

УНИВЕРСИТЕТ ИТМО

Факультет «инфокоммуникационных технологий»
Направление подготовки «Программирование в инфокоммуникационных
системах»

Лабораторная работа №1
«Основы работы с HTML»

Выполнил:
Егорова Валерия Игоревна
Группа №3323
Проверил:
Марченко Елена Вадимовна

Санкт-Петербург
2024

Цель работы

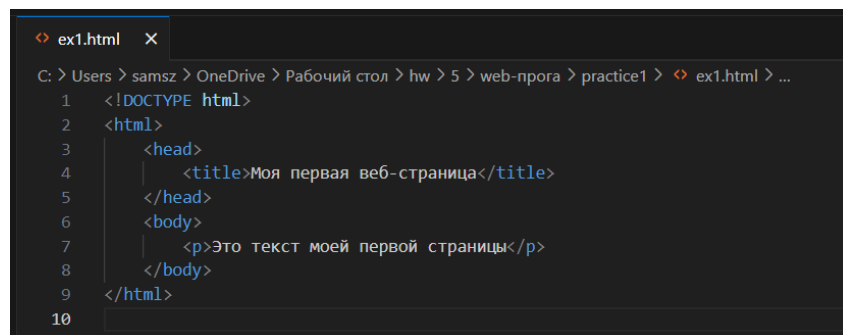
Изучить основы работы с HTML и научиться создавать веб-страницы с использованием различных тэгов, гиперссылок, изображений и медиафайлов.

Ход работы

1. Структура HTML-кода

Упражнение 1. Создание первой HTML-страницы

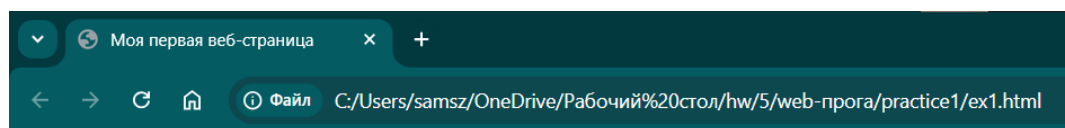
В первом упражнении был создан файл ex1.html с базовой структурой HTML-документа, включающей тэги `<html>`, `<head>`, `<title>`, `<body>`. На страницу также был добавлен абзац с помощью тэга `<p>` (см. рисунок 1).



```
ex1.html X
C: > Users > samsz > OneDrive > Рабочий стол > hw > 5 > web-npora > practice1 > ex1.html > ...
1  <!DOCTYPE html>
2  <html>
3    <head>
4      <title>Моя первая веб-страница</title>
5    </head>
6    <body>
7      <p>Это текст моей первой страницы</p>
8    </body>
9  </html>
10
```

Рисунок 1 – Код для упражнения 1

Файл был сохранен и открыт в браузере. Текст, указанный внутри тэга `<title>`, является заголовком страницы, а текст из тела файла отображается на самой странице (см. рисунок 2).



Это текст моей первой страницы

Рисунок 2 – Отображение страницы в браузере

Также посмотрели код отображаемой страницы через браузер (см. рисунок 3).

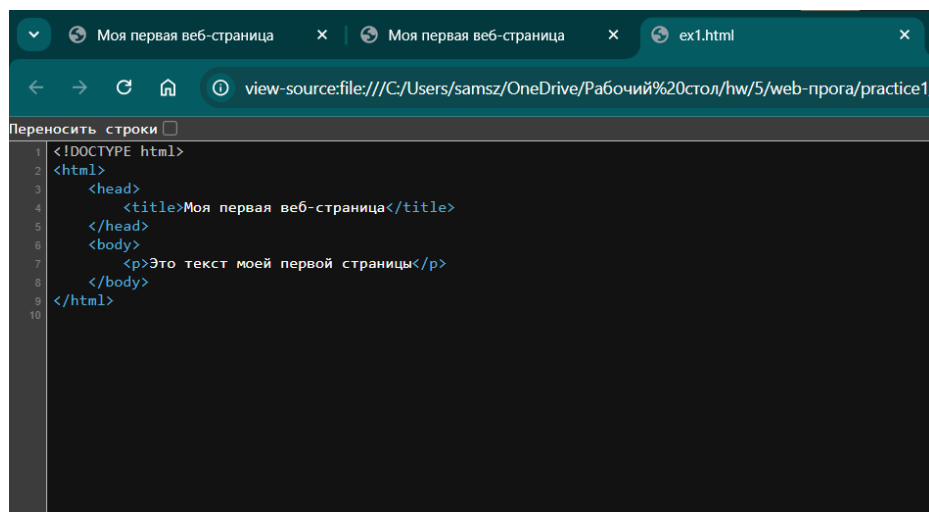


Рисунок 3 – Просмотр кода страницы через браузер

Упражнение 2. Тэги верхнего уровня и заголовка документа

Во втором упражнении были рассмотрены тэги «верхнего уровня» `<html>`, `<head>` и `<body>`, тэги заголовка документа `<title>` и `<meta>`, изучено использование комментариев в коде, а также был добавлен тэг `<meta>` для указания кодировки (см. рисунок 4).

```
1  <!-- Таким образом создаются комментарии -->
2  <!-- Комментарии могут быть в любой части кода HTML -->
3  <!DOCTYPE html>
4  <html>
5    <head>
6      <!--Добавляем тэг meta -->
7      <meta charset="utf-8">
8      <title>Моя первая веб-страница</title>
9    </head>
10   <body>
11     <!--Добавляем тэг p, который определяет текстовый абзац -->
12     <p>Это текст моей первой страницы</p>
13   </body>
14 </html>
```

Рисунок 4 – Код для упражнения 2

Заметим, что видимых изменений на веб-странице не произошло (см. рисунок 5), однако комментарии для удобства разработки в коде присутствуют, в чем можно убедиться, просмотрев код страницы через браузер (см. рисунок 6).

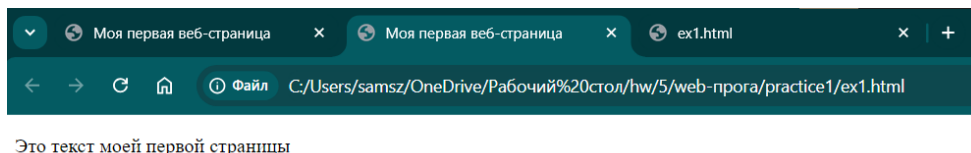


Рисунок 5 – Отображение страницы в браузере

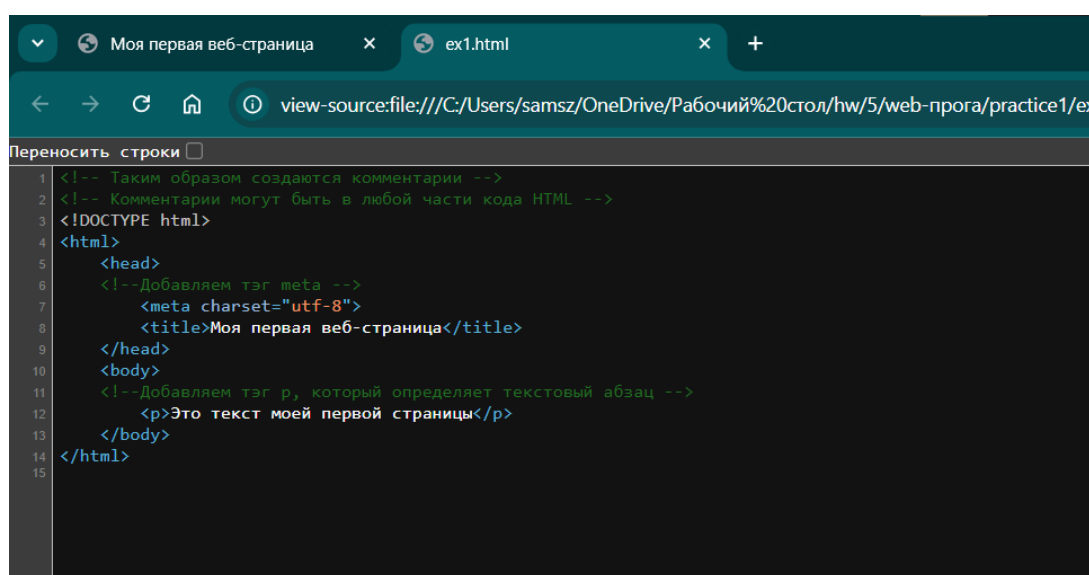


Рисунок 6 – Просмотр кода страницы через браузер

Упражнение 3. Атрибуты HTML-тэгов

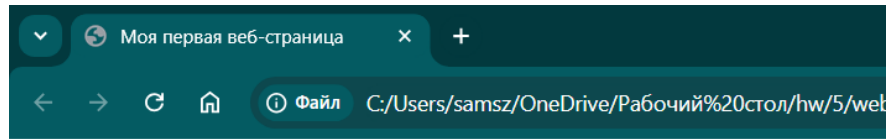
В упражнении 3 были изучены атрибуты тэгов. Для этого в код была добавлена гиперссылка с помощью тэга `<a>` и атрибута `href` (см. рисунок 7).

```

1  <!-- Таким образом создаются комментарии -->
2  <!-- Комментарии могут быть в любой части кода HTML -->
3  <!DOCTYPE html>
4  <html>
5      <head>
6          <!--Добавляем тэг meta -->
7          <meta charset="utf-8">
8          <title>Моя первая веб-страница</title>
9      </head>
10     <body>
11         <!--Добавляем тэг p, который определяет текстовый абзац -->
12         <p>Это текст моей первой страницы</p>
13         <a href = "http://www.w3.org">Это гиперссылка на сайт Консорциума Всемирной паутины</a>
14     </body>
15 </html>
  
```

Рисунок 7 – Добавление гиперссылки в код

При просмотре страницы в браузере видно, что гиперссылка отобразилась на странице (см. рисунок 8).



Это текст моей первой страницы

[Это гиперссылка на сайт Консорциума Всемирной паутины](http://www.w3.org)

Рисунок 8 – Отображение гиперссылки на странице

При нажатии на нее открывается сайт Консорциума Всемирной паутины (см. рисунок 9).

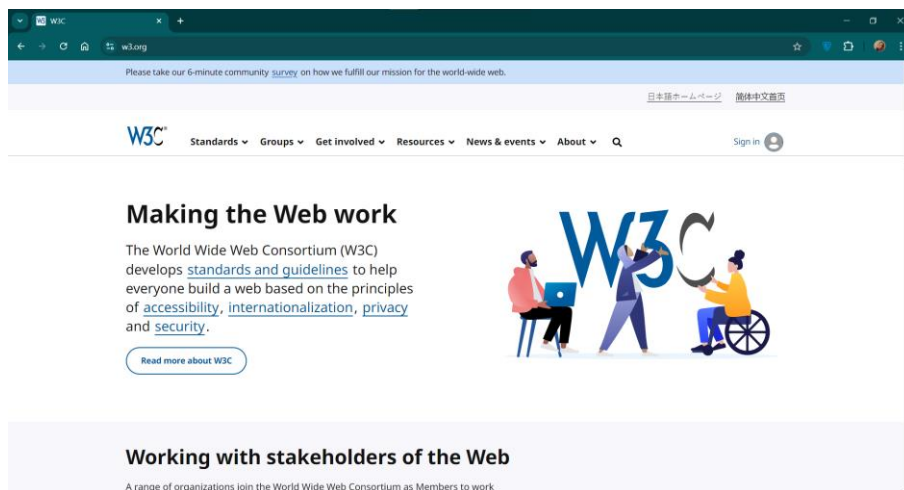


Рисунок 9 – Сайт Консорциума Всемирной паутины

Следующим шагом было необходимо добавить изображение с атрибутами src, alt, width и height, определяющими адрес изображения, его альтернативный текст, ширину и высоту соответственно (см. рисунок 10).

```
1 <!-- Таким образом создаются комментарии -->
2 <!-- Комментарии могут быть в любой части кода HTML -->
3 <!DOCTYPE html>
4 <html>
5   <head>
6     <!--Добавляем тэг meta -->
7     <meta charset="utf-8">
8     <title>Моя первая веб-страница</title>
9   </head>
10  <body>
11    <!--Добавляем тэг p, который определяет текстовый абзац -->
12    <p>Это текст моей первой страницы</p>
13    <a href = "http://www.w3.org">Это гиперссылка на сайт Консорциума Всемирной паутины</a>
14    
15  </body>
16 </html>
```

Рисунок 10 – Код с добавлением изображения

При просмотре страницы в браузере заметим, что изображение располагается на одной строке с гиперссылкой (см. рисунок 11).

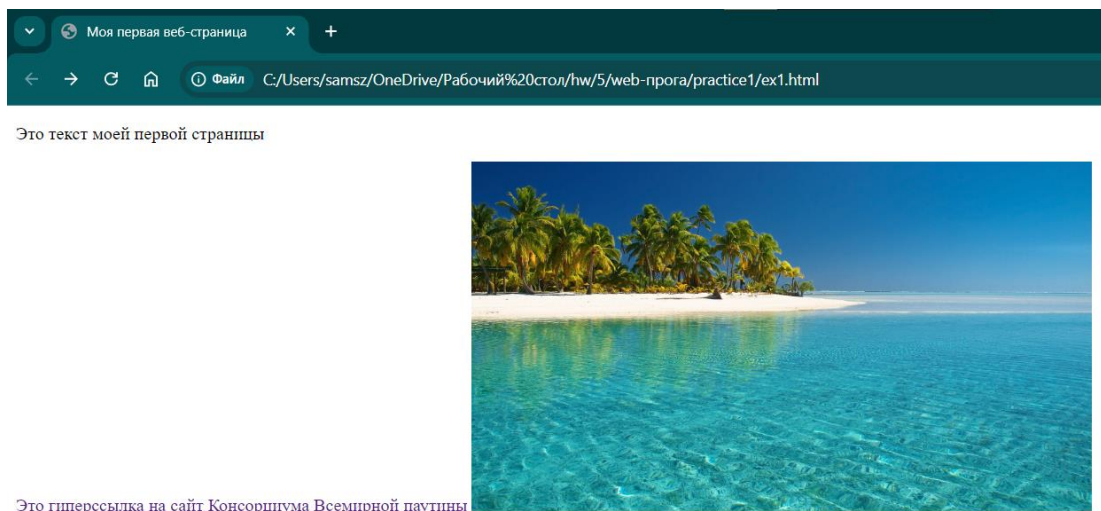


Рисунок 11 – Отображение картинки на странице

Чтобы расположить гиперссылку и изображение на разных строках, используем непарный тэг `
` (см. рисунок 12). Он добавляет перенос строки в том месте, где находится.

```
<a href = "http://www.w3.org">Это гиперссылка на сайт Консорциума Всемирной паутины</a><br/>  

```

Рисунок 12 – Добавление тэга `
`

Результат применения тэга `
` представлен на рисунке 13.

