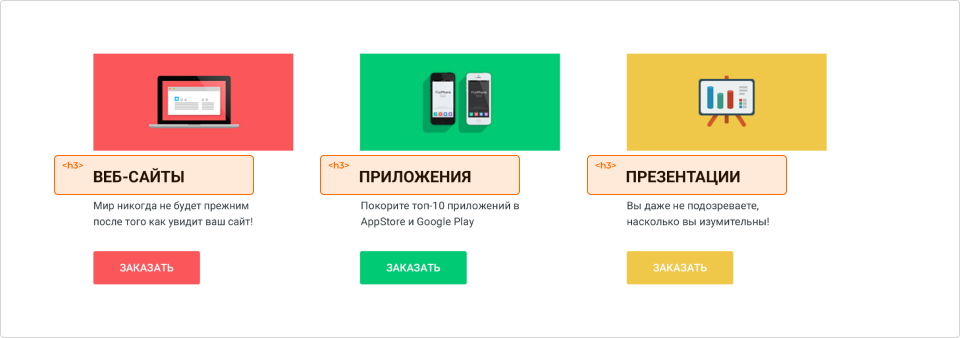




На первом макете выделены кандидаты на роль заголовков, но, как видно из второго макета, не все из них таковыми являются.

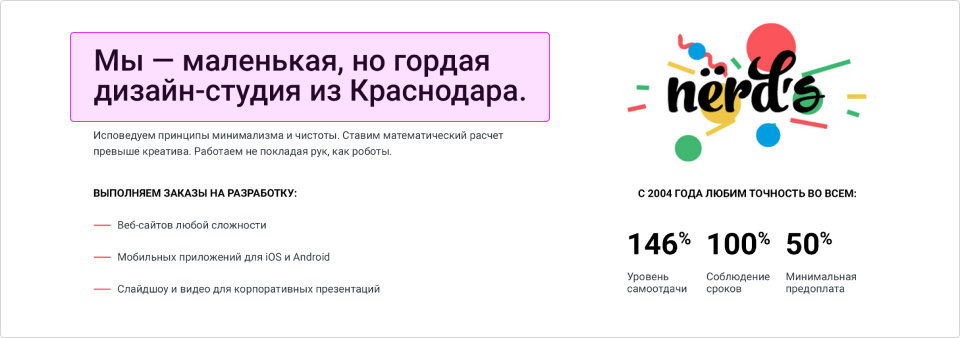
Начнём с заголовка первого уровня. Было бы большой ошибкой сделать заголовком первого уровня самый крупный текст на странице — текст «Долго, дорого, скрупулёзно». Во-первых, этот текст не содержательный. Во-вторых, это текст одного из нескольких слайдов, то есть существует как минимум несколько похожих блоков. То есть этот заголовок не может быть главным. Максимум — это заголовок второго уровня, но нужно ещё проверить его роль для последующего текста.

А на роль заголовка всего документа подойдёт текст: Интернет магазин «Nerds». Несмотря на то что этого текста нет в макете, в разметку его нужно добавить в «скрытом» виде.

Пример заголовков на макете

Дальше идёт смысловой блок «Услуги», у которого на макете также нет заголовка. Давайте добавим этому блоку «скрытый» содержательный заголовок второго уровня. Далее идут названия самих услуг. Являются ли эти элементы важными, смысловыми, содержательными? Да, являются, потому что это описание того, чем живёт студия, описание услуг. Значит, «веб-сайт», «приложения», «презентации» станут уже <h3>.

Есть также альтернативный вариант: пропустить «скрытый» заголовок «Услуги» и сделать названия услуг заголовками второго уровня.

Пример *не*заголовка на макете

Далее, текст *«Мы — маленькая, но гордая веб-студия из Краснодара»* — заголовок или нет? Не заголовок. Почему? Это само содержание, а не его резюме. Из этого текста непонятно, что будет дальше, что именно студия делает. Хорошим заголовком для этого блока был бы текст «О нас» или «О студии».

Текст «Давайте дружить, это выгодно!» также не является заголовком, это скорее призыв к действию, и его стоит разметить обычным тегом <p>.

