**МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ, СВЯЗИ И МАССОВЫХ**

**КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ**

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Поволжский государственный университет телекоммуникаций и информатики»

**КОЛЛЕДЖ СВЯЗИ**

УТВЕРЖДАЮ

Директор КС ПГУТИ

\_\_\_\_\_\_\_\_\_\_\_\_\_Андреев Р.В.

16.06.2021г.

**СБОРНИК**

**практических занятий**

по дисциплине

**ВЕБ – ДИЗАЙН**

для специальности: 09.02.03– Программирование в компьютерных системах

Номера занятий: **№ 1 – 20**

Сборник рассчитан на 40 часов

Составлен преподавателем Шомас Е.А.

Рассмотрен на заседании П(Ц)К

«Информационные системы и технологии»

Протокол № 1 от 15.06.2021г.

Председатель П(Ц)К\_\_\_\_\_\_\_\_Шомас Е.А.

Самара

2022 г

**ПЕРЕЧЕНЬ КОМПЕТЕНЦИЙ**

| **Общие компетенции** | |
| --- | --- |
| **ОК 1** | Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам |
| **ОК 2** | Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности |
| **ОК 3** | Планировать и реализовывать собственное профессиональное и личностное развитие |
| **ОК 4** | Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами |
| **ОК 5** | Осуществлять устную и письменную коммуникацию на государственном языке с учётом особенностей социального и культурного контекста |
| **ОК 6** | Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных общечеловеческих ценностей |
| **ОК 7** | Содействовать сохранению окружающей среды, ресурсосбережению, эффективно действовать в чрезвычайных ситуациях |
| **ОК 8** | Использовать средства физической культуры для сохранения и укрепления здоровья в процессе профессиональной деятельности и поддержания необходимого уровня физической подготовленности |
| **ОК 9** | Использовать информационные технологии в профессиональной деятельности |
| **ОК 10** | Пользоваться профессиональной документацией на государственном и иностранном языке |
| **ОК 11** | Планировать предпринимательскую деятельность в профессиональной сфере |
| **Профессиональные компетенции** | |
| **ПК 1.2** | Разрабатывать программные модули в соответствии с техническим заданием |
| **ПК 1.3** | Выполнять отладку программных модулей с использованием специализированных программных средств |
| **ПК 1.4** | Выполнять тестирование программных модулей |

**ПРАКТИЧЕСКОЕ ЗАНЯТИЕ №1**

**НАИМЕНОВАНИЕ**: Создание первой веб – страницы. Форматирование текста

1. **ЦЕЛЬ**: Научится применять основные теги HTML для создания веб – страниц. Формировать компетенции ОК 1 - 11; овладеть знаниями и умениями для освоения ПК 1.2, ПК 1.3, ПК 1.4
2. **ПОДГОТОВКА К ЗАНЯТИЮ**: по предложенной литературе повторить тему «Структура HTML- документа» и ответить на следующие вопросы:

2.1 Чем отличается текст от гипертекста?

2.2 Какие особенности HTML – документа вам известны?

2.3 Какое расширение имеют HTML – документы?

1. **ЛИТЕРАТУРА**:
   1. Шомас Е.А.Веб - дизайн. Учебное пособие. – Самара.: КС ПГУТИ, 2022. - 122c.
   2. <https://html5book.ru/osnovy-html/> - Основы HTML

**4.** **ПЕРЕЧЕНЬ ОБОРУДОВАНИЯ И ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**:

4.1 ПЭВМ, подключенные к сети Интернет

4.2 Браузеры

* 1. Текстовый редактор

**5. ЗАДАНИЕ:**

**Задание 1**

5.1.1 Создать веб – страницу, самостоятельно определив ее заголовок. Оформить фон страницы отличный от белого, а цвет текст на странице отличный от черного;

5.1.2 Скопировать любой текст (рассказ, реферат), оформить его, определив заголовки, подзаголовки, основной текст (абзацы);

5.1.3 Основные термины, важные фрагменты, выделить любым из способов форматирования теста (курсив, жирный текст, подчеркнутый, можно выбрать комбинации в оформлении);