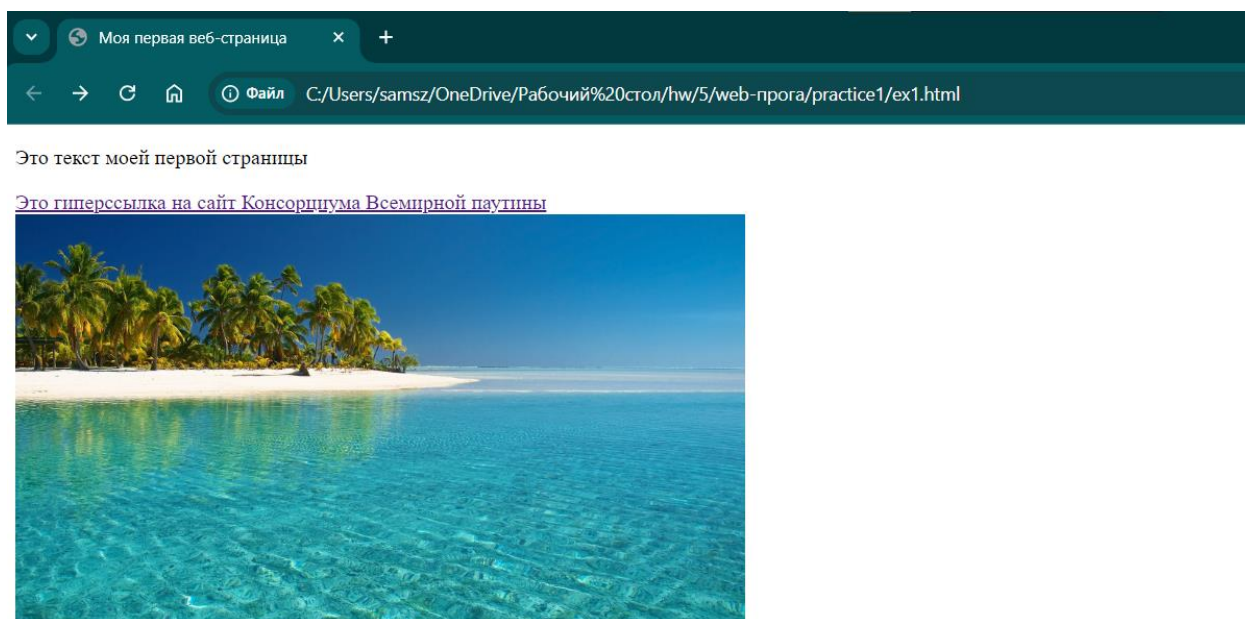


Рисунок 13 – Отображение переноса строки на странице

Далее к открывающему тэгу <p> добавим информационный атрибут title (см. рисунок 14).

```
<p title = "information">Это текст моей первой страницы</p>
```

Рисунок 14 – Добавление атрибута <title>

В результате при наведении курсора на текст появляется подсказка с надписью “information” (см. рисунок 15).

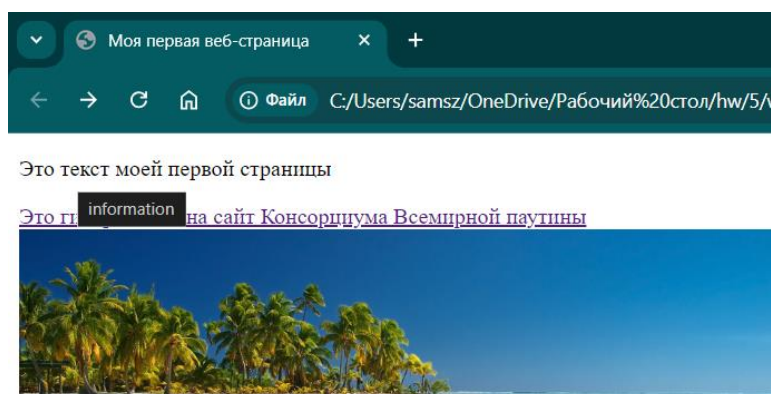


Рисунок 15 – Отображение подсказки при наведении на текст

Упражнение 4. Устаревшие HTML-атрибуты

Рассмотрим атрибуты, которые на данный момент считаются устаревшими, поскольку относятся к визуальной разметке текста, которая была перенесена в CSS.

Создадим файл ex2.html и скопируем в него код из предыдущего упражнения, удалим все комментарии и атрибут title из тэга <p>.

Далее перейдем к добавлению новых атрибутов. Необходимо добавить атрибуты изменения цвета фона и текста к <body>, а также атрибут выравнивания элемента align и вложенный парный элемент для изменения гарнитуры к тэгу <p> (см. рисунок 16).

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8">
5     <title>Моя первая веб-страница</title>
6   </head>
7   <body bgcolor="SeaGreen" text="#800080">
8     <p align="center"><font face="Arial" size="7">Это текст моей первой страницы</font></p>
9     <a href = "http://www.w3.org">Это гиперссылка на сайт Консорциума Всемирной паутины</a><br/>
10    
11  </body>
12 </html>
```

Рисунок 16 – Добавление bgcolor, text, align и font

Открыв страницу в браузере, убедимся, что вследствие применения атрибутов bgcolor и text цвета фона и текста документа изменились, из-за применения align текст абзаца стал отображаться в центре страницы, а в силу добавления font размер шрифта увеличился и гарнитура изменилась на Arial (см. рисунок 17).

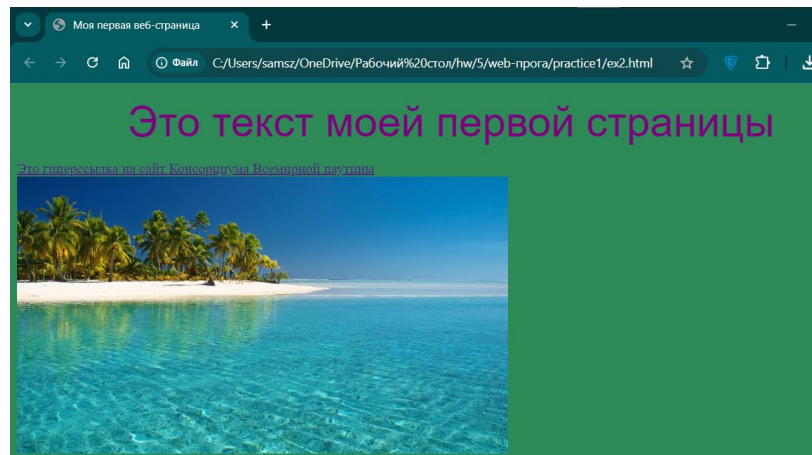


Рисунок 17 – Отображение изменений в браузере

Теперь осуществим изменение внешнего вида страницы с помощью каскадных таблиц стилей CSS. В файле ex3.html добавим в тэги <body> и <p> код стилового оформления (см. рисунок 18).

```
<body style = "background-color: SeaGreen; color: #800080">  
  <p style = "font: 3em Arial; text-align: center">Это текст моей первой страницы</p>
```

Рисунок 18 – Добавление стилового оформления с применением style

Откроем страницу в браузере и обратим внимание, что стили также применились (см. рисунок 19).

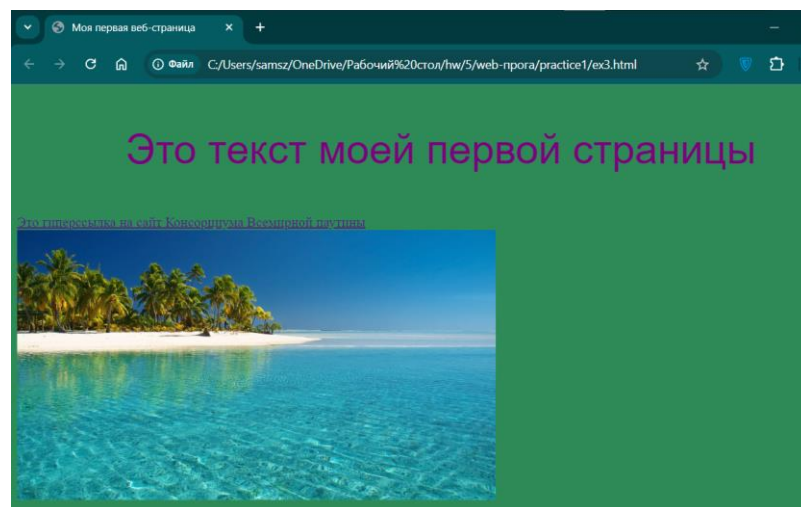


Рисунок 19 – Отображение изменений в браузере

Упражнение 5. Основные особенности работы с текстом в HTML

Рассмотрим основные особенности работы с текстом в HTML. Добавим в файл абзац текста, добавив в него перенос строк и произвольное количество идущих подряд пробельных символов (см. рисунок 20).

```
<p>CSS -  
это      язык      описания  
внешнего вида      документа,  
написанного  
с      использованием  
языка      разметки</p>
```

Рисунок 20 – Добавленный в код абзац

Откроем страницу в браузере и обратим внимание на то, что абзац отображается так, будто он написан в одну строку без лишних пробелов, поскольку HTML не поддерживает переход к новому абзацу с помощью клавиши «Enter», а любое количество идущих подряд пробелов в браузере отображается как один (см. рисунок 21).

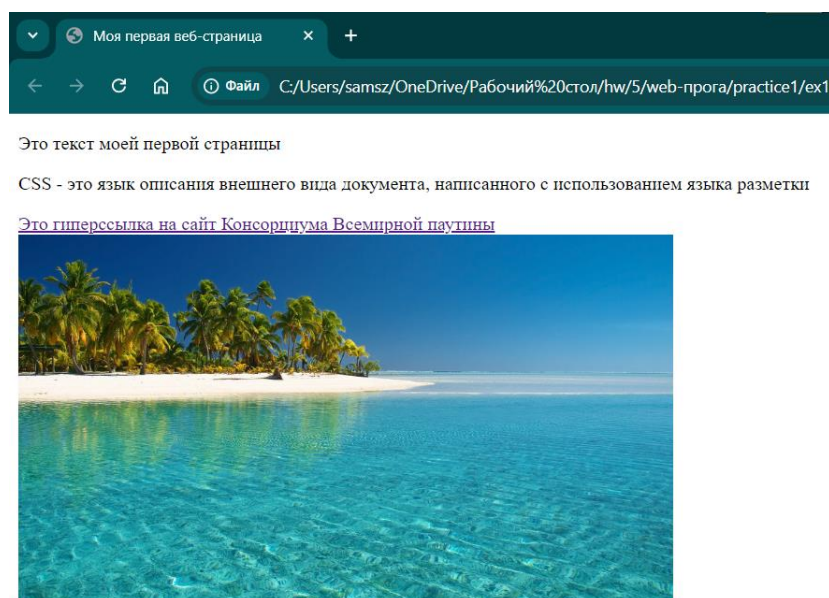


Рисунок 21 – Отображение нового абзаца в браузере

Вернемся к исходному коду и заменим тэг `<p>` на `<pre>` (см. рисунок 22).