В итоге получилась такая разметка:

<main>

<h1>Интернет магазин «Nerds»</h1>

<section class="services">

<h2>Услуги</h2>

…

</section>

<section class="about">

<h2>О нас</h2>

…

</section>

<section class="companies">

<h2>Производители</h2>

…

</section>

</main>

**Шаг 4. Размечаем мелкие элементы в смысловых разделах.**

Определить, какие теги использовать, можно методом исключения:

* Получилось найти самый подходящий смысловой тег — использовать его.
* Для потоковых контейнеров — <div>.
* Для мелких фразовых элементов (слово или фраза) — <span>.

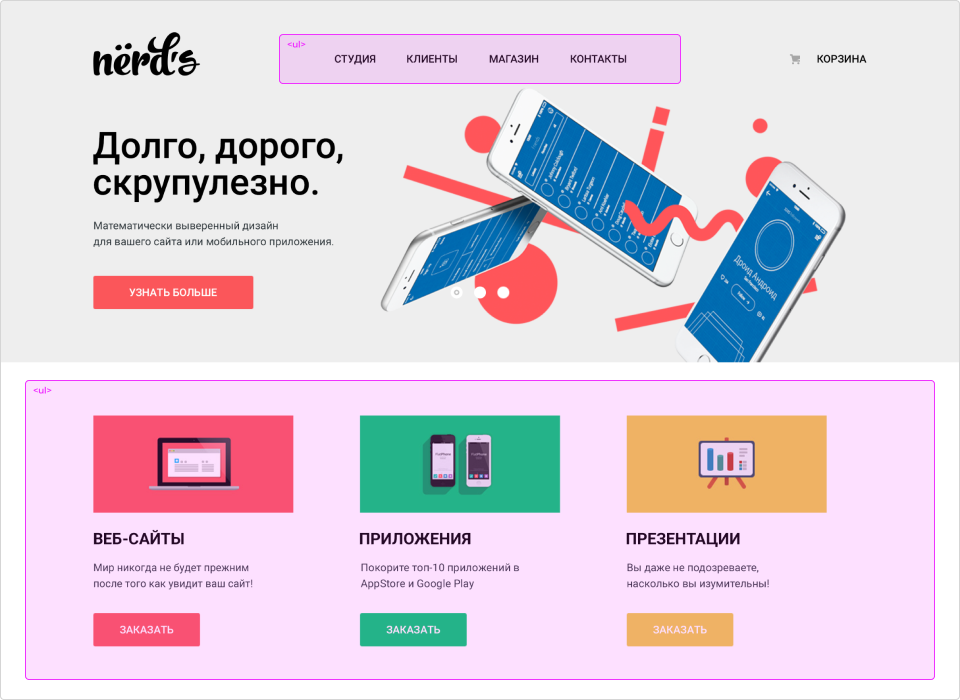
На этом этапе разберём, в каких случаях используются теги <ul>, <ol>, <dl>, <p> и <blockquote>.

Несколько элементов, выполняющих одну функцию и находящихся друг с другом в одном отношении, размечают списком. Списки объединяют однородные и равноправные элементы. Если порядок элементов в списке неважен, следует использовать тег <ul>, а для упорядоченных списков — <ol>.

Ссылки в навигации по сайту, перечень услуг, предоставляемых на сайте, — все эти элементы следует объединять в списки, поскольку они однотипные.

Корзину пользователя можно тоже считать списком, который включает всего один элемент.

Списки на макете:

Пример списков на макете

Главная ошибка при использовании списков — объединять в список то, что списком не является, то есть неоднородные и неравноправные элементы. К примеру, на макете ниже все выделенные элементы выполняют разные функции. Единственное, что их объединяет — визуальное расположение, а значит, эти элементы не могут быть объединены в список.

Пример несписков на макетеПример *не*списков на макете

Ещё один вид списков, которые могут встретиться на странице сайта — это <dl>. Элемент <dl> (от английского Description List) представляет собой список описаний и служит контейнером для списка пар терминов (определяемых элементом <dt>) и их описаний (определяемых элементами <dd>). Этот элемент обычно используют при создании глоссария, для отображения метаданных (списка пар ключ-значение), для описания характеристик товара и других.