**Задание 2**

5.2.1 Оформить фразу по приведенному ниже образцу:



В слове УТРО все буквы должны иметь **разные цвета**. В слове ПРОХЛАДОЙ оформить буквы ПРО – **красным** цветом, ОЙ – **синим**

**Задание 3**

5.3.1 Составить маркированный список – времена года (дни недели);

5.3.2 Составить нумерованный список – название месяцев (перечень предметов);

5.3.3 Составить комбинированный список Календарь (Расписание занятий):

* Зимние месяца:

1. Декабрь;
2. Январь;
3. Февраль;

* Весенние месяца:

1. …

5.3.4 К последнему заданию самостоятельно продумать способы оформления (цвета, форматирование и т.д.).

1. **ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ:**

6.1 По предложенной литературе изучить необходимый материал и ответить на вопросы для допуска к практическому занятию;

6.2 Выполнить задание практического занятия;

* 1. Дать ответы на контрольные вопросы;

**7. СОДЕРЖАНИЕ ОТЧЕТА**:

7.1 Наименование и цель работы

7.2 Код программы

7.3 Ответы на контрольные вопросы

* 1. Вывод о проделанной работе

**8. КОНТРОЛЬНЫЕ ВОПРОСЫ:**

8.1 Что такое веб – страница?

8.2 Что такое тег? Правило его написание. Основные виды тегов.

8.3 Перечислите обязательные теги.

**ПРИЛОЖЕНИЕ**:

**HTML (Hyper Text Markup Language)** – язык гипертекстовой разметки документов, который отвечает за структуру сайта. С помощью HTML текстовый документ разбивают на блоки смысловой информации (заголовки, параграфы, таблицы, рисунки и т.п.).

Основой языка HTML является дескриптор (тег). **Тег** определяет тип выводимого элемента HTML (например, заголовок, таблица, рисунок и т.п.). Сам тег не отображается браузером. Тег представляет собой последовательность элементов, заключенные в угловые скобки (< и >).

**Атрибуты –** необязательный набор параметров, определяющих дополнительные свойства элемента HTML (например, цвет или размер).

 Пример элемента HTML:

<h1 align= "center"> Глава 1</h1>, в этом примере:

<h1 align= "center"> – открывающий тег

</h1> – закрывающий тег

h1 – имя тега

align = "center" – атрибут

align – имя атрибута

"center" – значение атрибута

Глава 1 – содержание элемента HTML

**Структура HTML – документа**

Базовая структура HTML – документа будет выглядеть следующим образом:

<!DOCTYPE HTML>

<html>

<head>

<title>Заголовок</title>

<meta charset="UTF-8">

</head>

<body>

<div>Документ HTML5</div>

</body>

</html>

В разделе документа **body** располагается та информация, которая отображается в окне браузера. Данный тег имеет ряд атрибутов, определяющих внешний вид документа. Ниже приводится спецификация тега <BODY>.

<**body**

**text**="цвет текста"

**bgcolor**="цвет фона"

**background**="адрес фонового рисунка"

**link**="цвет непосещенной гиперссылки"

**vlink**="цвет посещенной гиперссылки"

**alink**="цвет активной гиперссылки">

**Создание заголовков**

Для разметки заголовков используются теги <**h1**>, <**h2**>, <**h3**>, <**h4**>, <**h5**> и <**h6**>. Эти теги требуют соответствующего закрывающего тега. Синтаксис тегов заголовков:

<**hn align**="выравнивание">

Атрибут align определяет способ выравнивания заголовка. Значения атрибута:

* **left** – текст выравнивается по левому краю окна браузера. Это значение используется по умолчанию, т.е. когда атрибут не указан;
* **center** – текст выравнивается по центру окна браузера;
* **right** – текст выравнивается по правому краю окна браузера;

**Создание абзацев:**

Одним из первых правил составления любых документов является разбиение его текста на отдельные абзацы, выражающие законченную мысль. В HTML-документе разделение на абзацы производится с помощью специального тега **<p>**. Синтаксис тега аналогичен тегу для создания заголовков.