```
<pre>CSS -  
это      язык      описания  
внешнего вида      документа,  
написанного  
с      использованием  
языка      разметки</pre>
```

Рисунок 22 – Изменение тэга `<p>` на `<pre>`

Открыв страницу в браузере, заметим, что количество отображаемых в браузере пробельных символов и переходов на новую строку теперь совпадает с их количеством в исходном документе, поскольку примененный тэг `<pre>` отображает предварительно отформатированный текст с сохранением всех пробелов и переносов (см. рисунок 23).

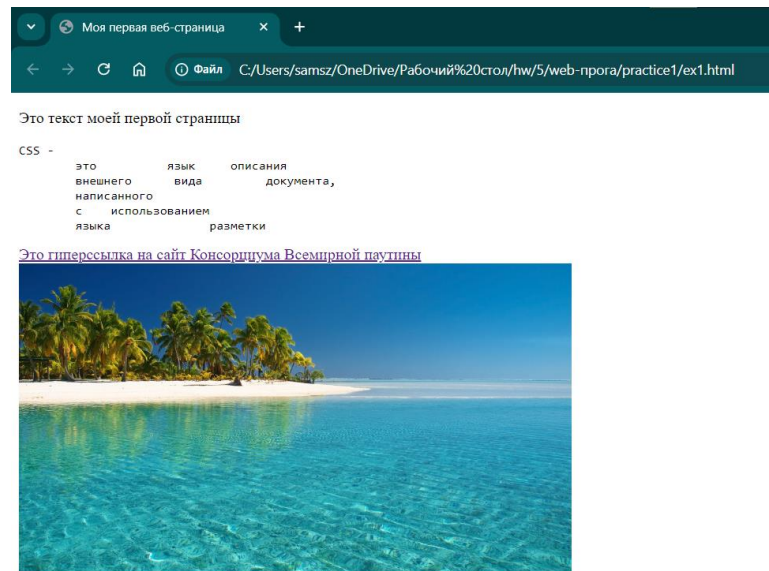


Рисунок 23 – Отображение текста в тэге <pre>

Также проверим, что при добавлении длинной строки с пробелами браузер разместит текст по ширине окна, автоматически перенеся строки, а в случае отсутствия пробелов появится полоса прокрутки. Для этого добавим на страницу произвольный набор символов с добавлением пробелов (см. рисунок 24), а затем без добавления пробелов (см. рисунок 25).

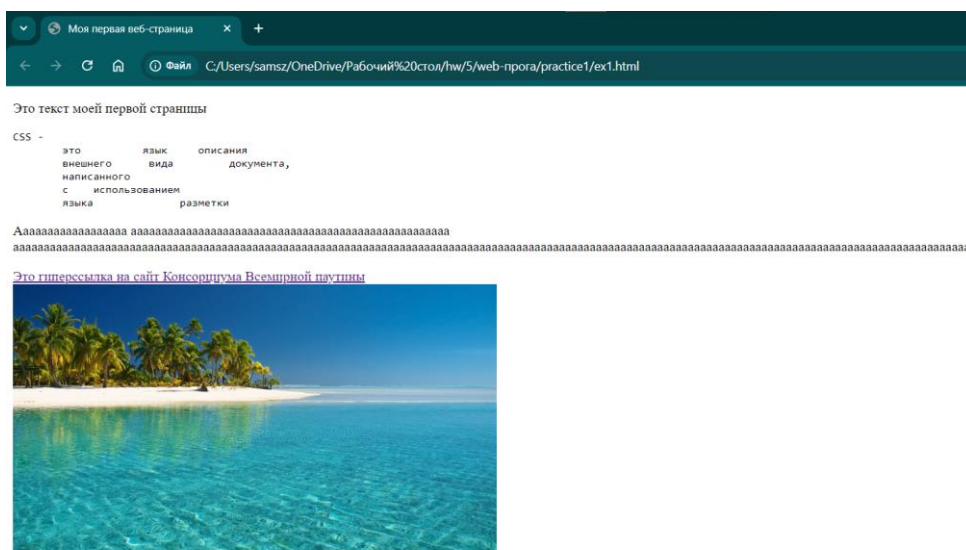


Рисунок 24 – Страница со строкой символов с пробелами

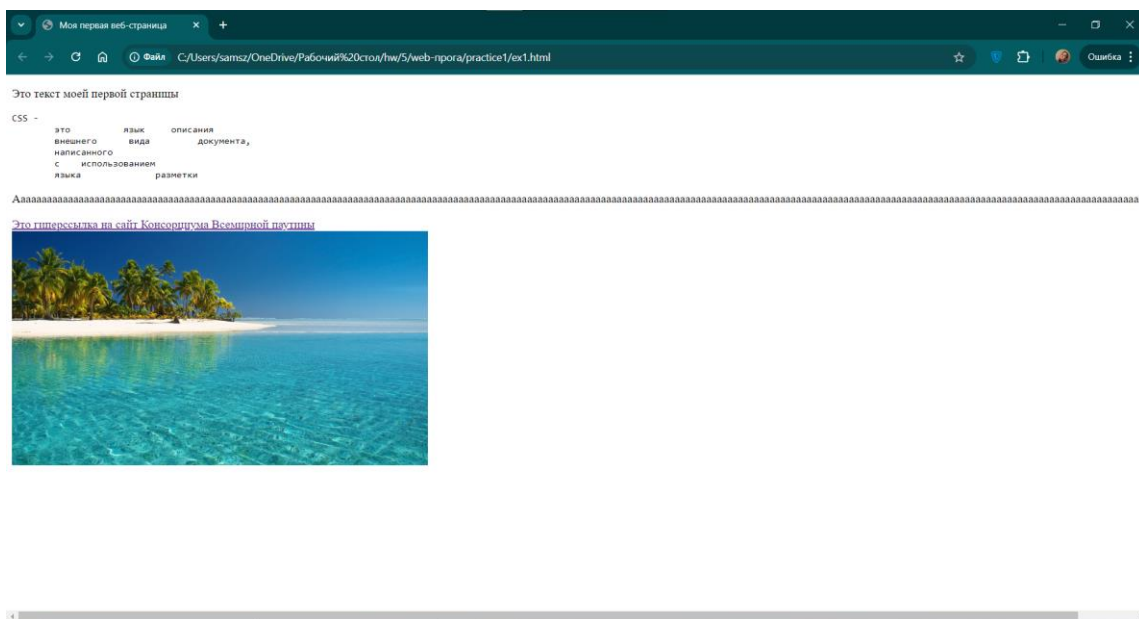


Рисунок 25 – Страница со строкой символов без пробелов

Упражнение 6. Использование спецсимволов

Рассмотрим применение символов, которые являются специализированными для HTML или которых нет на клавиатуре. Для этого в файл добавим текст с добавлением символов (см. рисунок 26) и посмотрим на их отображение, открыв страницу в браузере (см. рисунок 27).

```
<p> Цену товаров компании &quot;QWERTY &trade;&quot; необходимо пересчитать  
с учетом курса &euro; к &pound; &plusmn; наценка для региона</p><br>
```

Рисунок 26 – Добавление текста со спецсимволами

Цену товаров компании "QWERTY ™" необходимо пересчитать с учетом курса € к £ ± наценка для региона

Рисунок 27 – Отображение текста со спецсимволами в браузере

2. Основы логического форматирования контента

Упражнение 1. Использование заголовков

Рассмотрим применение заголовков в HTML. Для этого создадим файл, добавим в него заголовки разных уровней с помощью тэгов `<h1>...</h1>` – `<h6>...</h6>` (см. рисунок 28) и проверим их отображение в браузере (см. рисунок 29).

```

3      <head>
4      |   <title>Работа с текстом</title>
5      | </head>
6      | <body>
7      |   <h1>This is heading 1</h1>
8      |   <h2>This is heading 2</h2>
9      |   <h3>This is heading 3</h3>
10     |   <h4>This is heading 4</h4>
11     |   <h5>This is heading 5</h5>
12     |   <h6>This is heading 6</h6>
13     | </body>
14 </html>

```

Рисунок 28 – Код для добавления заголовков

This is heading 1

This is heading 2

This is heading 3

This is heading 4

This is heading 5

This is heading 6

Рисунок 29 – Отображение заголовков на странице в браузере

Упражнение 2. Возможности логического форматирования текста

Рассмотрим возможности форматирования текста в HTML. Для этого в файл добавим текст про пуговицы и отформатируем его с помощью тэгов (см. рисунок 30).

```

<body>
<h1>000 "Pora" & K<sup>o</sup>"</h1>
<h2><i>Наши пуговицы - Ваше все!</i></h2>

<p>Мы осуществляем выпуск <em>всех</em> разновидностей <b>пуговиц</b>,
изготавливаем пришивные пуговицы на ножке или традиционные,
<b>несколькими</b> отверстиями посередине.</p>

<p>Нашу продукцию можно приобрести по <del>всему Северо-Западному региону</del>
<ins>по всей России</ins> <b>представителей</b> компании.</p>

<p>Кроме того, именно <b>нас</b> вы можете сделать заказ на <strong>эксклюзивные изделия</strong>,
которые будут только на вашей одежде. Мы также выполняем заказы различных компаний
на производство пуговиц <mark><b>логотипами</b> или любыми другими надписями и рисунками</mark>.
Посетите наш каталог, и вы обязательно найдете для себя изделия по душе!</p>

<h2>Интересное <b>пуговицах</b></h2>

<p>Пуговица – небольшой предмет любой формы <b>отверстиями</b> или ушком для пришивания к одежде.
<b>В</b> более узком варианте – застёжка на одежде и других швейных изделиях,
предназначенная для соединения её частей (пуговица на одной части одежды вдевается в петлю,
находящуюся на другой части, и тем самым осуществляется застёгивание).</p>

<p>Коллекционирование пуговиц называется филобутонистика.</p>

<p>Знаете ли вы, что Первая Мировая война случилась из-за плохо расстегивающихся пуговиц?
Эрцгерцог Франц-Фердинанд умер после выстрела только из-за того, что на расстегивание
всех <b>erg</b> пуговиц для обработки раны потребовалось слишком много времени.</p>

<p><b>Наша контактная информация:</b></p>
<p>Россия, <br />
Санкт-Петербург, <br />
ул.Ленина, д.1<br />
(812)123-4567
</p>

```

Рисунок 30 – Код с форматированием текста при помощи тэгов

Посмотрим на получившийся результат в браузере и заметим, что были применены различные начертания, зачеркивание, подчеркивание, выделение текста (см. рисунок 31). Тэг отобразил жирное начертание, – смысловое ударение, <i> сделал шрифт курсивным, добавил акцентирование текста, <sup> отобразил надстрочный текст, <ins> выделил добавленный в новую версию документа текст, зачеркнул текст, а <mark> выделил цветом.

ООО "Рога & Коп"

Наши пуговицы - Ваше все!

Мы осуществляем выпуск всех разновидностей **пуговиц**, изготавливаем пришпывные пуговицы на ножке или традиционные, с несколькими отверстиями посередине.

Нашу продукцию можно приобрести по ~~всему Северо-Западному региону~~ по всей России у представителей компании.

Кроме того, именно у нас вы можете сделать заказ на **эксклюзивные изделия**, которые будут только на вашей одежде. Мы также выполняем заказы различных компаний на производство пуговиц **с логотипами или любыми другими надписями и рисунками**. Посетите наш каталог, и вы обязательно найдете для себя изделия по душе!

Интересное о пуговицах

Пуговица — небольшой предмет любой формы с отверстиями или ушком для пришивания к одежде. В более узком варианте — застёжка на одежде и других швейных изделиях, предназначенная для соединения её частей (пуговища на одной части одежды вдевается в петлю, находящуюся на другой части, и тем самым осуществляется застёгивание).

Коллекционирование пуговиц называется филобутонистикой.

Знаете ли вы, что Первая Мировая война случилась из-за плохо расстегивающихся пуговиц? Эрихгерцог Франц-Фердинанд умер после выстрела только из-за того, что на расстегивание всех его пуговиц для обработки раны потребовалось слишком много времени.

Наша контактная информация:

Россия,
Санкт-Петербург,
ул.Ленина, д.1
(812)123-4567

Рисунок 31 – Отображение отформатированного текста в браузере

Упражнение 3. Ссылки, цитаты, определения

Рассмотрим использование ссылок, цитат и определений. Для этого в отформатированный в упражнении 2 текст внесем дополнения: <dfn>, <blockquote> и <address> (см. рисунок 32). <dfn> добавит определение при наведении на текст, <blockquote> создаст цитату, а <address> выделит контактную информацию (см. рисунок 33).

```
<p>
  Коллекционирование пуговиц называется <dfn title = "Коллекционирование пуговиц">филобутонистикой</dfn>.
</p>

<p>Знаете ли вы, что Первая Мировая война случилась из-за плохо расстегивающихся пуговиц?
<blockquote cite="http://glamik.ru/blog/2012-03-07-640">Эрихгерцог Франц-Фердинанд
умер после выстрела только из-за того, что на расстегивание всех его пуговиц для обработки раны
потребовалось слишком много времени.</blockquote></p>

<p><b>Наша контактная информация:</b></p>
<p>
<address>
  Россия, <br />
  Санкт-Петербург, <br />
  ул.Ленина, д.1<br />
  (812)123-4567
</address>
</p>
```

Рисунок 32 – Добавление ссылок, цитат и определений в код

Коллекционирование пуговиц называется *филобутонистика*.

Знаете ли вы, что Первая Мировая война случилась из-за Коллекционирование пуговиц пуговиц?

Эрцгерцог Франц-Фердинанд умер после выстрела только из-за того, что на расстегивание всех его пуговиц для обработки раны потребовалось слишком много времени.

Наша контактная информация:

Россия,
Санкт-Петербург,
ул.Ленина, д.1
(812)123-4567

Рисунок 33 – Отображение ссылок, цитат и определений в браузере

Упражнение 4. Элементы компьютерного кода

Рассмотрим форматирование элементов компьютерного кода. Для отформатированного представления элементов, связанных с компьютерным кодом, используются тэги `<kbd>`, `<samp>`, `<code>`, `<var>`. `<kbd>` используется для обозначения текста, которые набирается на клавиатуре, `<samp>` – для отображения результата вывода программы, `<code>` – для отображения программного кода, `<var>` – для объявления переменных.

Создадим новый файл и используем в нем описанные выше тэги (см. рисунок 34), а затем посмотрим, как они отображаются при открытии страницы в браузере (см. рисунок 35).