Разберём пример использования тега <dl> в типичной карточке товара.

Пример списка определений на макете

Технические данные о модели перфоратора можно представить как набор пар: параметр и значение. И использовать в этом случае теги <dl> — <dt> — <dd>. Разметим технические данные как список определений. В итоге получим следующую разметку для этого блока:

<dl>

<dt>Питание</dt>

<dd>от сети</dd>

<dt>Тип патрона</dt>

<dd>SDS-Plus</dd>

<dt>Количество скоростей работы</dt>

<dd>1</dd>

<dt>Потребляемая мощность</dt>

<dd>880 Вт</dd>

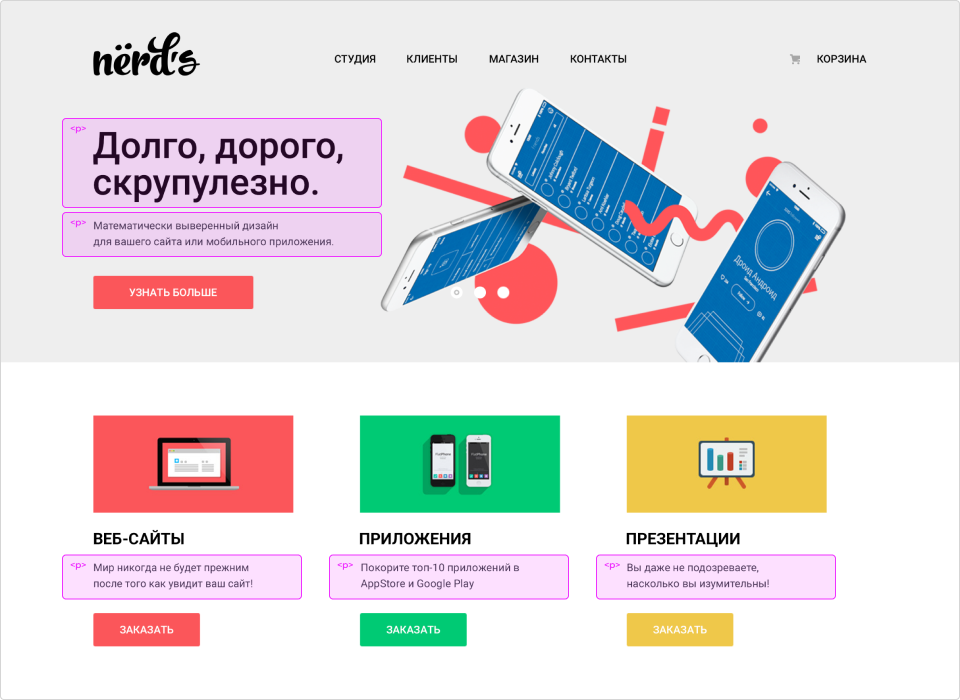
…

</dl>

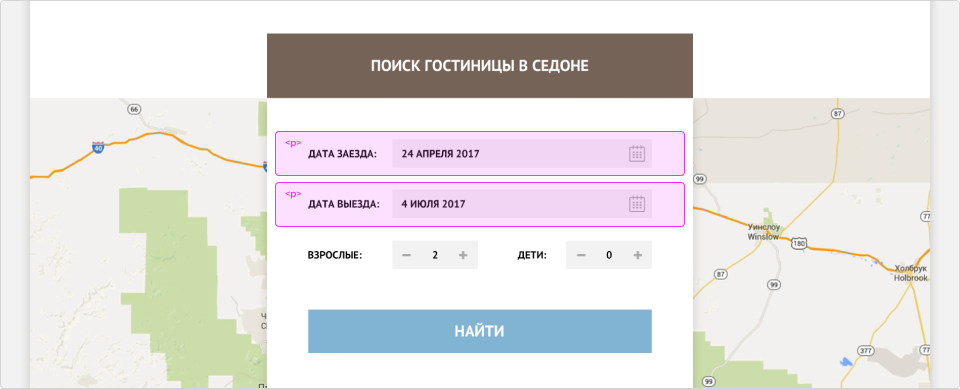
Замечание

Больше примеров использования тега <dl> можно найти в главе интерактивного курса «[Разметка текста](https://htmlacademy.ru/courses/301/run/6)».