**Создание списков**

**Списки –** применяются для организации и группировки данных. В HTML выделяются следующие типы списков:

**Маркированный список** – это список, в котором пункты отмечаются с помощью различных символов. Для их создания предусмотрен элемент **ul**, требующий наличие закрывающего тега. Пункты списка находятся внутри элемента **ul**. Каждый пункт начинается с элемента **li**.

Синтаксис элемента:

**<ul type = “**вид маркера**”>**

Атрибут принимает следующие значения:

* + **circle** – маркер в виде пустого круга;
  + **square** – маркер в виде квадрата;
  + **disc**- маркер в виде черного круга.

**Нумерованный список** применяется, когда порядок следования пунктов списка имеет большое значение. Для его создания предусмотрен элемент **ol**, требующий наличие закрывающего тега. Пункты списка находятся внутри элемента **ol**. Каждый пункт начинается с элемента **li**.

Синтаксис элемента:

**<ol type = «**вид символа, используемого для нумерации» **start=** «число, с которого начинается нумерация списка»>

Атрибут **type** принимает следующие значения:

* **А** - заглавные буквы латинского алфавита;
* **a** – строчные буквы латинского алфавита;
* **I** – римские цифры;
* **1** – арабские цифры.

**Вложенный список –** комбинированный список, в который вкладывается и нумерованный и маркированный списки.

**Форматирование текста**

Элемент **font** – задает параметры шрифта для текста. Параметры текста задаются с помощью атрибутов тега:

* **face** – название шрифта;
* **size** – размер шрифта;
* **color** – цвет шрифта.

**ПРАКТИЧЕСКОЕ ЗАНЯТИЕ №2**

**НАИМЕНОВАНИЕ**: Вставка объектов на веб - страницу

1. **ЦЕЛЬ**: Научится размещать рисунки, аудио и видеофайлы на веб – странице. Формировать компетенции ОК 1 – ОК11; овладеть знаниями и умениями для освоения ПК 1.2, ПК 1.3, ПК 1.4
2. **ПОДГОТОВКА К ЗАНЯТИЮ**: по предложенной литературе повторить тему «Вставка объектов на веб – страницу» и ответить на следующие вопросы:

2.1 Перечислите основные атрибуты элемента img?

2.2 Какие форматы изображений можно вставить на веб страницу?

2.3 Какие объекты кроме изображений можно внедрить в HTML – документ?

1. **ЛИТЕРАТУРА**:
   1. Шомас Е.А.Веб - дизайн. Учебное пособие. – Самара.: КС ПГУТИ, 2022. - 122c.

3.2 <https://html5book.ru/osnovy-html/> - Основы HTML

**4. ПЕРЕЧЕНЬ ОБОРУДОВАНИЯ И ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**:

4.1 ПЭВМ, подключенные к сети Интернет

4.2 Браузеры

* 1. Текстовый редактор

1. **ЗАДАНИЕ**:

5.1 На веб – странице разместить 3-4 изображения, выставить для них размеры, чтобы смотрелись аккуратно. Для картинок добавить небольшую рамку и подписать картинку. Обязательно вставить атрибут **alt** с альтернативным текстом, в случае, если картинки не загрузятся;

* 1. Определит бегущую строку;
  2. Вставить видеоизображение и звук;

5.4 Отделить задания друг от друга горизонтальной линией.

**6. ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ:**

6.1 По предложенной литературе изучить необходимый материал и ответить на вопросы для допуска к практическому занятию;

6.2 Выполнить задание практического занятия;

* 1. Дать ответы на контрольные вопросы;

**7. СОДЕРЖАНИЕ ОТЧЕТА**:

7.1 Наименование и цель работы

7.2 Код программы

7.3 Ответы на контрольные вопросы

* 1. Вывод о проделанной работе

**8. КОНТРОЛЬНЫЕ ВОПРОСЫ:**

8.1 С помощью какого тега производится вставка изображений на веб – страницу? Какие форматы могут иметь файлы с изображением?

8.2 Как задать изображение в виде фона веб – страницы?

