

B Academy
RO



Методичка

Lesson 2

HTML5 HyperText Markup Language

– is the standard markup language for describing the structure of documents displayed on the web



Виды элементов **по тегу**

● **Одиночные** (void elements)

- не содержат текст или вложенные элементы

```
<br>, <col>, <embed>, <hr>,  
<img>, <input>, <link>,  
<meta>, <source>, <track>,  
<wbr>
```

● **replaced elements**

- информация для отображения берется из их атрибутов

```
<iframe>, <video>, <embed>  
  
<input type="range"/>
```

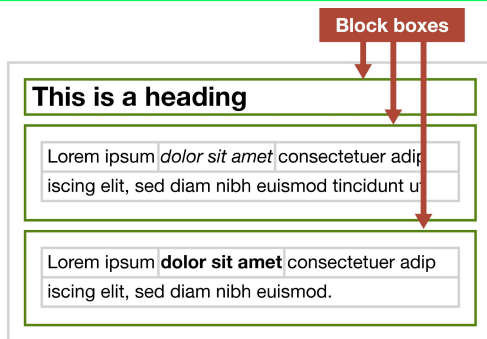
● **Закрывающиеся** (Non-replaced elements)

- имеют открывающий и закрывающий тег
- внутри может быть контент и другие теги

```
<p>  
  This paragraph has some  
  <strong> strong </strong>  
  content  
</p>
```

Виды элементов **по типу**

Блочные

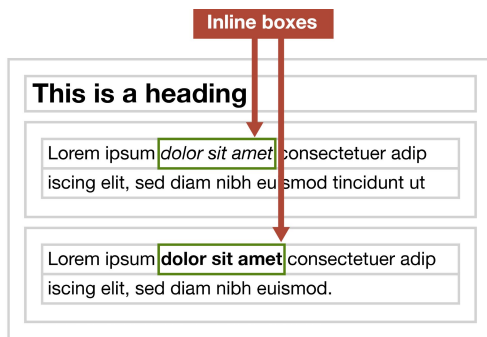


занимают всю ширину страницы (или все доступное пространство родителя)

не вкладываются в строчные элементы, но вкладываются в блочные. `<p>` и `<h1>-<h6>` не могут содержать блочные элементы, только строчные

большие структурные блоки

Строчные



занимают ровно столько, сколько контент

могут содержать только данные(текст) и другие строчные элементы, но `<a>` может содержать внутри блочные элементы

для выделения кусочка текста в блочном элементе

Анатомия HTML документа

1. доктайп (исторический артефакт)
2. корневой элемент (root)
3. элемент для подключения того, что на странице не будет видно
4. элемент устанавливает UTF-8 кодировку документу
5. элемент для названия страницы (отображается во вкладке браузера или закладках)
6. элемент для контента всей страницы



Let us see how we can break the code up in different components:

<html> ... </html>

This tag specifies that the webpage is written in HTML. It appears at the very first and last line of the webpage. It is mainly used to show that the page uses HTML5 – the latest version of the language. Also known as the root element, this tag can be thought of as a parent tag for every other tag used in the page.

<head> ... </head>

This tag is used to specify meta data about the webpage. It includes the webpage's name, its dependencies (JS and CSS scripts), font usage etc.

<title> ... </title>

As the name suggests, this tag contains the title/name of the webpage. You can see this in your browser's title bar for every webpage open in the browser. Search engines use this tag to extract the topic of the webpage, which is quite convenient when ranking relevant search results.

<body> ... </body>

Everything the user sees on a webpage is written inside this tag. It is a container for all the contents of the webpage.

Document summary

Example

```
<html>
  <head>
    <title>My First Website</title>
  </head>
  <body>

  </body>
</html>
```

Document Information

<base/>

Used to specify the base URL of your site, this tag makes linking to internal links on your site cleaner.

<meta/>

This is the meta data tag for the webpage. Can be useful for mentioning the page's author, keywords, original published date etc.

<link/>

This is used to link to scripts external to the webpage. Typically utilized for including stylesheets.