```
1  <!DOCTYPE html>
2  <html>
3    <head>
4      <title>Элементы компьютерного кода</title>
5    </head>
6    <body>
7
8      <p>Для обозначения ввода q клавиатуры используется тэг <kbd></p>
9      <p><kbd>File | Save...</kbd></p>
10
11     <samp>
12       demo.info Jan 2015 09:10:17
13       Linux -grsec+gg3+e+gr2b-reslog-
14     </samp>
15
16     <p>В данном примере делается запрос к базе данных
17     для получения поля <var>content_title</var>
18     из таблицы <var>content</var>, причем
19     должно удовлетворяться условие <code>section_id = 1</code>.</p>
20
21   </body>
22 </html>
```

Рисунок 34 – Применение тэгов форматирования компьютерного кода

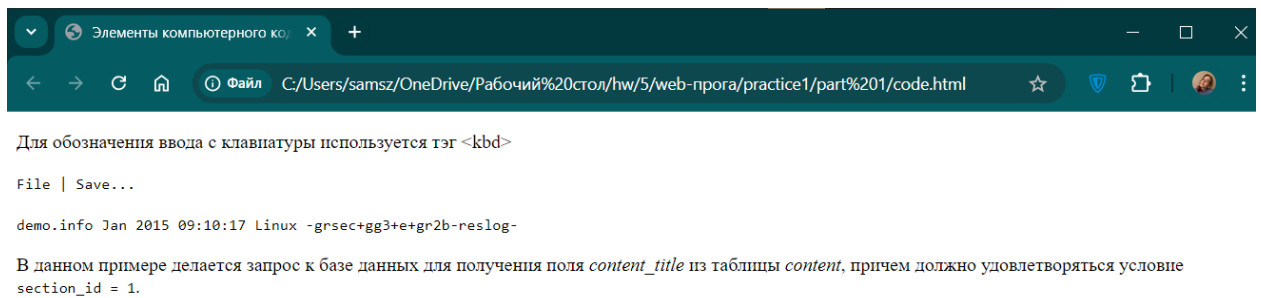


Рисунок 35 – Отображение элементов форматирования кода в браузере

Упражнение 5. Маркированные списки

В данном упражнении было необходимо научиться работать с маркированными списками. Создадим файл и поместим в тело документа маркированный список с помощью тэгов `` и помещаемых внутрь ``: тэг `` устанавливает маркированный список, а каждый пункт списка должен начинаться с элемента `` (см. рисунок 36).

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>Классификация пуговиц</title>
5   </head>
6   <body>
7     <p>Разновидности пуговиц</p>
8     <ul>
9       <li>Пуговка со сквозными отверстиями;</li>
10      <li>Пуговка с ушком;</li>
11      <li>Джинсовая пуговка;</li>
12      <li>Канадка двухщелевая.</li>
13    </ul>
14  </body>
15 </html>
```

Рисунок 36 – Создание маркированного списка

Посмотрим, как этот код отображается в браузере (см. рисунок 37).

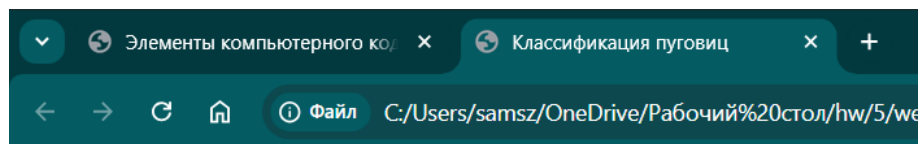


Рисунок 37 – Отображение маркированного списка в браузере

Для замены маркеров на квадратные к тэгу `` добавим атрибут `type` (см. рисунок 38) и посмотрим, как теперь отображается список в браузере (см. рисунок 39).

```
<ul type="square">
```

Рисунок 38 – Использование атрибута `type` для изменения вида маркеров

Разновидности пуговиц

- Пуговица со сквозными отверстиями;
- Пуговица с ушком;
- Джинсовая пуговица;
- Канадка двухщелевая.

Рисунок 39 – Отображение списка в браузере

Для изменения вида маркеров средствами CSS заменим созданный ранее код кодом CSS, применив `style` (см. рисунок 40), и снова проверим отображение в браузере (см. рисунок 41).

```
<ul style="list-style-type: circle;">
```

Рисунок 40 – Изменение вида маркеров с помощью CSS

Разновидности пуговиц

- Пуговица со сквозными отверстиями;
- Пуговица с ушком;
- Джинсовая пуговица;
- Канадка двухщелевая.

Рисунок 41 – Отображение списка в браузере

Далее научимся изменять отдельные элементы списка. Для этого используем `style`, добавив его к конкретному элементу списка (см. рисунок 42). Посмотрим отображение в браузере: цвет элемента списка изменился на красный, так как мы изменили свойство `color` (см. рисунок 43).

```
<p>Разновидности пуговиц</p>
<ul style="list-style-type: circle;">
  <li>Пуговица со сквозными отверстиями;</li>
  <li>Пуговица с ушком;</li>
  <li style="color: red;">Джинсовая пуговица;</li>
  <li>Канадка двухщелевая.</li>
</ul>
```

Рисунок 42 – Применение `style` к тэгу ``

Разновидности пуговиц

- Пуговица со сквозными отверстиями;
- Пуговица с ушком;
- Джинсовая пуговица;
- Канадка двухщелевая.

Рисунок 43 – Отображение изменения отдельного элемента списка

Упражнение 6. Нумерованные списки

В данном упражнении рассмотрим нумерованные списки. Для их создания используется тэг `` (см. рисунок 44).

```
<p>По форме пуговицы могут быть:</p>
<ol>
  <li>Квадратные;</li>
  <li>Треугольные;</li>
  <li>Цилиндрические;</li>
  <li>Шарообразные;</li>
  <li>Другие.</li>
</ol>
```

Рисунок 44 – Создание нумерованного списка

Проверим отображение созданного списка в браузере. Заметим, что перед каждым пунктом вместо маркера стоит номер (см. рисунок 45).

По форме пуговицы могут быть:

1. Квадратные;
2. Треугольные;
3. Цилиндрические;
4. Шарообразные;
5. Другие.

Рисунок 45 – Отображение нумерованного списка в браузере

Для изменения начального номера пунктов списка добавим к тэгу `` атрибут `start` (см. рисунок 46).

```
<ol start="5">
```

Рисунок 46 – Добавление атрибута `start`

Посмотрим полученный результат в браузере и заметим, что нумерация началась с указанного номера (см. рисунок 47).

5. Квадратные;
6. Треугольные;
7. Цилиндрические;
8. Шарообразные;
9. Другие.

Рисунок 47 – Отображение списка с измененной нумерацией

Для изменения типа нумерации пунктов списка к тэгу добавим атрибут type (см. рисунок 48).

```
<ol start="5" type="i">
```

Рисунок 48 – Добавление атрибута type

Сохраним изменения и проверим полученный результат в браузере (см. рисунок 49).

- v. Квадратные;
- vi. Треугольные;
- vii. Цилиндрические;
- viii. Шарообразные;
- ix. Другие.

Рисунок 49 – Отображение списка с другим типом нумерации

Упражнение 7. Список определений

Рассмотрим создание списка определений в HTML. Список определений состоит из двух элементов – термина и его определения. Сам список задается с помощью тэга <dl>, термин с помощью тэга <dt>, а его определение – <dd>. Пример кода для его создания представлен на рисунке 50.

```
<p>Пуговицы могут выполнять следующие функции:</p>
<dl>
  <dt>Утилитарная</dt>
  <dd> - застежка на одежде, аксессуаре и т.п.;</dd>
  <dt>Декоративная</dt>
  <dd> - украшение; </dd>
  <dt>Магическая</dt>
  <dd> - выступает в качестве оберега или талисмана,
    отпугивающего враждебные силы; </dd>
  <dt>Информативная</dt>
  <dd> - опознавательный знак принадлежности
    к определенной группе, профессии, роду войск (сил) и т. д. </dd>
</dl>
```

Рисунок 50 – Создание списка определений

Посмотрим, как он отображается в браузере (см. рисунок 51).

Пуговицы могут выполнять следующие функции:

Утилитарная

- застежка на одежде, аксессуаре и т.п.;

Декоративная

- украшение;

Магическая

- выступает в качестве оберега или талисмана, отпугивающего враждебные силы;

Информативная

- опознавательный знак принадлежности к определенной группе, профессии, роду войск (сил) и т. д.

Рисунок 51 – Отображение списка определений в браузере

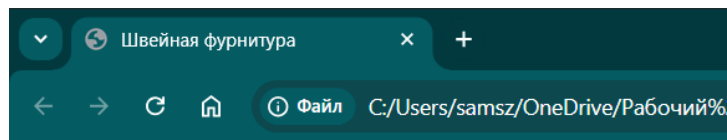
Упражнение 8. Вложенные списки

В данном упражнении рассмотрим вложенные списки. Создадим вложенный список, как показано на рисунке 52.

```
1  <!DOCTYPE html>
2  <html>
3    <head>
4      <title>Швейная фурнитура</title>
5    </head>
6    <body>
7      <p>Швейная фурнитура</p>
8      <ol>
9        <li>Аксессуары для одежды и обуви:
10          <ol type="a">
11            <li>Броши;</li>
12            <li>Булавки;</li>
13            <li>Другие принадлежности:
14              <ul type="square">
15                <li>Принадлежности для глажки;</li>
16                <li>Принадлежности для стирки;</li>
17                <li>Принадлежности для ухода за одеждой;</li>
18              </ul>
19            </li>
20            <li>Другие принадлежности;</li>
21          </ol>
22        </li>
23        <li>Нитки;</li>
24        <li>Молнии:
25          <ol type="i">
26            <li>Витые;</li>
27            <li>Металлические;</li>
28          </ol>
29        </li>
30        <li>Пуговицы.</li>
31      </ol>
32    </body>
33  </html>
```

Рисунок 52 – Создание вложенного списка

Проверим корректность его отображения и логику в браузере (см. рисунок 53).



Швейная фурнитура

1. Аксессуары для одежды и обуви:
 - a. Брошки;
 - b. Булавки;
 - c. Другие принадлежности:
 - Принадлежности для глажки;
 - Принадлежности для стирки;
 - Принадлежности для ухода за одеждой;
 - d. Другие принадлежности;
2. Нитки;
3. Молнии:
 - i. Витые;
 - ii. Металлические;
4. Пуговицы.

Рисунок 53 – Отображение вложенного списка в браузере

3. Гиперссылки

Упражнение 1. Основы работы с гиперссылками

Изучим основы гиперссылок. Для этого создадим две гиперссылки: одну со страницы index.html на страницу classification.html, вторую – ведущую обратно с classification.html на index.html. Также создадим на странице index.html ссылку на статью в Википедии о пуговицах, затем – на страницу Яндекса и добавим способ открытия этого ресурса – в новой вкладке. Добавить ссылки можно с помощью тэга `<a>` и атрибута `href`, а указать способ открытия целевого ресурса с помощью атрибута `target` (см. рисунок 54).

```
<a href = "classification.html">Переход к странице классификации</a><br />
<a href = "https://ru.wikipedia.org/wiki/%D0%9F%D1%83%D0%B3%D0%BE%D0%B2%D0%B8%D1%86%D0%B0">Статья в Википедии</a><br />

<p>Знаете ли вы, что Первая Мировая война случилась из-за плохо расстегивающихся пуговиц?
<blockquote cite="http://glamik.ru/blog/2012-03-07-640">Эрцгерцог Франц-Фердинанд
умер после выстрела только из-за того, что на расстегивание всех его пуговиц для обработки раны
потребовалось слишком много времени.</blockquote></p>

<a href = "https://ya.ru/?npr=1&utm_referrer=https%3A%2F%2Fyandex.ru%2F" target = "_blank" >Яндекс</a>
```

Рисунок 54 – Код с созданием гиперссылок

Откроем наш файл в браузере. Заметим, что при наведении на ссылку снизу отображается адрес, куда она ведет (см. рисунок 55).

[Переход к странице классификации](#)
[Статья в Википедии](#)

Знаете ли вы, что Первая Мировая война случилась из-

Эрцгерцог Франц-Фердинанд умер после выстрела, потребовавшегося слишком много времени.

[Яндекс](#)

Наша контактная информация:

Россия,
Санкт-Петербург,
ул. Ленина, д. 1

<https://ru.wikipedia.org/wiki/Пуговица>

Рисунок 55 – Отображение страницы с гиперссылками

Также обратим внимание, что страница Яндекса действительно открывается в новой вкладке (см. рисунок 56). Это произошло из-за добавления атрибута target.

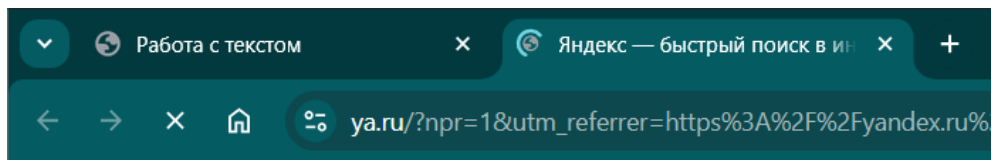


Рисунок 56 – Открытие ссылки в новой вкладке

Упражнение 2. Работа с цветом гиперссылок

Рассмотрим, как изменять цвета гиперссылок. Как правило, для этого используют CSS, но можно встретить вариант с HTML. Для того, чтобы сделать это с помощью HTML, необходимо задать цвета ссылок в качестве атрибутов тэга `<body>`, где `<alink>` – это активная ссылка, `<vlink>` – посещенная (см. рисунок 57).

```
<body alink="Fuchsia" vlink="Aqua">
```

Рисунок 57 – Изменение цвета ссылок

Код, приведенный выше, окрашивает активные ссылки в цвет фуксия (см. рисунок 58), а посещенные – в голубой (см. рисунок 59).

This is a link:

[Yandex](#)

Рисунок 58 – Отображение активной ссылки

This is a link:

[Yandex](http://www.yandex.ru)

Рисунок 59 – Отображение посещенной ссылки

Более актуальным является вариант с использованием CSS. Для его применения удалим атрибуты, добавленные ранее в тэг `<body>`, а в раздел `<head>` добавим тэг `<style>`, внутри которого укажем необходимые параметры цвета для ссылок при наведении, а также для посещенных и активных ссылок (см. рисунок 60).

```
1 <html>
2   <head>
3     <title>Links</title>
4     <style>
5       a:visited{color:blue;}
6       a:active{color:Fuchsia;}
7       a:hover {color:yellow;}
8     </style>
9   </head>
10  <body>
11    <p>This is a link:</p>
12    <a href = "http://www.yandex.ru" target = "_blank">Yandex</a>
13  </body>
14 </html>
```

Рисунок 60 – Применение CSS для изменения цвета ссылок

Посмотрим результат в браузере и убедимся в изменении цветовых решений. При наведении на ссылку ее цвет должен становиться желтым (см. рисунок 61), в момент нажатия – розовым (см. рисунок 62), а после посещения – голубым (см. рисунок 63).