**ПРИЛОЖЕНИЕ**:

**Работа с изображениями**

На веб – страницах используются следующие форматы графических изображений: **\*.jpg, \*.gif, \*.png.** Для вывода изображений в HTML – документ применяется элемент **img**. Элемент имеет два важных параметра:

* **src -** путь к изображению. Это может быть относительный или абсолютный путь в файловой системе или адрес в интернете;

**- alt** - текстовое описание изображения.

Дополнительно в элементе img можно указать размеры изображения через атрибуты:

* **width** = “ширина (в пикселях или %)”;
* **height**= “высота (в пикселях или %)”.

**Вставка горизонтальной линии**

Горизонтальная линия визуально подчеркивает законченность той или иной области документа. Тегом **<hr>** создается горизонтальная линия.

**Вставка бегущей строки**

**<marquee> –** начало бегущей строки, **</marquee> –** конец бегущей строки. Фрагмент текста, картинки или любые другие объекты между этими тегами будут двигаться так, как определят атрибуты.

**Мультимедиа**

**Видео**

Для воспроизведения видео в html5 используется элемент **video**. Чтобы настроить элемент, используются следующие атрибуты:

* **src** – источник видео;
* **width** – ширина элемента;
* **height** – высота элемента;
* **controls** – добавление элемента управления воспроизведением;
* **autoplay** – устанавливает автовоспроизведение;
* **loop** – количество повторений;
* **muted** – отключает звук по умолчанию.

**Аудио**

Для воспроизведения звука без видео в html5 применяется элемент audio. Для его настройки используют следующие атрибуты:

* **src** – путь к аудиофайлу;
* **controls** – добавление элемента управления воспроизведением;
* **autoplay** – устанавливает автовоспроизведение;
* **loop** – количество повторений;
* **muted** – отключает звук по умолчанию;
* **preload** – устанавливает режим загрузки файла.

**Ссылки**

Ссылки обеспечивают навигацию между отдельными документами. Определяются ссылки элементом **<a></a>**, который имеет следующие атрибуты:

* **href** – определяет адрес ссылки;
* **target** – определяет, как документ по ссылке должен открываться. Атрибут может принимать следующие значения: \_blank – открывается документ в новой вкладке.

**ПРАКТИЧЕСКОЕ ЗАНЯТИЕ №3**

**НАИМЕНОВАНИЕ**: Использование ссылок на веб - страницах

1. **ЦЕЛЬ**: Научится вставлять различные виды ссылок на веб - страницы. Формировать компетенции ОК 1 – ОК11; овладеть знаниями и умениями для освоения ПК 1.2, ПК 1.3, ПК 1.4
2. **ПОДГОТОВКА К ЗАНЯТИЮ**: по предложенной литературе повторить тему «Вставка ссылок на веб – страницу» и ответить на следующие вопросы:
   1. Что такое ссылка? Для каких целей она организуется?
   2. С помощью какого тега формируется ссылка?
3. **ЛИТЕРАТУРА**:
   1. Шомас Е.А.Веб - дизайн. Учебное пособие. – Самара.: КС ПГУТИ, 2022. - 122c.

3.2 <https://html5book.ru/osnovy-html/> - Основы HTML

**4. ПЕРЕЧЕНЬ ОБОРУДОВАНИЯ И ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**:

4.1 ПЭВМ, подключенные к сети Интернет

* 1. Браузеры

4.3 Текстовый редактор

**5. ЗАДАНИЕ**:

5.1 Создайте веб – страницу на определенную тематику. Тему определите через заголовок первого уровня. Выравнивание по центру.

5.2 Создайте несколько ссылок друг под другом, которые будут являться содержанием разделов вашей тематической веб – страницы.

5.3 Под каждую ссылку заведите отдельные веб страницы. Каждая страница будет являться разделом темы. Заполните их содержимым, включая рисунки.

5.4 Соедините отдельные страницы в единый ресурс.

5.5 Одна из страниц должна содержать внутренние ссылки (якоря);

5.6 Попробуйте изменить цвет ссылок.