<style> ... </style>

The style tag can be used as an alternative to an external style sheet, or complement it. Includes the webpage's appearance information.

<script> ... </script>

Used to add code snippets, typically in JavaScript, to make webpage dynamic. It can also be used to just link to an external script.

Example

```
<html>
  <head>
    <meta charset="utf-8">

    <base href="http://myfirstwebsite.com"
          target="_blank" />

    <title>My Beautiful Website</title>

    <link rel="stylesheet"
          href="/css/master.css">

    <script type="text/javascript">
      var dummy = 0;
    </script>
  </head>
  <body>

  </body>
</html>
```

1 Семантика

— это использование тегов по их предназначению.

1. Нельзя проверить автоматически

2. Зачем использовать:

- a. Поисковые роботы и SEO
- b. Доступность через screen reader и reading mode
- c. Понятность кода
- d. Статус разработчика

Базовые



теги

“рабочие
лошадки”

<code><h1>...<h6></code>	Заголовки на странице для оформления структуры.
<code><p></code>	Размечает абзац текста.
<code><div></code>	Универсальный строительный блок без семантического значения, обертка для группировки элементов. Некоторые сайты на 99% состоят из этого тега, не нужно так.
<code></code>	Помогает выделить стилем часть текста в блоке. Не имеет семантики.
<code><header></code>	Шапка страницы или блока.
<code><footer></code>	Подвал страницы или блока
<code><main></code>	Блок с главным контентом страницы.
<code><section></code>	Тег для логических блоков на странице.
<code><article></code>	Тег для создания самостоятельных единиц контента: от карточек до комментов.
<code><nav></code>	Тег, объединяющий элементы навигации по сайту.
<code></code>	Тег для вставки изображений на страницу.
<code><a></code>	Тег для ссылок
<code><button></code>	Тег для кнопок
<code>//</code>	Упорядоченный список / Неупорядоченный список / Элемент любого из списков
<code><dl>/<dt>/<dd></code>	Три тега для оформления списка терминов. Внешняя обертка / Термин / Значение термина
<code><table>/<tr>/<td></code>	Таблица / Строка таблицы / Ячейка в строке
<code>
</code>	Перенос строки в тексте. Не нужно использовать всегда, только если стилями нельзя добиться такого же соответствия.

`<h1..h6> ... </h1..h6>`

Six different variations of writing a heading.
<h1> has the largest font size, while <h6> has the smallest.

`<div> ... </div>`

A webpage's content is usually divided into blocks, specified by the div tag.

` ... `

This tag injects inline elements, like an image, icon, emoticon without ruining the formatting / styling of the page.

`<p> ... </p>`

Plain text is placed inside this tag.

`
`

A line break for webpages. Is used when wanting to write a new line.

`<hr/>`

Similar to the above tag. But in addition to switching to the next line, this tag also draws a horizontal bar

Document structure

Example

```
<div>
  <h1>Top 5 Greatest Films</h1>
  <p>These are considered the greatest
    <span>reel-icon</span> of all time
  </p>
  <hr/>
  <h2>1. The Godfather</h2>
  <p>This 1972 classic stars Marlon Brando
and Al Pacino.</p>
</div>
```

Text formatting

** ... **

Makes text bold. Used to emphasize a point

** ... **

Alternative to the above tag, also creates bold text.

** ... **

Another emphasis tag, but this displays text in italics.

<i> ... </i>

Also used to display text in italics, but does not emphasize it like the above tag.

<cite> ... </cite>

Tag for citing author of a quote.

** ... **

Pre-formatted, 'monospace' text laid out with whitespace inside the element intact.

<small> ... </small>

Reduces text size. In HTML5, it often refers to redundant or invalid information.

*** – редко используются**

Example

```
<p>
    <strong>Bold text</strong> Regular text
    <em>some words in italics</em> regular
text once again.
</p>
<pre>
    Some pre-formatted text
</pre>

<p>A code snippet:
    <code>some code</code>
</p>
```

Text formatting

<ins> ... </ins>

Denotes text that has been inserted into the webpage.