Теперь давайте разберёмся, в каких случаях следует использовать тег <p> — параграф, или абзац. В HTML параграф — это неразрывная последовательность фразовых элементов. То есть абзац — это структурная сущность, форма. Найдём такие элементы на макете:

Пример параграфов на макете

Тег <p> удобно использовать как контейнер для мелких фразовых элементов. Простой пример — форма поиска отелей. Типовая задача разметки форм, в которой нужно сгруппировать подпись — <label> и само поле для ввода — <input>. Теги <label> и <input> — два фразовых элемента, которые формируют отдельную структурную сущность. Для группировки этих элементов можно использовать тег <p>.

Пример параграфов на макете

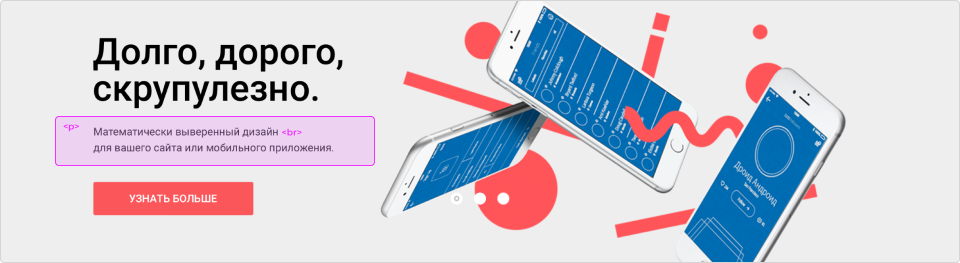
<p>

<label>Дата заезда:</label>

<input type="text" value="24 апреля 2017">

</p>

Обратите внимание на текст *«Математически выверенный дизайн для вашего сайта или мобильного приложения.»* на макете ниже. Для его разметки отлично подойдёт тег <p>, но будет ли этого достаточно, чтобы добиться полного соответствия макету? Даже если ограничить ширину абзаца с помощью стилей, получить перенос в требуемом месте не удастся. Поэтому для этого случая воспользуемся тегом <br>.

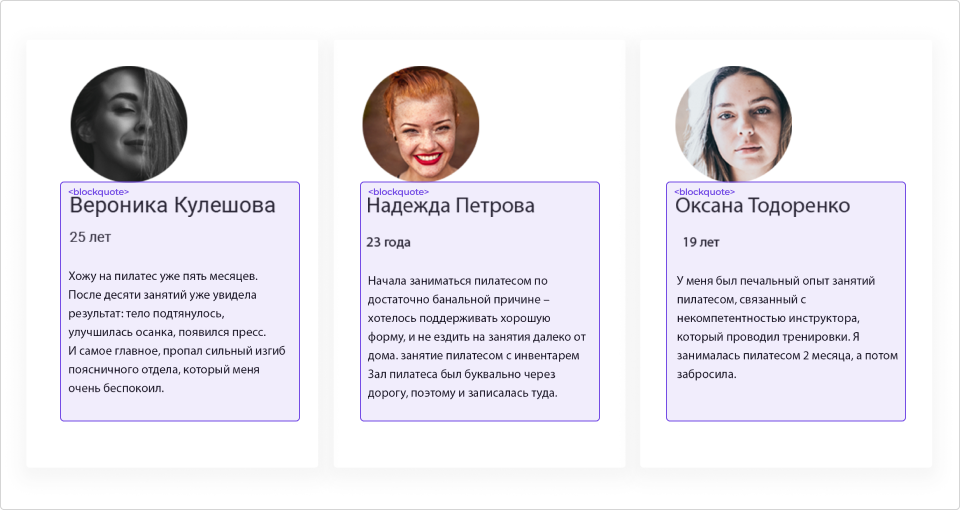
Пример переноса на макете

Стоит помнить, что использование <br> допустимо точечно в небольших информационных и промо-текстах, если так требуется в макете.