This is a link:

[Yandex](http://www.yandex.ru)

Рисунок 61 – Отображение цвета ссылки при наведении

This is a link:

[Yandex](http://www.yandex.ru)

Рисунок 62 – Отображение цвета ссылки во время нажатия

This is a link:

[Yandex](#)

Рисунок 63 – Отображение цвета ссылки после посещения

Упражнение 3. Ссылка на адрес электронной почты

Организуем ссылку на адрес электронной почты. Для этого добавим тэг <a>, а в атрибуте href перед ссылкой укажем префикс mailto (см. рисунок 64).

```
<a href="mailto:myAdress@domain.ru">Отправьте мне письмо</a>
```

Рисунок 64 – Создание ссылки на адрес электронной почты

Посмотрим, как ссылка отобразилась в браузере (см. рисунок 65) и убедимся, что при нажатии на гиперссылку вызывается почтовый клиент и открывается окно нового сообщения с заполненным полем получателя (см. рисунок 66).

[Отправьте мне письмо](mailto:myAdress@domain.ru)

Рисунок 65 – Отображение ссылки на адрес электронной почты в браузере

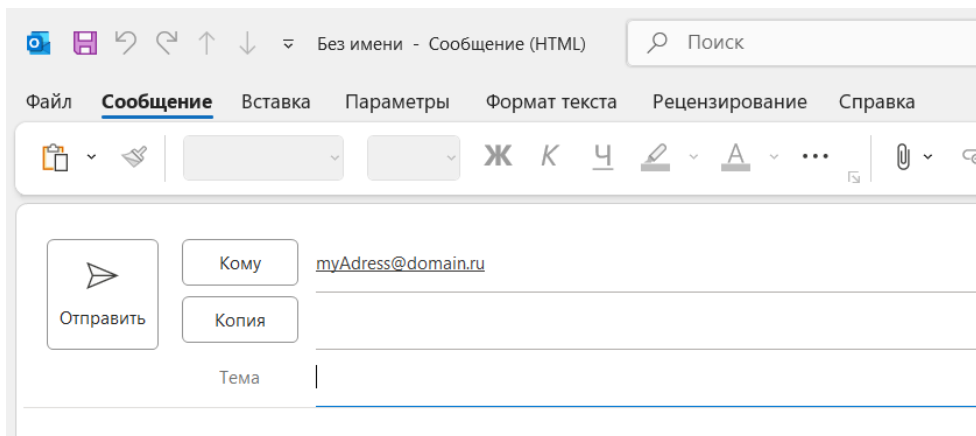


Рисунок 66 – Открытие почтового клиента при нажатии на ссылку

Упражнение 4. Внутренние ссылки документа

Для организации навигации в рамках одного документа можно создать один или несколько якорей или внутренних ссылок документа. Рассмотрим их использование. Для этого заголовкам текста добавим id и затем в верхней части документа добавим простейшее меню для навигации по тексту страницы, состоящее из двух ссылок (см. рисунок 67).

```

<a href = "#history_1">История первая</a><br />
<a href = "#history_2">История вторая</a><br />

<div>
<h2 id="history_1">История первая</h2>

```

Рисунок 67 – Создание внутренних ссылок

Посмотрим результат в браузере (см. рисунок 68) и проверим работоспособность ссылок, нажав на них и убедившись, что при нажатии браузер переходит к выбранной истории (см. рисунок 69).

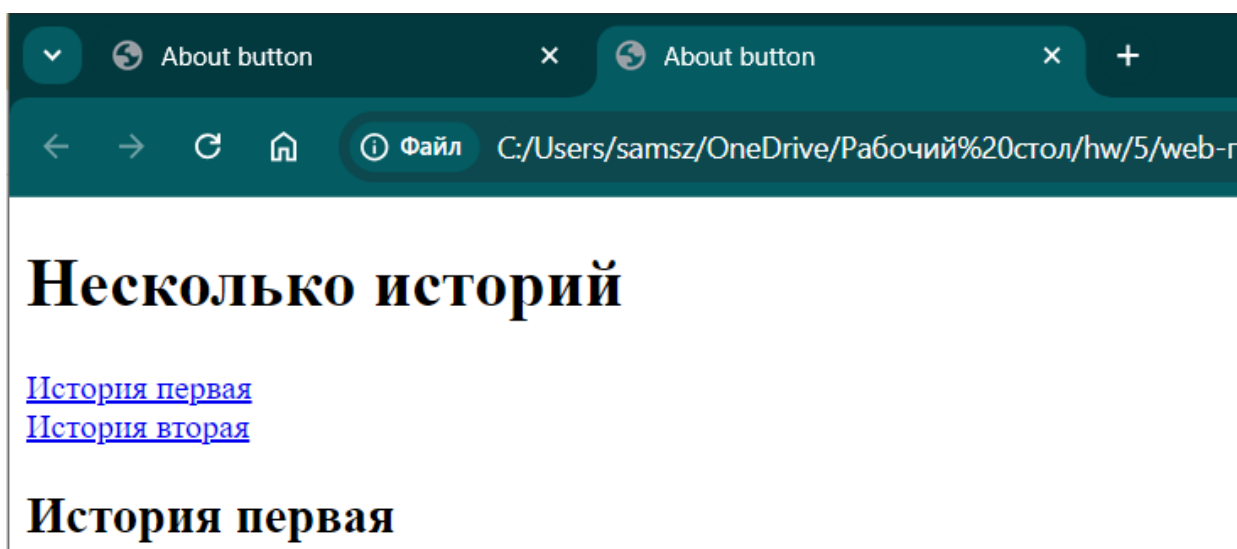


Рисунок 68 – Отображение внутренних ссылок в браузере

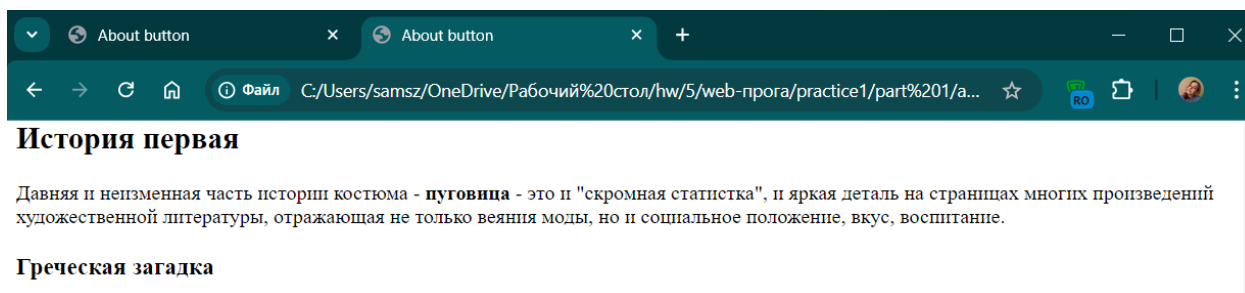


Рисунок 69 – Результат нажатия на ссылку «История первая»

4. Таблицы

Упражнение 1. Создание таблицы

Рассмотрим создание таблиц в HTML. Для этого используем тэги <table>, <tr> и <td>. Элемент <table> служит контейнером для элементов, определяющих содержимое таблицы, а тэги <tr> и <td> задают строки и ячейки таблицы (см. рисунок 70).

```

<table>
  <tr>
    <td>Фамилия</td>
    <td>Имя</td>
    <td>Отчество</td>
    <td>Город</td>
  </tr>
  <tr>
    <td>Пупкин</td>
    <td>Василий</td>
    <td>Петрович</td>
    <td>Москва</td>
  </tr>
  <tr>
    <td>Пупкин</td>
    <td>Феофан</td>
    <td>Васильевич</td>
    <td>Москва</td>
  </tr>
  <tr>
    <td>Шашкин</td>
    <td>Сидор</td>
    <td>Богданович</td>
    <td>Санкт-Петербург</td>
  </tr>
  <tr>
    <td>Лютикова</td>
    <td>Аделина</td>
    <td>Федоровна</td>
    <td>Тверь</td>
  </tr>
</table>

```

Рисунок 70 – Создание таблицы

Посмотрим на ее отображение в браузере (см. рисунок 71).

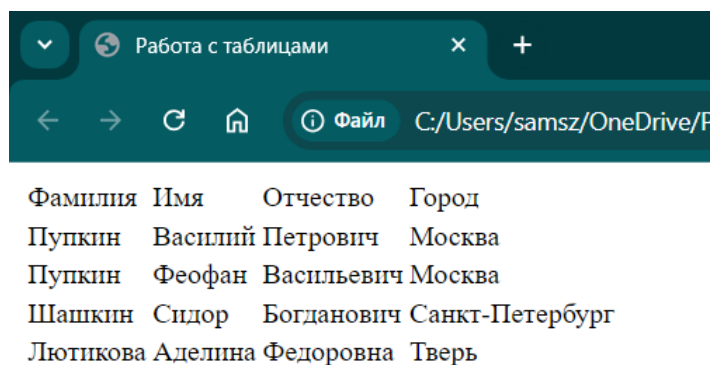


Рисунок 71 – Отображение таблицы в браузере

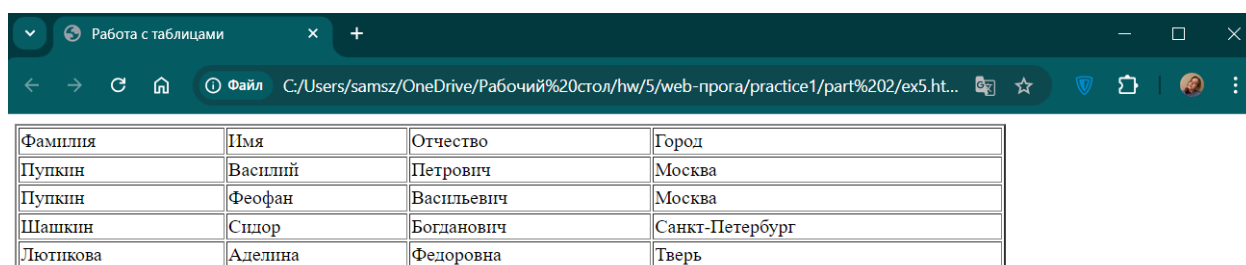
Упражнение 2. Работа с границами таблицы

Для создания границ таблицы можно использовать CSS или средства HTML. К тэгу `<table>` применим атрибут `border` со значением «2», а также добавим стиль для изменения ширины таблицы, чтобы таблица составляла

80% от ширины окна (см. рисунок 72). Посмотрим результат отображения в браузере (см. рисунок 73).

```
<body>
  <table border="2" style="width: 80%">
    <tr>
```

Рисунок 72 – Изменение внешнего вида таблицы



The screenshot shows a web browser window with the title 'Работа с таблицами'. The address bar shows a local file path. The table displayed has four columns: 'Фамилия', 'Имя', 'Отчество', and 'Город'. It contains five rows of data.

Фамилия	Имя	Отчество	Город
Пупкин	Василий	Петрович	Москва
Пупкин	Феофан	Васильевич	Москва
Шашкин	Сидор	Богданович	Санкт-Петербург
Лютикова	Аделина	Федоровна	Тверь

Рисунок 73 – Отображение таблицы в браузере

Упражнение 3. Создание заголовка и подписи таблицы

Рассмотрим создание заголовка и «шапки» таблицы. Изменим тэги первой строки таблицы на тэг `<th>` (см. рисунок 74). Он служит для создания заголовочных ячеек в таблице.

```
<tr>
  <th>Фамилия</th>
  <th>Имя</th>
  <th>Отчество</th>
  <th>Город</th>
</tr>
```

Рисунок 74 – Создание «шапки» таблицы

Для создания заголовка таблицы дополним код парным тэгом `<caption>` (см. рисунок 75).

```
<table border="2" style="width: 100%">
  <caption>Список сотрудников</caption>
  <tr>
```

Рисунок 75 – Добавление заголовка таблицы

Посмотрим отображение изменений в браузере. У таблицы появилась «шапка», а также заголовок (см. рисунок 76).