<blockquote> ... </blockquote>

Quotes often go into this tag. Is used in tandem with the <cite> tag.

<q> ... </q>

Similar to the above tag, but for shorter quotes.

<abbr> ... </abbr>

Denotes abbreviations, along with the full forms.

<address> ... </address>

Tag for specifying author's contact details.

<dfn> ... </dfn>

Tag dedicated for definitions.

<code> ... </code>

This is used to display code snippets within a paragraph.

Example

```
<blockquote>
```

```
    Anyone who has never made a mistake has  
never tried anything new.<cite>- Albert  
Einstein</cite>
```

```
</blockquote>
```

```
<p>A code snippet: <code>some code</code></p>
```

Superscript & Subscript

<sup>

The <sup> element is used to contain characters that should be superscript such as the suffixes of dates or mathematical concepts like raising a number to a power such as 22.

```
<p>On the 4<sup>th</sup> of September you  
will learn about E=MC<sup>2</sup></p>
```

```
<p>The amount of CO<sub>2</sub> in the  
atmosphere grew by 2ppm in 2009 <sub>1</sub>  
</p>
```

On the 4th of September you will learn about $E=MC^2$

The amount of CO₂ in the atmosphere grew by 2ppm in 2009₁

<sub>

The <sub> element is used to contain characters that should be subscript. It is commonly used with foot notes or chemical formulas such as H₂O.

Links

` ... `

Anchor tag. Primarily used for including hyperlinks.

` ... `

Tag dedicated to sending emails.

` ... `

Anchor tag for mentioning contact numbers.
As the numbers are clickable, this can be particularly beneficial for mobile users.

` ... `

This tag can be used to quickly navigate to a different part of the webpage.

` ... `

A variation of the above tag, this is only meant to navigate to a div section of the webpage.

Example

```
<a href="mailto:test@gmail.com">Text to  
our email</a>
```

a vs button

Goes
somewhere

The image shows a vertical rectangular box representing a sign-in form. At the top, the text "Sign in" is displayed. Below it are two input fields: the first contains the placeholder text "email@email.com" and the second contains "*****". Under the password field is a link labeled "Forgot password?". At the bottom of the form is a dark teal rounded rectangular button with the text "Sign in". A blue arrow points from the text "Goes somewhere" to the "Forgot password?" link. Another blue arrow points from the text "Does something" to the "Sign in" button.

Ссылка – нам нужно перейти на другую страницу или в другое место этой же страницы

Кнопка – должно произойти что-то в этом же месте страницы

Does
something

``

A tag to display images in the webpage.

img, figure & figcaption

`<figure>` can be used to contain any content that is referenced from the main flow of an article (not just images).

It is important to note that the article should still make sense if the content of the `<figure>` element were moved (to another part of the page, or even to a different page altogether).

For this reason, it should only be used when the content simply references the element (and not for something that is absolutely integral to the flow of a page).

Examples of usage include:

- Images
- Videos
- Graphs
- Diagrams
- Code samples
- Text that supports the main body of an article

The `<figure>` element should also contain a `<figcaption>` element which provides a text description for the content of the `<figure>` element. In this example, you can see a `<figure>` has been added inside the `<article>` element.

Example

```


<figure>
  
  <figcaption>Bok Choi</figcaption>
</figure>
```


Lists

** ... **

Tag for ordered or numbered list of items.

** ... **

Contrary to the above tag, used for unordered list of items.

** ... **

Individual item as part of a list.

<dl> ... </dl>

Tag for list of items with definitions.

<dt> ... </dt>

The definition of a single term inline with body content.

<dd> ... </dd>

The description for the defined term.

Example

```
<ol>
  <li>Monday</li>
  <li>Tuesday</li>
  <li>Wednesday</li>
</ol>

<ul>
  <li>France</li>
  <li>Germany</li>
  <li>Italy</li>
</ul>

<dl>
  <dt>Toyota</dt>
  <dd>Japanese car brand</dd>

  <dt>Armani</dt>
  <dd>Italian fashion brand</dd>
</dl>
```

Tables

<table> ... </table>

Marks a table in a webpage.