Но в то же время недопустимо использовать тег <br> для разделения абзацев вместо тега <p>. Также его нельзя использовать в крупных текстовых массивах для подгонки переносов текста «как в макете». Ошибкой будет проставить теги <br> в конце каждой строки, чтобы текст переносился один в один как на макете.

Во всех этих случаях вёрстка становится «негибкой». Перестройкой текста с добавленными принудительными переносами становится неудобно управлять.

Один из типовых разделов сайта — блок с отзывами посетителей сайта, покупателями товаров/услуг. Для разметки отдельного отзыва в таком блоке может пригодиться тег <blockquote>. Следует помнить, что в тег оборачивается не только цитата, но и элемент с источником цитаты, а также дополнительная информация об источнике.

Пример использования тега <blockquote>

<blockquote>

<cite>Оксана Тодоренко</cite>

<p>19 лет</p>

<p>У меня был печальный опыт занятий пилатесом, связанный с некомпетентностью инструктора, который проводил тренировки. Я занималась пилатесом 2 месяца, а потом забросила.</p>

</blockquote>

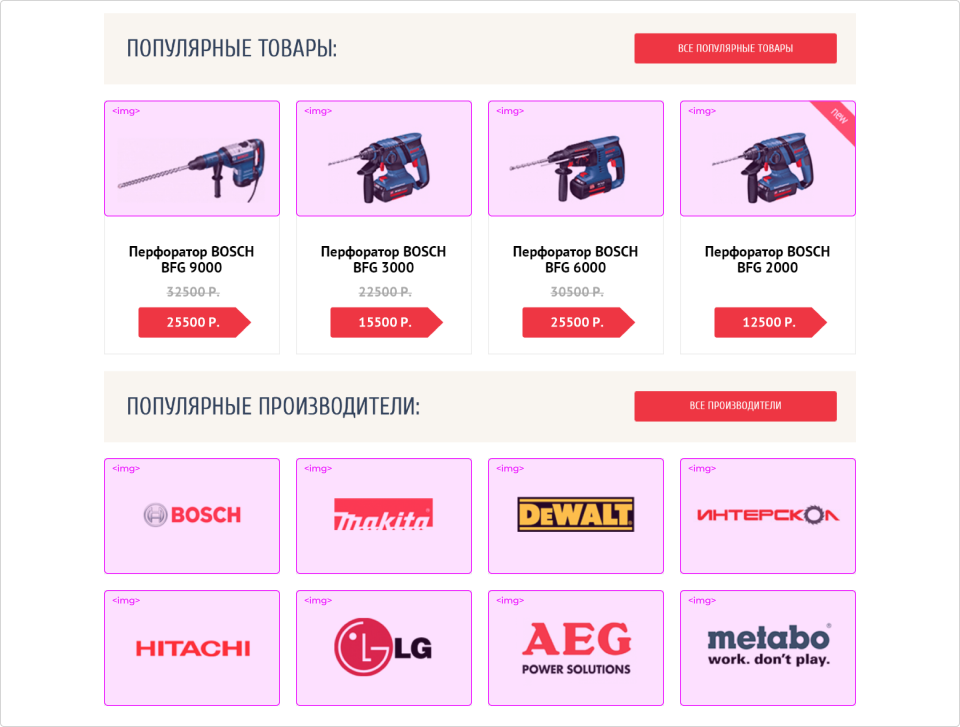
Подробнее с тегом <blockquote> можно ознакомиться в интерактивном курсе [Разметка текста](https://htmlacademy.ru/courses/301/run/12).

Попробуем найти на макетах фразовые элементы:

* <img>
* <a>
* <video>
* <b>, <i> и их семантических «родственников» <strong> и <em>
* <span>

Тегом <img> размечают содержательные изображения. Такие изображения имеют информационное значение и несут смысл для пользователя, так как влияют на его выбор, на принимаемое пользователем решение. К примеру, это фотографии товара в интернет-магазине.

К содержательным изображениям также относится логотип сайта и логотипы брендов.