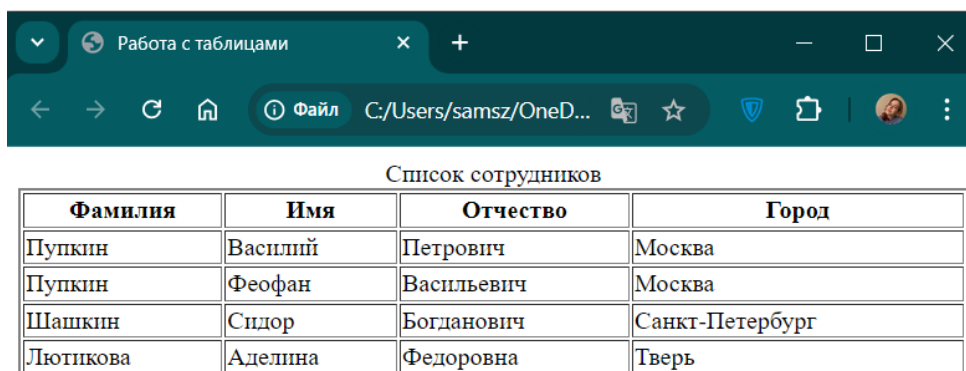


Рисунок 76 – Отображение «шапки» и заголовка таблицы

Для изменения расположения заголовка можно использовать устаревший вариант с HTML-кодом или современный подход с использованием CSS. Для перемещения заголовка средствами HTML дополним тэг <caption> атрибутом align (см. рисунок 77) и посмотрим результат в браузере (см. рисунок 78).

```
<table border="2" style="width: 100%">
  <caption align="bottom">Список сотрудников</caption>
```

Рисунок 77 – Изменение расположения заголовка таблицы средствами HTML

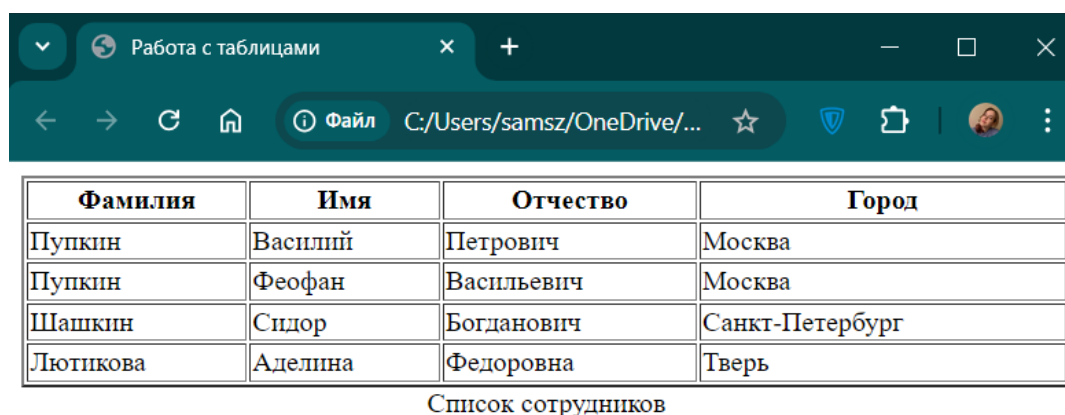


Рисунок 78 – Отображение изменения расположения заголовка таблицы

Для перемещения заголовка средствами CSS заменим атрибут align включением стиля CSS (см. рисунок 79).

```
<table border="2" style="width: 100%">
  <caption style="caption-side: bottom;">Список сотрудников</caption>
```

Рисунок 79 - Изменение расположения заголовка таблицы средствами CSS.

Убедимся, что результат отображения в браузере идентичен первому варианту (см. рисунок 80).

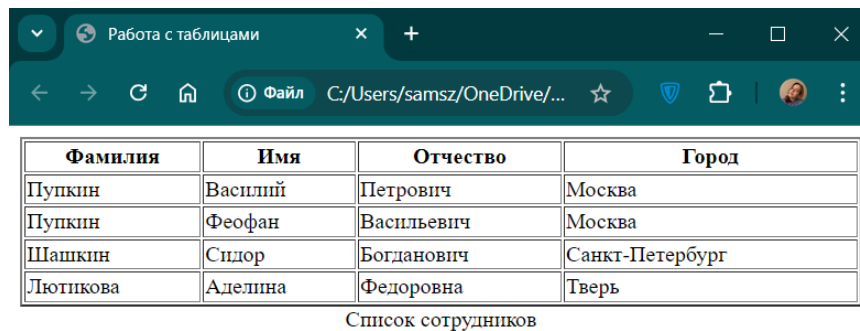


Рисунок 80 – Отображение изменения расположения заголовка таблицы

Упражнение 4. Объединение ячеек

Рассмотрим возможности объединения ячеек. Для объединения ячеек по горизонтали применяется атрибут `colspan`: он устанавливает число ячеек, которые должны быть объединены по горизонтали. Изменим код последней строки так, как показано на рисунке 81, и проверим корректность отображения в браузере (см. рисунок 82).

```
<tr>
  <td colspan = "3">Лютикова Аделина Федоровна</td>
  <td>Тверь</td>
</tr>
```

Рисунок 81 – Объединение ячеек таблицы по горизонтали

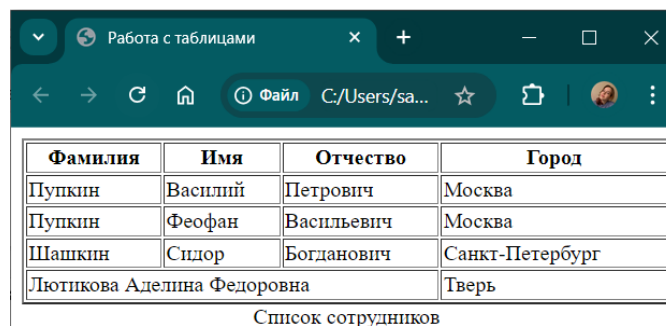


Рисунок 82 – Отображение объединения ячеек по горизонтали

Для объединения ячеек по вертикали применяется `rowspan`. Внесем изменения в код (см. рисунок 83) и посмотрим изменения в браузере (см. рисунок 84).

```
<tr>
  <td rowspan="2">Пупкин</td>
  <td>Василий</td>
  <td>Петрович</td>
  <td rowspan="2">Москва</td>
</tr>
<tr>
  <td>Феофан</td>
  <td>Васильевич</td>
</tr>
```

Рисунок 83 – Объединение ячеек таблицы по вертикали

Фамилия	Имя	Отчество	Город
Пупкин	Василий	Петрович	Москва
	Феофан	Васильевич	
Шашкин	Сидор	Богданович	Санкт-Петербург
Лютикова Аделина Федоровна			Тверь

Список сотрудников

Рисунок 84 – Отображение объединения ячеек по вертикали

Упражнение 5. Структурные блоки таблиц

Рассмотрим структурные блоки таблиц. Для структурирования таблиц используются тэги `<thead>`, `<tbody>` и `<tfoot>`. `<thead>` определяет заголовок таблицы, `<tbody>` содержит основное содержимое таблицы, а `<tfoot>` – подвал таблицы (например, итоги). Создадим таблицу с применением этих тэгов, а также с добавлением стилей CSS (см. рисунок 85).

```

<table>
  <thead style="color: green; background-color: lavender;">
    <tr>
      <th>Месяц</th>
      <th>Продажи</th>
      <th>Менеджер</th>
    </tr>
  </thead>
  <tfoot style="color: blue;">
    <tr>
      <td>Итого</td>
      <td colspan="2" style="text-align: right;">170 000</td>
    </tr>
  </tfoot>
  <tbody style="color: red;">
    <tr>
      <td>Ноябрь</td>
      <td>70 000</td>
      <td>Шашкин</td>
    </tr>
    <tr>
      <td>Декабрь</td>
      <td>100 000</td>
      <td>Пупкин</td>
    </tr>
  </tbody>
</table>

```

Рисунок 85 – Создание структурированной таблицы

Отобразим результат в браузере и обратим внимание на изменившийся стиль и порядок расположения элементов (см. рисунок 86).

Месяц	Продажи	Менеджер
Ноябрь	70 000	Шашкин
Декабрь	100 000	Пупкин
Итого	170 000	

Рисунок 86 – Обновленный стиль таблицы

Добавим к таблице тэг `colgroup` (см. рисунок 87). Он предназначен для задания ширины и стиля одной или нескольких полей таблицы. Посмотрим получившийся результат в браузере (см. рисунок 88).

```
<colgroup span = "3" style = "background-color:silver; width: 100%;">
```

Рисунок 87 – Добавление тэга `colgroup`

Месяц	Продажи	Менеджер
Ноябрь	70 000	Шашкин
Декабрь	100 000	Пупкин
Итого		170 000

Рисунок 88 – Отображение изменений

Добавим тэги `<col>` (см. рисунок 89). Он задает ширину и другие характеристики одной или нескольких колонок таблицы.

```
<!--<colgroup span = "3" style = "background-color:silver">-->
<colgroup>
  <col style="background-color: silver">
  <col span="2" style="background-color:yellow">
</colgroup>
```

Рисунок 89 – Добавление `<col>`

Посмотрим на произошедшие с таблицей изменения (см. рисунок 90).

Месяц	Продажи	Менеджер
Ноябрь	70 000	Шашкин
Декабрь	100 000	Пупкин
Итого		170 000

Рисунок 90 – Изменение стиля таблицы

5. Изображения и медиаконтент

Упражнение 1. Основы работы с изображениями

Для отображения на веб-странице значимых для контента изображений используется тег ``, а адрес файла с изображением задаётся с помощью атрибута `src`. Добавим изображение на страницу, с помощью атрибута `alt` добавим альтернативный текст, а с помощью `title` зададим подсказку к изображению (см. рисунок 91).

```
<img src= "images/button_1.jpg" alt="Button" title="This is a button">
```

Рисунок 91 – Добавление изображения на страницу

Проверим отображение результата в браузере (см. рисунок 92).

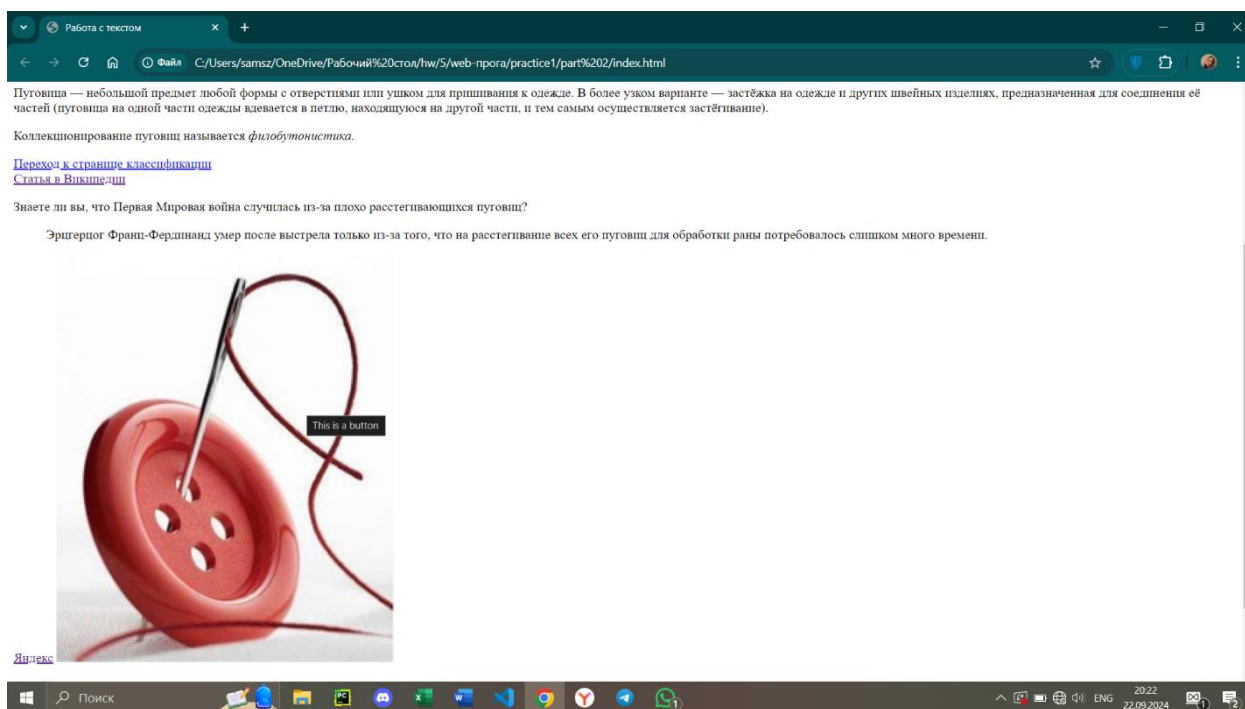


Рисунок 92 – Отображение изображения в браузере

Упражнение 2. Изменение размеров изображения

Для изменения размеров изображения можно использовать возможности HTML или CSS. Чтобы изменить размер изображения средствами HTML, добавим атрибуты `width` и `height` (см. рисунок 93).

```
<img src= "images/button_1.jpg" alt="Button" title="This is a button" width="200" height="230">
```

Рисунок 93 – Изменение размера изображения средствами HTML

Проверим, что размер изображения в браузере изменился (см. рисунок 94).



Наша контактная информация:

Россия,
Санкт-Петербург,
ул.Ленина, д.1
(812)123-4567

Рисунок 94 – Отображение изображения с измененными размерами
Изменим размеры изображения с помощью CSS (см. рисунок 95).

```
<img src= "images/button_1.jpg" alt="Button" title="This is a button" style="width: 100px; height: 130px">
```

Рисунок 95 – Изменение размера изображения средствами CSS

Проверим, что отображение в браузере также корректно (см. рисунок 96).



Наша контактная информация:

Россия,
Санкт-Петербург,
ул.Ленина, д.1
(812)123-4567

Рисунок 96 – Отображение изображения с измененными размерами

Упражнение 3. Создание изображения-гиперссылки

Добавим на страницу изображение-гиперссылку. Для этого вложим изображение в тэг <a> (см. рисунок 97).

```
<a href = "classification.html">  
  <img src= "images/button_1.jpg" alt="classification" title = "Classification Page"  
  style="width:100px; height:130px">  
</a><br />
```

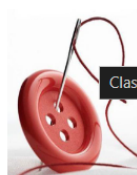
Рисунок 97 – Добавление изображения-гиперссылки

Теперь проверим, что при открытии страницы в браузере картинка отображается, при наведении на нее высвечивается текст title, а также отображается ссылка, на которую ведет изображение (см. рисунок 98).

Интересное о пуговицах

Пуговка — небольшой предмет любой формы с отверстиями или ушком для пришивания к одежде. В более узком варианте — застёжка на одежде и других швейных изделиях, предназначенная для соединения её частей (пуговка на одной части одежды вдевается в петлю, находящуюся на другой части, и тем самым осуществляется застёгивание).

Коллекционирование пуговиц называется *филобутонистика*.