<caption> ... </caption>

Description of the table is placed inside this tag.

<thead> ... </thead>

Specifies information pertaining to specific columns of the table.

<tbody> ... </tbody>

The body of a table, where the data is held.

<tfoot> ... </tfoot>

Determines the footer of the table.

<tr> ... </tr>

Denotes a single row in a table.

<th> ... </th>

The value of a heading of a table's column.

<td> ... </td>

A single cell of a table. Contains the actual value/data.

<colgroup> ... </colgroup>

Used for grouping columns together.

<col>

Denotes a column inside a table.

Example

```
<table>
  <colgroup>
    <col span="2">
    <col>
  </colgroup>
  <tr>
    <th>Name</th>
    <th>Major</th>
    <th>GPA</th>
  </tr>
  <tr>
    <td>Bob</td>
    <td>Law</td>
    <td>3.55</td>
  </tr>
  <tr>
    <td>Alice</td>
    <td>Medicine</td>
    <td>3.61</td>
  </tr>
</table>
```

The `<header>` and `<footer>` elements can be used for:

- The main header or footer that appears at the top or bottom of every page on the site.
- A header or footer for an individual `<article>` or `<section>` within the page.

In this example, the `<header>` element used to contain the site name and the main navigation.

The `<footer>` element contains copyright information, along with links to the privacy policy and terms and conditions. Each individual `<article>` and `<section>` element can also have its own `<header>` and `<footer>` elements to hold the header or footer information for that section within the page.

For example, on a page with several blog posts, each individual post can be thought of as a separate section. The `<header>` element can therefore be used to contain the title and date of each individual post, and the `<footer>` might contain links to share the article on social networking sites.

Please note that all of the code shown in this chapter is referenced in one HTML

Headers & Footers

Example

```
<header>
  <h1>Yoko's Kitchen</h1>
  <nav>
    <ul>
      <li><a href=""
        class="current">home</a></li>
      <li><a href="">classes</a></li>
      <li><a href="">catering</a></li>
      <li><a href="">about</a></li>
      <li><a href="">contact</a></li>
    </ul>
  </nav>
</header>

<footer>
  &copy; 2011 Yoko's Kitchen
</footer>
```

main

`<main>...</main>`

Marks the main content of the webpage.

This content is unique to the individual page, and should not appear elsewhere on the site. Repeating content like headers, footers, navigation, logos, etc., is placed outside the element.

The `<main>` element should only ever be used at most once on a single page.

The `<main>` element must not be included as a descendant of an article, aside, footer, header or nav element.

Example

```
<body>
  <header>
    <nav>...</nav>
  </header>
  <main>
    <h1>Individual Blog Post</h1>
    <p>An introduction for the
post.</p>
    <article>
      <h2>References</h2>
      <p>...</p>
    </article>
    <article>
      <h2>Comments</h2>
      ...
    </article>
  </main>
  <footer>...</footer>
</body>
```

article

<article>...</article>

Denotes an article.

The <article> element acts as a container for any section of a page that could stand alone and potentially be syndicated.

This could be an individual article or blog entry, a comment or forum post, or any other independent piece of content.

If a page contains several articles (or even summaries of several articles), then each individual article would live inside its own <article> element.

The <article> elements can even be nested inside each other. For example, a blog post might live inside one <article> element and each comment on the article could live inside its own child <article> element.

Example

```
<article>
  <h2>Japanese Vegetarian</h2>
  <h3>Five week course in London</h3>
  <p>A five week introduction to traditional
Japanese vegetarian meals, teaching you a
selection of rice and noodle dishes.</p>
</article>

<article>
  <figure>
    
    <figcaption>Teriyaki
Sauce</figcaption>
  </figure>
  <h2>Sauces Masterclass</h2>
  <h3>One day workshop</h3>
  <p>An intensive one-day course looking at
how to
create the most delicious sauces for use in a
range of Japanese cookery.</p>
</article>
```

aside

`<aside> ... </aside>`

Denotes content displayed in a sidebar of the webpage. The `<aside>` element has two purposes, depending on whether it is inside an `<article>` element or not.