Пример содержательных изображений на макете

При использовании тега <img> хорошим тоном считается заполнение атрибута alt, в значении которого кратко описывается изображённое на картинке. Также желательно указывать размеры изображения, чтобы браузер лишний раз не перестраивал страницу при загрузке.

<img src="img/photo-BFG9000.jpg" width="220" height="165" alt="Перфоратор BOSCH">

В аббревиатуре HTML «HT» значит hypertext, а «ML» — markup language, то есть язык разметки гипертекста. Гипертекст — это документы, связанные между собой ссылками. В разметке ссылки размечаются тегом <a>. В разметке наших макетов мы используем тег <a> для разметки навигации по сайту.

Также часто тегом <a> оборачивают логотип сайта, чтобы дать пользователю возможность из любого места сайта вернуться на главную страницу.

Рассмотрим примеры:

Пример ссылок на макете

<ul class="site-navigation">

<li><a href="info.html">Информация</a></li>

<li><a href="photos.html">Фото и видео</a></li>

<li><a href="map.html">Карта штата</a></li>

<li><a href="hotels.html">Гостиницы</a></li>

</ul>

<a class="main-header-logo">

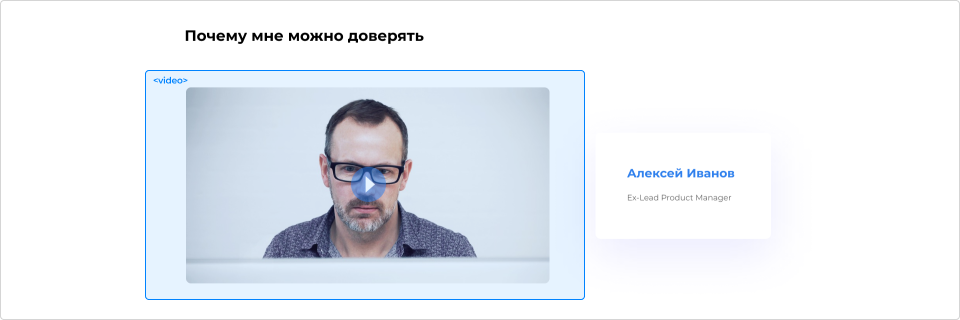
<img src="img/logo.svg" width="170" height="70" alt="Логотип сайта городка Седона">

</a>

Замечание

Иногда ссылки могут выглядеть как кнопки, а кнопки — как ссылки. Подробно об этом вы можете прочитать в [статье](https://htmlacademy.ru/skills/semantic-markup/accessory-05).

На страницах типичных лендингов с большим количеством визуального контента часто используется встроенный видео-контент. До появления HTML5, если вы хотели добавить видео на веб-страницу, приходилось использовать элемент <object>, представляющий собой универсальный контейнер для внешних объектов, либо подключать внешний модуль браузера. Сейчас же для добавления видео на страницу достаточно тега <video>, который поддерживает большое количество видео форматов. Пример ниже содержит видео-контент.

Пример использования тега <video>

Разметка для такого контента может выглядеть так:

…

<video poster="img/preview.jpg" width="580" height="290">

<source src="video.mp4" type="video/mp4">

</video>

<button type="button">Смотреть видео</button>

…

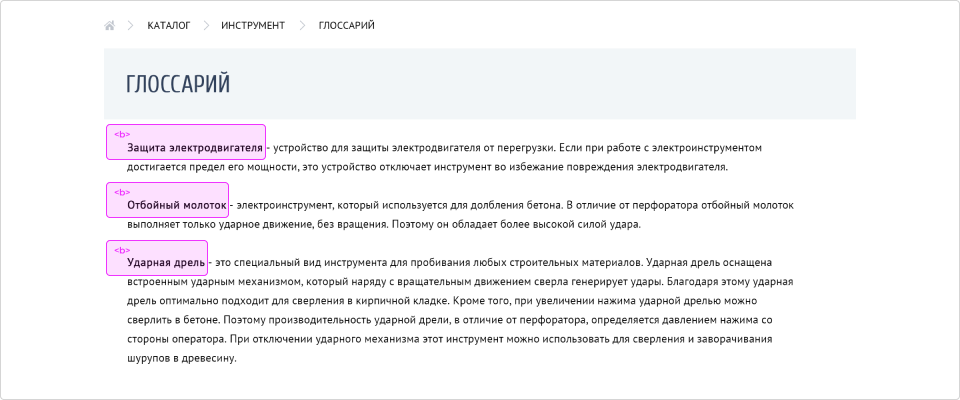
В этом примере показано, как с помощью атрибута poster можно добавить картинку-обложку для видео. Часто встречается ситуация, когда стандартные элементы управления видео, которые подключаются с помощью атрибута controls, не подходят под макет сайта. Есть возможность их отключать, а взамен использовать кнопки, внешний вид которых задаётся с помощью стилей. В нашем случае — это кнопка «Смотреть видео».