[Статья в Википедии](#)

file:///C:/Users/samsz/OneDrive/Рабочий стол/hw/5/web-npora/practice1/part 2/classification.html

Рисунок 98 – Просмотр изображения-гиперссылки в браузере

Упражнение 4. Карты изображений

Карта изображений позволяет привязывать ссылки к разным областям одного изображения. Для ее создания добавим изображение и тэг <map> с атрибутом name, а к добавим атрибут usemap. Внутри <map> добавим пока пустой тэг <area>. Затем на сайте Online Image Map Editor получим координаты областей ссылок (см. рисунки 99–100).

Online Image Map Editor

This is an online editor to create so-called HTML Image maps. The underlying code is also [available on Github](#) to use it in your open source projects or to license it for your commercial use cases.

1. Select or upload an image

From your computer:

OR from the Internet:

OR use a sample image:

Uploaded button_4.jpg - [upload another](#)

☒ [accept](#)

☒ [accept](#)

☒ [accept](#)

2. Draw image map areas

Zoom: 100%

Standard imagemap

Label with numbers

rectangle

coords

href

alt

Target: <not set>

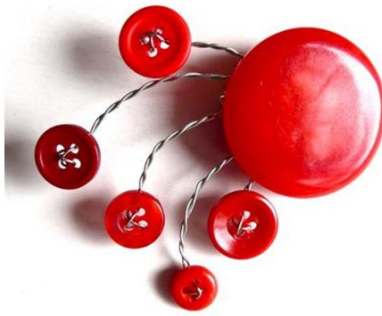


Рисунок 99 – Добавление картинки на сайт Online Image Map Editor

1. Select or upload an image

From your computer:

OR from the Internet:

OR use a sample image:

Uploaded button_4.jpg - [upload another](#)

[accept](#)

[accept](#)

[accept](#)

2. Draw image map areas

Zoom: 100% Standard imagemap Label with numbers

circle 404,199, https://ya.ru/ yandex.ru Target: new wi

Ready

3. Get the code

This is the generated image map HTML code. Click into the textarea below and press Ctrl+C to copy the code to your clipboard. Please note, that you have to attach this code to your image, via the usemap property ([read more](#)).

```
<map id="imgmap202492414619" name="imgmap202492414619"><area shape="circle" alt="yandex.ru" title="" coords="404,199,114" href="https://ya.ru/" target="_blank" /></map>
```

-- Created by Online Image Map Editor (<http://www.maschek.hu/imagemap/index>) --></map>

Рисунок 100 – Выбор области и получение координат

Далее вернемся к редактированию нашего кода. Сгенерированный код добавим в тэг <area>, учитывая, что тэги <map> и <area> у нас уже созданы. Также удалим комментарии сайта (см. рисунок 101).

```
1 <!DOCTYPE html>
2 <html>
3     <head>
4         <title>Картинки</title>
5     </head>
6     <body>
7         <img src = "images/button_4.jpg" usemap="#MyMap">
8         <map name="MyMap">
9             <area shape="circle" alt="yandex.ru" title="" coords="404,199,114"
10             href="https://ya.ru/" target="_blank" />
11         </map>
12     </body>
13 </html>
```

Рисунок 101 – Код для создания карты изображения

Проверим отображение и работоспособность карты, открыв файл в браузере. Заметим, что при наведении на большой красный круг на

изображении появляется ссылка на сторонний сайт (см. рисунок 102), а при нажатии на него – происходит переход по ссылке (см. рисунок 103).

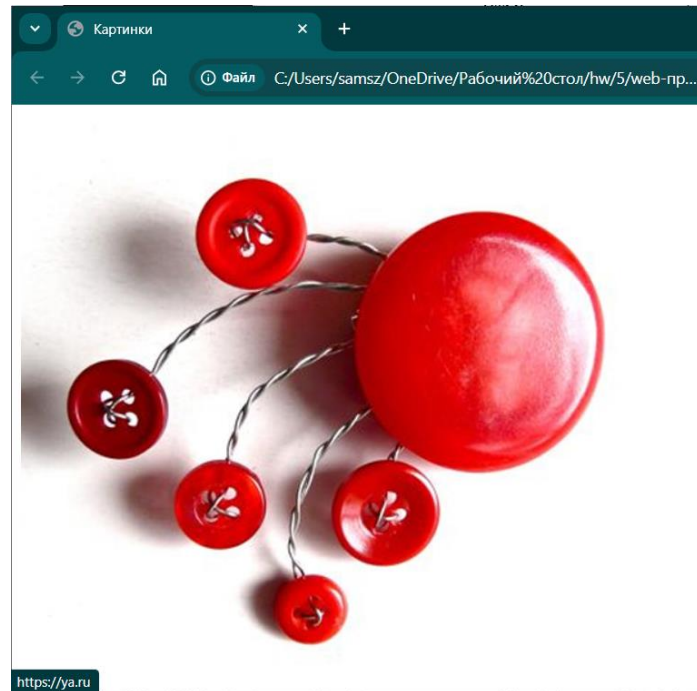


Рисунок 102 – Отображение изображения в браузере

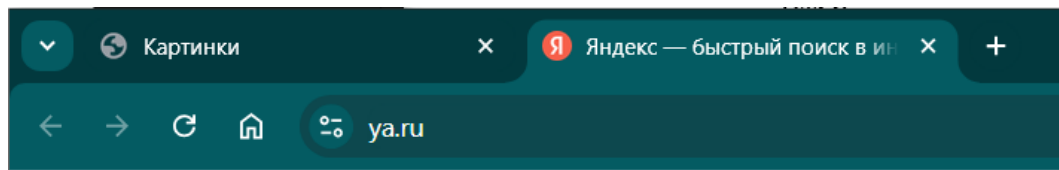


Рисунок 103 – Переход на сторонний сайт при нажатии на изображение

Упражнение 5. Добавление медиаконтента

Для добавления видеоконтента на страницу используем тэг `<video>` (см. рисунок 104), а затем проверим его отображение в браузере (см. рисунок 105).

```
1  <!DOCTYPE html>
2  <html>
3    <head>
4      <title>Media</title>
5    </head>
6    <body>
7      <video width="320" height="240" controls>
8        <source src="video.mp4" type="video/mp4">
9        <p>Your browser does not support the video tag.</p>
10     </video>
11   </body>
12 </html>
```

Рисунок 104 – Добавление видео на страницу

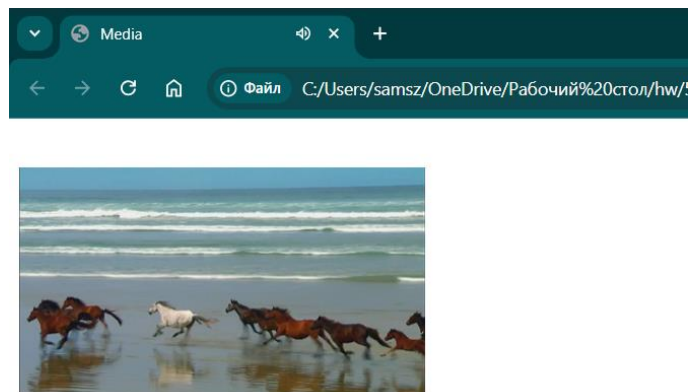


Рисунок 105 – Отображение видео на странице в браузере

6. Формы

Упражнение 1. Текстовые поля формы

Изучим текстовые поля формы. Для добавления формы на страницу используется тэг `<form>`. Атрибут `method` определяет, каким способом данные формы будут отправлены на сервер, а `enctype` определяет, как данные будут кодироваться при отправке формы (см. рисунок 106). С помощью тэга `input` с типом “text” создадим простое текстовое поле для ввода текста и поле для ввода текста с максимальной длиной 5 символов, с типом “submit” – кнопку для отправки данных, а с помощью `<textarea>` – многострочное текстовое поле (см. рисунок 107).

```
<body>
  <p>Основные элементы форм:</p>
  <form action = "ex7.html" method = "POST" enctype = "multipart/form-data">
    Простое текстовое поле: <input type = "text" name = "txtField" /><br/>
    Поле длиной максимум 5 символов: <input type = "text" name = "txtField" maxlength = "5" /><br/>
    Многострочное текстовое поле:<br />
    <textarea name = "txtArea">This is a text area</textarea><br/>
    <input type = "submit" name = "submit" value = "Отправить!" />
  </form>
</body>
```

Рисунок 106 – Создание текстовых полей и кнопки

Рисунок 107 – Отображение текстовых полей и кнопки

Далее добавим в форму поле ввода пароля (см. рисунки 108–109), а также скрытое поле (см. рисунки 110–111). Оно не отображается на странице и прячет своё содержимое от пользователя.

```
Поле для ввода пароля:<input type = "password" name = "psw"/><br/>
```

Рисунок 108 – Добавление поля ввода для пароля

Поле для ввода пароля:

Рисунок 109 – Отображение поля ввода для пароля

```
Скрытое поле: <input type = "hidden" name = "hdnTxt"/><br/>
```

Рисунок 110 – Добавление скрытого поля ввода

Скрытое поле:

Рисунок 111 – Отображение скрытого поля

Упражнение 2. Типы полей HTML5

Рассмотрим новые типы полей. Они создаются с помощью изменения типа в атрибуте type (см. рисунок 112). Значение color создает форму с выбором цвета, date с выбором даты, email – форму для ввода электронной почты с проверкой соответствия введенного значения формату, number – для ввода числового значения, week – для выбора определенной недели в году, url – для ввода веб-адреса с проверкой формата введенного значения (см. рисунок 113).

```
<form>
  Работа с цветом:<input type = "color" name = "color"/><br/>
  Работа с датой:<input type = "date" name = "date"/><br/>
  Проверка e-mail:<input type = "email" name = "mail"/><br/>
  Проверка чисел:<input type = "number" name = "num"/><br/>
  Работа с днем недели:<input type = "week" name = "week"/><br/>
  Проверка URL:<input type = "url" name = "url"/><br/>
  <input type = "submit" name = "submit" value = "Отправить!"/>
</form>
```

Рисунок 112 – Создание полей новых типов

Основные элементы форм:

Простое текстовое поле:

Поле длиной максимум 5 символов:

Многострочное текстовое поле:

Поле для ввода пароля:

Скрытое поле:

Работа с цветом:

Работа с датой:

Проверка e-mail:

Проверка чисел:

Работа с днем недели:

Проверка URL:

Отправить!

Рисунок 113 – Просмотр отображения полей новых типов в браузере

Упражнение 3. Кнопки

Кнопки в HTML создаются с помощью элемента `<input>`. В зависимости от значения атрибута `type` можно создавать кнопки с разным функционалом.

Создадим новую форму с текстовым полем, кнопками `reset` для очистки формы и `submit` для отправки, а также простой кнопкой. Далее создадим кнопку с изображением на ней (см. рисунок 114). Посмотрим, как отображаются созданные элементы формы в браузере (см рисунок 115).

```
<form>
  Введите данные:<input type = "text" name = "txt"><br/>
  <input type = "reset" name = "reset" value = "Очистить поле!" /><br/>
  <input type = "submit" name = "submit" value = "Отправить!" /><br/>
  <input type = "button" name = "MyButton" value = "Просто кнопка" /><br />
  <button name = "myButton">
    <img src = "images/enot.jpg" title = "This is a button too">
  </button>
</form>
```

Рисунок 114 – Создание кнопок

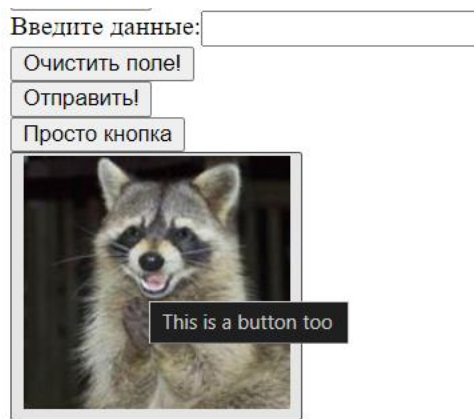


Рисунок 115 – Отображение кнопок в браузере

Упражнение 4. Флажки и переключатели

Рассмотрим создание флажков и переключателей. Создадим новую форму и поместим на нее несколько переключателей, для этого необходимо использовать `<input>` с типом `radio`. Дадим им одинаковое имя (атрибут `name`), чтобы они составляли одну группу (см. рисунок 116).

```
<form>
  <p><input type = "radio" name = "MyRadio" value = "First">First Variant</p>
  <p><input type = "radio" name = "MyRadio" value = "Second">Second Variant</p>
  <p><input type = "radio" name = "MyRadio" value = "Third">Third Variant</p>
  <p><input type = "submit" name = "submit">
</form>
```

Рисунок 116 – Создание переключателей

Посмотрим отображение переключателей в браузере (см. рисунок 117). Этот тип элементов по умолчанию допускает выбор только одного значения из группы.

☐ First Variant
☒ Second Variant
☐ Third Variant

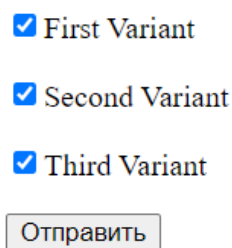
Рисунок 117 – Отображение переключателей в браузере

Аналогично создадим элементы типа `checkbox` (см. рисунок 118).

```
<form>
  <p><input type = "checkbox" name = "MyChkBx[]" value = "First">First Variant</p>
  <p><input type = "checkbox" name = "MyChkBx[]" value = "Second">Second Variant</p>
  <p><input type = "checkbox" name = "MyChkBx[]" value = "Third">Third Variant</p>
  <p><input type = "submit" name = "submit">
</form>
```

Рисунок 118 – Создание флажков

Посмотрим их отображение в браузере (см. рисунок 119). Этот тип элементов по умолчанию может допускать выбор нескольких значений.