When the `<aside>` element is used inside an `<article>` element, it should contain information that is related to the article but not essential to its overall meaning. For example, a pullquote or glossary might be considered as an aside to the article it relates to.

When the `<aside>` element is used outside of an `<article>` element, it acts as a container for content that is related to the entire page. For example, it might contain links to other sections of the site, a list of recent posts, a search box, or recent tweets by the author.

Example

```
<aside>
  <section class="popular-recipes">
    <h2>Popular Recipes</h2>
    <a href="">Yakitori (grilled
chicken)</a>
    <a href="">Tsukune (minced chicken
patties)</a>
    <a href="">Okonomiyaki (savory
pancakes)</a>
    <a href="">Mizutaki (chicken
stew)</a>
  </section>
  <section class="contact-details">
    <h2>Contact</h2>
    <p>Yoko's Kitchen <br />
27 Redchurch Street <br />
Shoreditch<br />
London E2 7DP</p>
  </section>
</aside>
```

`<section>...</section>`

Specifies a particular section in the webpage.

The `<section>` element groups related content together, and typically each section would have its own heading.

For example, on a homepage there may be several `<section>` elements to contain different sections of the page, such as latest news, top products, and newsletter signup.

Because the `<section>` element groups related items together, it may contain several distinct `<article>` elements that have a common theme or purpose.

Alternatively, if you have a page with a long article, the `<section>` element can be used to split the article up into separate sections.

The `<section>` element should not be used as a wrapper for the entire page (unless the page only contains one distinct piece of content). If you want a containing element for the entire page, that job is still best left to the `<div>` element.

section

Example

```
<section class="popular-recipes">
  <h2>Popular Recipes</h2>
  <a href="">Yakitori (grilled chicken)</a>
  <a href="">Tsukune (minced chicken
patties)</a>
  <a href="">Okonomiyaki (savory
pancakes)</a>
  <a href="">Mizutaki (chicken stew)</a>
</section>

<section class="contact-details">
  <h2>Contact</h2>
  <p>Yoko's Kitchen <br />
27 Redchurch Street <br />
Shoreditch<br />
London E2 7DP</p>
</section>
```

nav

`<nav>...</nav>`

Navigation links for the user in a webpage.

The `<nav>` element is used to contain the major navigational blocks on the site such as the primary site navigation.

Going back to our blog example, if you wanted to finish an article with links to related blog posts, these would not be counted as major navigational blocks and therefore should not sit inside a `<nav>` element.

At the time of writing, some of the developers that were already using HTML5 decided to use the `<nav>` element for the links that appear at the bottom of every page (links to things like privacy policy, terms and conditions and accessibility information).

Whether this will be widely adopted is still yet to be seen.

Example

```
<nav>
  <ul>
    <li><a href=""
      class="current">home</a></li>
    <li><a href="">classes</a></li>
    <li><a href="">catering</a></li>
    <li><a href="">about</a></li>
    <li><a href="">contact</a></li>
  </ul>
</nav>
```


Semantic tags

<dialog>...</dialog>

Used to create a dialog box.

<mark>...</mark>

Used to highlight a particular portion of the text.

<rt>...</rt>

Displays East Asian typography character details.

<time>...</time>

Tag for formatting date and time.

<wbr>

A line-break within the content.

<details> ... </details>

Used for additional information. User has the option to view or hide this.

<summary> ... </summary>

Used as a heading for the above tag. Is always visible to the user.

Example

```
<dialog>
  <button autofocus>Close</button>
  <p>This modal dialog has a groovy backdrop!</p>
</dialog>
<button>Show the dialog</button>

<time datetime="2016-03-13">13th March
2016</time>

<details>
  <summary>Details</summary>
  Something small enough to escape casual notice.
</details>
```

HTML

Атрибуты

Атрибуты содержат дополнительную информацию об элементе, которую вы не хотите показывать в контенте.