Подробнее об использовании тега <video> можно узнать из [интерактивных курсов](https://htmlacademy.ru/courses/73/run/15).

Дальше давайте разберём примеры разметки мелких текстовых элементов.

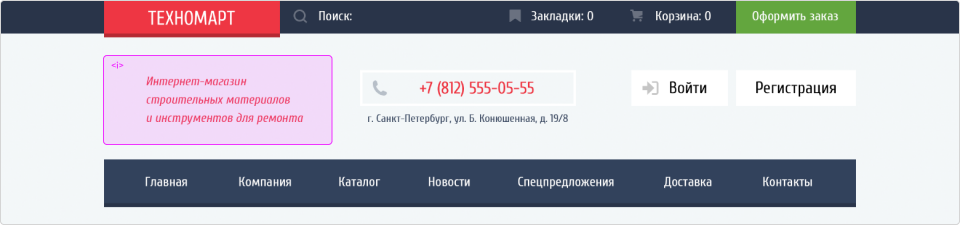
Авторы HTML5 предлагают использовать <b> для выделения отрывков текста с целью обратить внимание читателя, однако не подразумевая усиления значимости текста или интонации. Спецификация приводит примеры использования тега <b> для разметки ключевых слов в документе и лидов (первый абзац статьи в журналистике).

На макете страницы «Глоссарий» интернет-магазина электроинструмента даётся определение узкоспециализированных терминов (или ключевых слов). Предложения состоят из термина и текста, который поясняет этот термин. Название терминов при разметке можно обернуть в тег <b>.

Пример тега <b> на макете

<p><b>Защита электродвигателя</b> — устройство для защиты электродвигателя от перегрузки. Если при работе с электроинструментом достигается предел его мощности, это устройство отключает инструмент во избежание повреждения электродвигателя.</p>

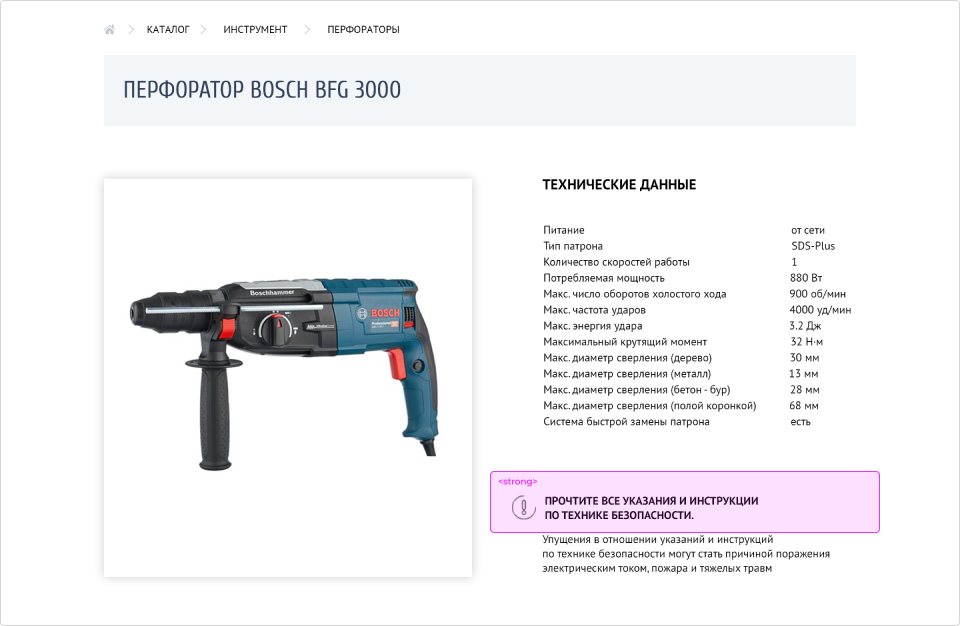
Разметим текст *«Интернет-магазин строительных материалов и инструментов для ремонта»*.