☒ First Variant
☒ Second Variant
☒ Third Variant

Рисунок 119 – Отображение флажков в браузере

Упражнение 5. Поле со списком

Создадим новую форму и добавим в нее поле со списком – в нем можно выбирать одно или несколько значений. Для этого используем тэг `<select>` (см. рисунок 120).

```
<form>
  <select name = "cars" size = "2">
    <option value = "Ford">Ford</option>
    <option value = "Chevrolet">Chevrolet</option>
    <option value = "Vw">Volkswagen</option>
  </select>
  <input type = "submit" name = "submit"/>
</form>
```

Рисунок 120 – Создание поля со списком

Посмотрим, как отображается данный элемент в браузере (см. рисунок 121).



Ford
Chevrolet

Рисунок 121 – Отображение поля со списком в браузере

Сгруппируем пункты в выпадающем списке. Для этого добавим тэг `<optgroup>` (см. рисунок 122). Проверим, как добавленные элементы отображаются в браузере (см. рисунок 123).

```

<form>
  <select name = "cars" size = "1">
    <optgroup label = "Ford">
      <option value = "F">Focus</option>
      <option value = "M">Mondeo</option>
      <option value = "K">Kuga</option>
    </optgroup>
    <optgroup label = "Volkswagen">
      <option value = "G">Golf</option>
      <option value = "B">Beetle</option>
      <option value = "P">Passat</option>
    </optgroup>
    <optgroup label = "Chevrolet">
      <option value = "Cr">Cruze</option>
      <option value = "Cam">Camaro</option>
      <option value = "Cor">Corvette</option>
    </optgroup>
  </select>
  <input type = "submit" name = "submit"/>
</form>

```

Рисунок 122 – Создание разделов в выпадающем списке

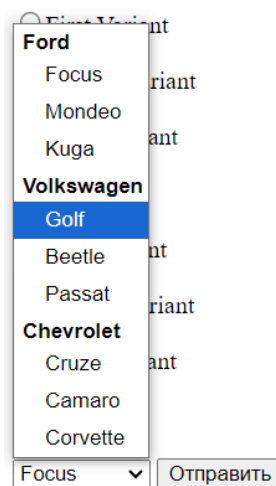


Рисунок 123 – Отображение разделов в выпадающем списке

Упражнение 6. Поле для загрузки файлов

Добавим поле для загрузки файлов. Для этого создадим `<input>` с типом `file` (см. рисунок 124).

```

<form>
  <input type = "file" name = "files" multiple>
</form>

```

Рисунок 124 – Создание поля для загрузки файлов

В браузере появляется поле «Выбрать файлы», при нажатии на которое открывается проводник с возможностью загрузить файл с компьютера (см. рисунок 125).

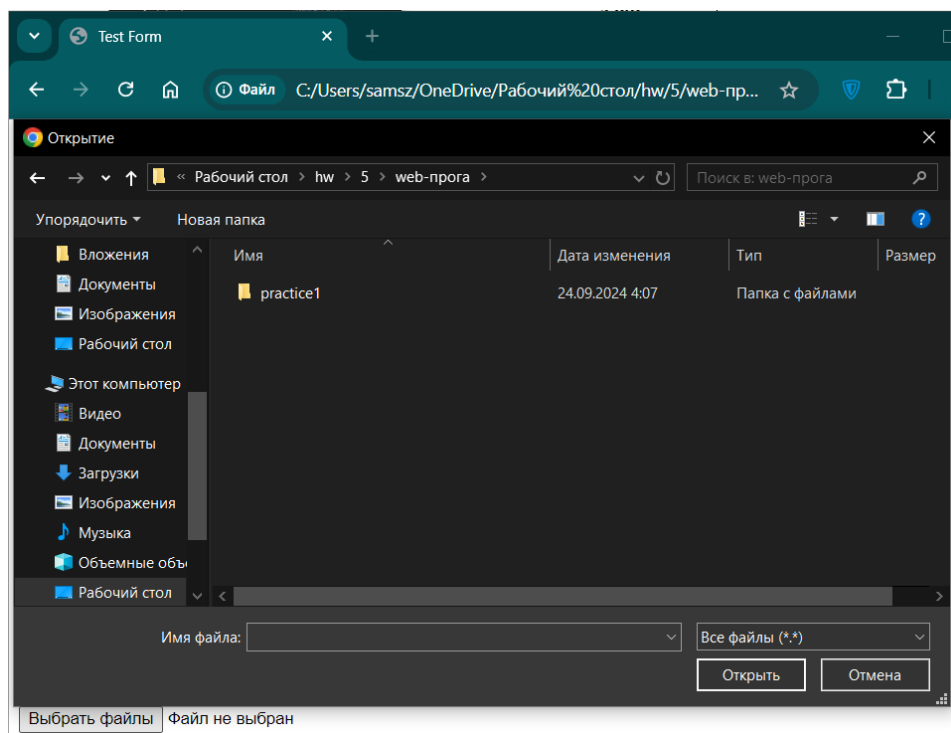


Рисунок 125 – Отображение поля для загрузки файлов

7. Основы работы с блочной структурой

Упражнение 1. Основы организации структуры документа

Рассмотрим организацию структуры документа. Добавим код, который представляет веб-страницу с заголовком, навигацией, основной информацией и нижним колонтитулом, при этом каждый из элементов лежит в своем тэге `<div>`, и можно понять, что это за структурный элемент, только с помощью `id`, которые присвоены каждому тэгу `<div>`. Подключим к странице стиль с помощью тэга `<link>` в разделе `<head>` (см. рисунок 126).

```

1  <html>
2      <head>
3          <link rel = "stylesheet" type = "text/css" href = "block_1.css">
4      </head>
5
6      <body>
7          <div id="header">
8              <h1>Монетный двор</h1>
9          </div>
10
11         <div id="nav">
12             Санкт-Петербургский монетный двор<br>
13             Московский монетный двор<br>
14             Екатеринбургский монетный двор<br>
15             Сузунский монетный двор<br>
16         </div>
17
18         <div id="section">
19             <h1>Санкт-Петербургский монетный двор</h1>
20             <p>
21                 Санкт-Петербургский монетный двор Гознака -
22                 старейшее действующее предприятие по выпуску монетно-медальной продукции в России.
23             </p>
24             <p>
25                 За почти три столетия производственной деятельности накоплен уникальный опыт,
26                 который успешно используется при изготовлении изделий.
27             </p>
28         </div>
29
30         <div id="footer">
31             Copyright © http://www.mintspb.ru/
32         </div>
33     </body>
34 </html>
35

```

Рисунок 126 – Организация структуры с помощью разделения информации на блоки div

Посмотрим подключенный файл со стилем. Знаки решетки означают, что стиль необходимо применить к конкретному элементу с указанным id (см. рисунок 127). Посмотрим, как отображается данная страница в браузере, чтобы убедиться, что стиль применился (см. рисунок 128).

```

1  #header {
2      background-color: green;
3      color: white;
4      text-align: center;
5      padding: 3px;
6  }
7  #nav {
8      line-height: 30px;
9      background-color: silver;
10     height: 300px;
11     width: 270px;
12     float: left;
13     padding: 5px;
14 }
15 #section {
16     width: 350px;
17     float: left;
18     padding: 10px;
19 }
20 #footer {
21     background-color: green;
22     color: white;
23     clear: both;
24     text-align: center;
25     padding: 3px;
26 }
27

```

Рисунок 127 – Файл со стилем CSS

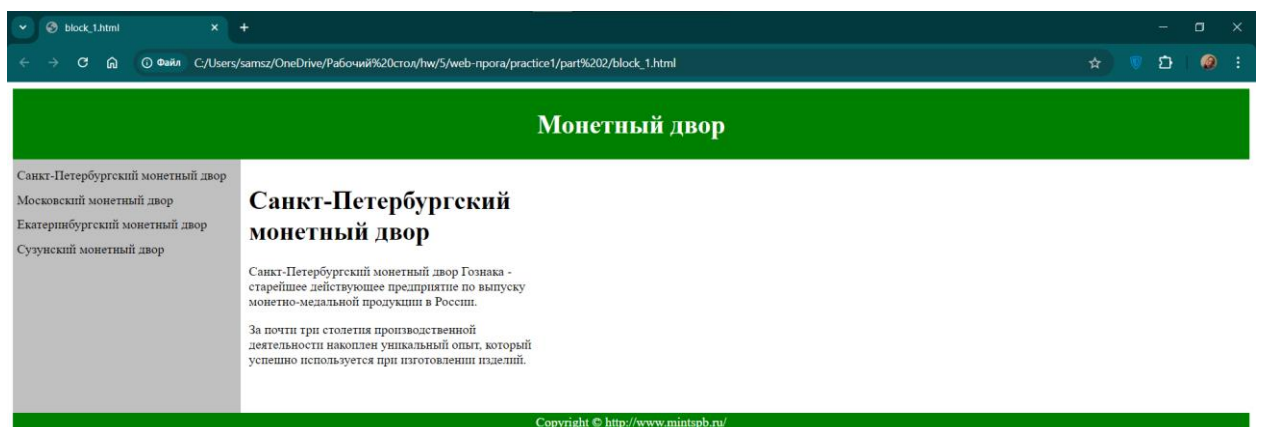


Рисунок 128 – Отображение страницы в браузере

Упражнение 2. Работа с разделами документа

В данном файле каждый структурный блок выделен с помощью специальных тэгов. Семантический элемент `<header>` выделяет заголовочную часть страницы, `<nav>` – элемент для навигации по сайту, `<section>` – секция

с основным контентом, `<footer>` – элемент, располагающийся внизу страницы. Также подключим файл со стилем (см. рисунок 129).

```
1 <html>
2   <head>
3     <link rel = "stylesheet" type = "text/css" href = "block_2.css">
4   </head>
5
6   <body>
7     <header>
8       <h1>Монетный двор</h1>
9     </header>
10
11    <nav>
12      Санкт-Петербургский монетный двор<br>
13      Московский монетный двор<br>
14      Екатеринбургский монетный двор<br>
15      Сузунский монетный двор<br>
16    </nav>
17
18    <section>
19      <h1>Санкт-Петербургский монетный двор</h1>
20      <p>
21        Санкт-Петербургский монетный двор Гознака -
22        старейшее действующее предприятие по выпуску монетно-медальной прод
23      </p>
24      <p>
25        За почти три столетия производственной деятельности накоплен уникал
26        который успешно используется при изготовлении изделий.
27      </p>
28    </section>
29
30    <footer>
31      Copyright © http://www.mintspb.ru/
32    </footer>
33  </body>
34</html>
```

Рисунок 129 – Организация структуры с помощью семантических элементов

Семантические теги делают код более понятным для разработчиков, поисковых систем, скринридеров. Это улучшает SEO-оптимизацию и повышает доступность веб-страницы, поэтому второй вариант является предпочтительным.

Посмотрим подключенный файл со стилем. CSS-код в нем определяет стили для элементов `<header>`, `<nav>`, `<section>`, и `<footer>`, при этом выбирая не конкретные элементы по `id`, как в предыдущем упражнении, а по названию тэга (см. рисунок 130). Посмотрим, как отображается данная страница в браузере, и убедимся, что стиль применился (см. рисунок 131).

```

1  header {
2      background-color: green;
3      color: white;
4      text-align: center;
5      padding: 3px;
6  }
7  nav {
8      line-height: 30px;
9      background-color: silver;
10     height: 300px;
11     width: 270px;
12     float: left;
13     padding: 5px;
14 }
15 section {
16     width: 350px;
17     float: left;
18     padding: 10px;
19 }
20 footer {
21     background-color: green;
22     color: white;
23     clear: both;
24     text-align: center;
25     padding: 3px;
26 }
27

```

Рисунок 130 – Файл со стилем CSS

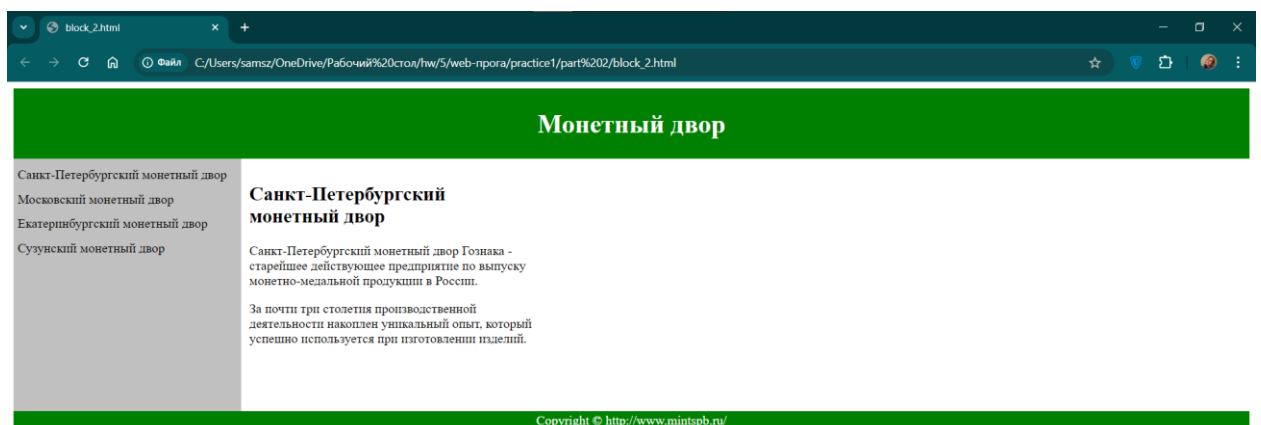


Рисунок 131 – Отображение страницы в браузере

Вывод

В ходе выполнения лабораторной работы были изучены основы работы с HTML, была подробно рассмотрена работа с текстом, со списками, с гиперссылками, изображениями и медиафайлами. Также была рассмотрена

работа с таблицами и их оформлением. Более того, было изучено применение форм, стилей CSS и правильная организация структуры документа. Поставленная в лабораторной работе цель была выполнена успешно, а ее выполнение помогло освоить процесс создания веб-страниц.