**6. ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ:**

6.1 По предложенной литературе изучить необходимый материал и ответить на вопросы для допуска к практическому занятию;

* 1. Выполнить задание практического занятия;

6.3 Дать ответы на контрольные вопросы;

**7. СОДЕРЖАНИЕ ОТЧЕТА**:

7.1 Наименование и цель работы

7.2 Код программы

* 1. Ответы на контрольные вопросы

7.4 Вывод о проделанной работе

**8. КОНТРОЛЬНЫЕ ВОПРОСЫ:**

8.1 Основные виды ссылок?

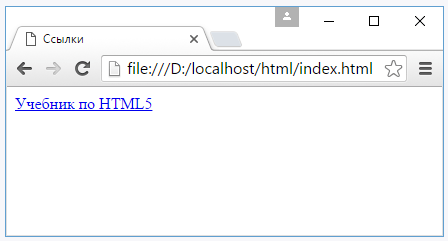
8.2 Каким образом можно изменить цвет у ссылок?

**ПРИЛОЖЕНИЕ**:

Ссылки обеспечивают навигацию между отдельными документами. Определяются ссылки элементом **<a></a>**, который имеет следующие атрибуты:

* **href** – определяет адрес ссылки.

<a href="content.html">Учебник по HTML5</a>



В примере для ссылки используется относительный путь ***content.html***, т.е. в одной папке с данным документом должен находиться файл content.html, на который будет идти переход по нажатию ссылки. Также может использоваться абсолютный путь, с полным указанием адреса, где находится документ.

**Навигация внутри документа (якоря)**

Можно организовывать внутренние ссылки, которые будут переходить к определенным блокам внутри документа. Удобно в том случае, когда на странице слишком много текста. Внутренние ссылки также создаются с помощью тега <**a**>, в атрибуте **href** содержится имя указателя (***якоря***), а не URL – адрес. Перед именем ставится знак **#.** Переход будет осуществлен к той части документа, где указан **id** с нужным именем.

<h1> Времена года</h1>

<h2> Оглавление</h2>

<a href="#p1"> Лето</a> <!-- создаём якорь, указав #id элемента-->

<a href="#p2"> Осень</a>

<a href="#p3"> Зима</a>

<a href="#p4"> Весна</a>

<p id="p1"> ...</p> <!-- добавляем соответствующий id элементу-->

<p id="p2"> ...</p>

<p id="p3"> ...</p>

<p id="p4"> ...</p>

**ПРАКТИЧЕСКОЕ ЗАНЯТИЕ №4**

**НАИМЕНОВАНИЕ**: Создание интерактивных карт

**2. ЦЕЛЬ**: Научится делась разметку изображения по областям, формировать ссылки из определенной области на внешнюю страницу. Формировать компетенции ОК 1 – ОК11; овладеть знаниями и умениями для освоения ПК 1.2, ПК 1.3, ПК 1.4

1. **ПОДГОТОВКА К ЗАНЯТИЮ**: по предложенной литературе повторить тему «Вставка ссылок на веб – страницу» и ответить на следующие вопросы:
   1. Как создаются интерактивные карты?
   2. Приведите пример использования интерактивной карты?
2. **ЛИТЕРАТУРА**:
   1. Шомас Е.А.Веб - дизайн. Учебное пособие. – Самара.: КС ПГУТИ, 2022. - 122c.

3.2 <https://html5book.ru/osnovy-html/> - Основы HTML

**4. ПЕРЕЧЕНЬ ОБОРУДОВАНИЯ И ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**:

* 1. ПЭВМ, подключенные к сети Интернет

4.2 Браузеры

4.3 Текстовый редактор

**5. ЗАДАНИЕ**:

* 1. Создаем интерактивную карту (метро, города и т.д.)
  2. На карте определяем пять областей: три области оформляем как ссылки на внешний ресурс, две оформляем как самостоятельные веб – страницы (каждая страница оформляется вами самостоятельно, должна иметь рисунок и пояснительный текст, который соответствует месту ссылки)