Атрибут всегда должен иметь:

- Пробел между ним и именем элемента (или предыдущим атрибутом, если элемент уже имеет один или несколько атрибутов).
- Имя атрибута(name), за которым следует знак равенства.
- Значение атрибута(value), заключенное с двух сторон в кавычки. Значений может быть несколько
- Порядок следования атрибутов не важен

Типы

Атрибутов

Глобальные (global)

можно применить к любому элементу

```
<p class="box error">Setting 'box' class to the  
element.</p>
```

Обязательные (required)

без них элемент не валиден

```

```

Логические (boolean)

не требуют значения

```
<button disabled>You can't click on this button</button>
```

Пользовательские (custom)

можно создать самому, чтобы потом
использовать в js с помощью свойства

dataset[name]

data-your-name

```
<button data-machine-learning="workshop">Click</button>
```

Самые используемые Атрибуты

href	<code><a></code>	URL для ссылки
target="_blank"	<code><a></code>	Отвечает за открытие ссылки в новом окне
class	все	Используется вместе с CSS, чтобы стилизовать элементы.
id	все	Значение атрибута должно быть уникальным на странице, используется для ссылок на блоки одной страницы.
alt / src	<code></code>	Альтернативный текст, в случае, если изображение не может быть отображено. / Путь к изображению, без него картинка не отобразится.
width/height	<code></code> , <code><iframe></code>	Задаёт дефолтную высоту/ширину изображениям, которую можно переопределить в CSS.
style	все	Определяет стили, которые перепишут установленные ранее в CSS файле стили.

Атрибут

class

- название класса может быть произвольным
- хороший класс должен отвечать на вопрос "Что делает этот элемент?"
- не нужно задавать имя класса **по контенту, стилю или по тегу**, так как они могут поменяться
- не нужно использовать в названиях классов **цифры или числительные**, порядок блоков может измениться
- используются только **английские слова**, не должно быть **транслита** с русского языка
- пишется **маленькими буквами**
- в имени класса не должно быть пробелов(это будет два разных класса), используется **тире** для составных классов
- лучше не использовать **больше трех слов** в названии класса
- у одного элемента может быть много классов

Комментарии в HTML

```
<div>
  <!-- HTML Comment
  This comment will not be seen in the browser window but
  is sent to the browser which can add to the delay of a
  page load. -->
```

- Невидимы для пользователя
- Пояснения в коде для себя в будущем или для коллег
- Скрывают блоки кода со страницы, не удаляя их из файла
- Нельзя вкладывать один комментарий в другой
- Чтобы быстро закомментировать или раскомментировать текущую строку или выделенный блок кода, в большинстве редакторов можно нажать **Ctrl /** или **Cmd /**
- Комментарии часто используют, чтобы пометить начало какого-то крупного куска кода, пояснить его назначение
- Комментарии видны в браузере в инструментах разработчика, поэтому в них надо писать только то, что может увидеть кто угодно

2 Иерархия заголовков

— составление дерева заголовков, которое должно выглядеть как оглавление книги.

1. Основа контента

2. Зачем об этом думать:

a. Причины как и у семантики

3. Если на странице не хватает визуальных заголовков, то нужно добавить скрытые

Contents

Chapter 1-Basic of InDesign	1
History of InDesign	1
InDesign in Ancient Greek	1
The Birth of InDesign	1
The Death of InDesign	2
The Best Publishing Software	2
What About Quark	3
What About Office	3
Why Use inDesign?	3
I Love InDesign	4
I Also Love Photoshop	4
Who Create InDesign?	4
 Chapter 2-Typography.....	 5
Definition of Typography	5
Typography is Text In Visual	5
Typography as Art	5
Dealing With Typography	6
Mr. Typography Himself	6