Пример тега <i> на макете

По задумке дизайнера внешний вид этого текста должен иметь наклонное начертание. При этом не должно быть акцентирования на тексте. В этом случае нам подойдёт тег <i> и разметка будет такой:

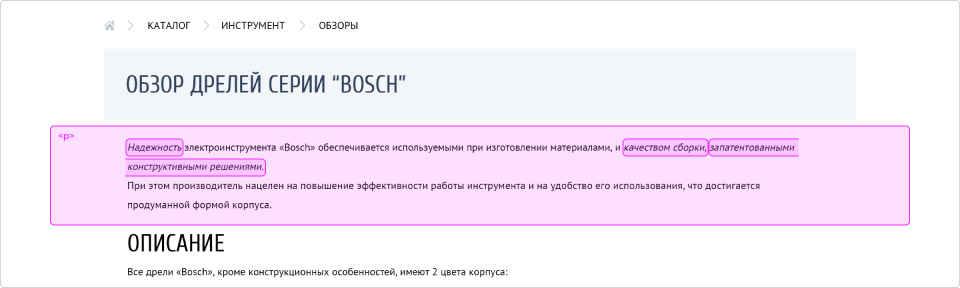
<i>Интернет-магазин строительных материалов и инструментов для ремонта</i>

При описании технических данных товара необходимо чтобы пользователь обратил внимание на технику безопасности при использовании. Возможно, вы встречали подобный текст в инструкциях по применению. В макете этот текст выделен жирным начертанием. Поскольку на этом тексте есть и смысловое акцентирование, в разметке используем подходящий для этого случая тег <strong>.

Пример тега <strong> на макете

<strong>Прочтите все указания и инструкции по технике безопасности.</strong>

В статье с обзором дрелей серии «Bosch» хочется сделать акцент на таких важных для пользователя понятиях, как «Надёжность», «Качество сборки», «Запатентованные конструктивные решения». В разметке в этом случае поможет тег <em>:

Пример тега <em> на макете

<p><em>Надёжность</em> электроинструмента «Bosch» обеспечивается используемыми при изготовлении материалами и <em>качеством сборки</em>, <em>запатентованными конструктивными решениями</em>. При этом производитель нацелен на повышение эффективности работы инструмента и на удобство его использования, что достигается продуманной формой корпуса.</p>

Замечание

Типичным примером использования фразового тега <span> может стать фрагмент текста, который выглядит немного иначе, чем окружающий его текст. К примеру, на макете ниже размер единицы измерения товара **/кг** явно меньше числа и символа рубля. Чтобы задать тексту меньший размер шрифта, можно обернуть его в тег <span> для удобства дальнейшей стилизации.

Пример тега <span> на макете

Разметка элемента будет такой:

<p>310 ₽<span>/кг</span></p>

Примеры использования на страницах сайтов форм и таблиц рассмотрены в отдельных разделах методики: [Таблицы на странице](https://htmlacademy.ru/skills/semantic-markup/accessory-06) и [Формы на странице](https://htmlacademy.ru/skills/semantic-markup/accessory-07).

Для дополнительного изучения рекомендуем [статью](https://htmlacademy.ru/skills/semantic-markup/accessory-04), где собраны часто встречающиеся теги и описаны типовые ошибки их использования.

**Итог**

Следуя алгоритму разметки, мы идём от общего к частному: сначала размечаем крупные блоки, а затем детализируем каждый блок. Такой подход позволяет чётко представлять структуру размечаемого документа и подробно реализовывать её. По сути, мы сначала создаём оглавление нашего документа, а затем пишем каждую его главу.