<h1>InDesign book</h1>

<h2>Basics of InDesign</h2>

<h3>History of InDesign</h3>

<h4>InDesign in Ancient Greek</h4>

<h4>The Birth of InDesign</h4>

<h4>The Death of InDesign</h4>

<h3>The best publishing software</h3>

<h4>What about Quark</h4>

<h4>What about Office</h4>

<h3>Whu use InDesign?</h3>

<h4>I love InDesign</h4>

<h4>I also love Photoshop</h4>

<h3>Who Create InDesign?</h3>

<h2>Typography</h2>

<h3>Defenition of Typography</h3>

<h4>Typography is Text in Visual</h4>

<h4>Typography is Art</h4>

<h3>Dealing with Typography</h3>

<h4>Mr. Typography Himself</h4>

3 Валидность

— это соответствие стандарту (спецификации)

The World Wide Web Consortium(W3C) – это международное сообщество, которое разрабатывает открытые стандарты для интернета.

1. Правильный порядок открывающих и закрывающих тегов
2. Наличие закрывающих тегов
3. Наличие всех обязательных атрибутов
4. Правильное использование тегов внутри других тегов

1. Правильный порядок открывающих и закрывающих тегов

`<p> Lorem ipsum <a> dolor set </p> `



`<p> Lorem ipsum <a> dolor set </p>`



`<p> Lorem ipsum</p> <a> dolor set `



2. Наличие закрывающих тегов

`` Lorem ipsum dolor set
`` Lorem ipsum dolor set



`` Lorem ipsum dolor set ``
`` Lorem ipsum dolor set ``



``
`<input type="text">`



3. Наличие всех обязательных атрибутов

``



``



`<a> Lorem ipsum `



`<div id="value01"></div>`

`<div id="value01"></div>`



4. Правильное использование тегов внутри других тегов

` Lorem ipsum <div> dolor sit </div> `



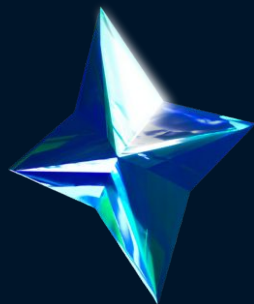
`<div> Lorem ipsum dolor sit </div>`



` Lorem ipsum
 <p> dolor sit </p>
`



` Lorem ipsum
 <p> dolor sit </p>
`



4 Доступность

— это удобство использования сайта для людей с дополнительными потребностями

1. Обычно никто об этом не думает
2. Если соблюдать остальные три правила, то доступность появится автоматически
3. Можно проверить - **chrome - audits - accessibility**, еще и расскажет как исправить
4. Людям будет приятно пользоваться вашим сайтом с помощью клавиатуры и скринридера

Подсказки

Emmet

!	базовая структура сайта	<!DOCTYPE html> <html lang="en"> ... </html>
lorem	первые 5 слов рыбного текста	Lorem ipsum dolor sit amet.
#example	id	<div id="page"></div>
.example	class	<div class="logo"></div>
p*2	количество элементов	<p></p> <p></p>
ul>li	вложенность	
ul>li.item\$*5	счетчик	<li class="item1"> ... <li class="item5">
p>span+em^bq = (^+)	подняться на уровень выше	<p></p> <blockquote></blockquote>
div>(header>p)+footer	группировка	<div> <header><p></p></header> <footer></footer> </div>
a[href="index.html"]	атрибут	
a{Click me}	текст	<a>Click me

Правила

Emmet

класс без тега = div с классом	<code>.wrap>.content = div.wrap>div.content</code>
если у элемента есть правила вложенности, то они применяются автоматически	<code>ul>.item*3 = ul>li.item*3</code>
<ul style="list-style-type: none">li for ul and ol	<code>ol>.item = ol>li.item</code>
<ul style="list-style-type: none">tr for table, tbody, thead and tfoot	<code>table>.row = table>tr.row</code>
<ul style="list-style-type: none">td for tr	<code>table>tr>.cell = table>tr>td.cell</code>
<ul style="list-style-type: none">option for select and optgroup	<code>select>.select-item*5 = select>option.select-item*5</code>
соблюдаются правила вложенности по типам элементов	<code>em>.info = em>span.info div</code>