3.4 Внешние ссылки должны открываться в отдельной вкладке, а не заменять собой главную страницу (добавляем атрибут **target=“\_blank”)**

**6. ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ:**

6.1 По предложенной литературе изучить необходимый материал и ответить на вопросы для допуска к практическому занятию;

* 1. Выполнить задание практического занятия;

6.3 Дать ответы на контрольные вопросы;

**7. СОДЕРЖАНИЕ ОТЧЕТА**:

7.1 Наименование и цель работы

* 1. Код программы

7.3 Ответы на контрольные вопросы

7.4 Вывод о проделанной работе

**8. КОНТРОЛЬНЫЕ ВОПРОСЫ:**

8.1 Назначение атрибута usemap?

8.2 Какие виды областей существуют при формировании карты?

**ПРИЛОЖЕНИЕ**:

**Алгоритм создания интерактивной карты:**

1. В сети интернет находим подходящую карту, например:



1. Вставляем изображение с картой на веб – страницу как обычный рисунок;
2. Добавляем атрибут картирования – usemap = “#...”, где вместо точек пишется имя карты (название любое на английском языке);
3. Карта определяется тегами <map name = “…”> … </map>, где вместо точек пишется имя карты (# уже не нужен);
4. Между <map> пишутся области с координатами, которые будут являться ссылками.

**Виды областей и их координаты**

Код **координат областей** имеет вид:

Если область прямоугольная -

**<area shape=”rect” coords=”n,n,n,n” href=”место ссылки”>**

Если область круглая -

**<area shape=”circle” coords=”n,n,n” href=”место ссылки”>**

**Пример кода:**

     <body>

          <img src="\*.jpg" usemap=”**#karta**”>

          <map name=”**karta**”>

                <area shape=”circle” coords=”152,167,10” href=”page1.html”>

                <area shape=”circle” coords=”189,102,10” href=”page2.html”>

        </map>

    </body>

**Определение координат:**

1. Открываем карту в Paint
2. Наводим курсор на нужную точку и в строке состояния видим координаты этой точки.

Если область круглая, то третью координату (радиус действия ссылки) проставляем самостоятельно. Если прямоугольная область – это координаты двух точек по диагонали, пересечение которых формирует прямоугольник.

**ПРАКТИЧЕСКОЕ ЗАНЯТИЕ №5**

**НАИМЕНОВАНИЕ**: Создание таблиц на веб – страницах

**1. ЦЕЛЬ**: Применение таблиц на веб – страницах, формирование компетенций ОК 1 – ОК11; овладеть знаниями и умениями для освоения ПК 1.2, ПК 1.3, ПК 1.4

**2. ПОДГОТОВКА К ЗАНЯТИЮ**: по предложенной литературе повторить тему «Таблицы на веб – страницах» и ответить на следующие вопросы:

2.1 Каким образом можно вставить таблицу?

2.2 Как формируется ячейка в таблице?

1. **ЛИТЕРАТУРА**:
   1. Шомас Е.А.Веб - дизайн. Учебное пособие. – Самара.: КС ПГУТИ, 2022. - 122c.

3.2 <https://html5book.ru/osnovy-html/> - Основы HTML

**4. ПЕРЕЧЕНЬ ОБОРУДОВАНИЯ И ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**:

4.1 ПЭВМ, подключенные к сети Интернет

* 1. Браузеры

4.3 Текстовый редактор

**5. ЗАДАНИЕ:**

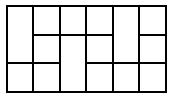
5.1 Создайте простую таблицу следующего вида:

**Прайс - лист**

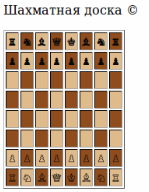
| **Товар** | **Код** | **Количество** | **Цена** |
| --- | --- | --- | --- |
| Молоко | 0021 | 3 к. | 180 руб. |
| Масло | 0032 | 2 п. | 210 руб. |
| Сыр плавленый | 0012 | 1 уп. | 115 руб. |
| ИТОГО: | | | **505 руб.** |

\* таблица может быть заполнена другими данными

5.2 Создайте сложную таблицу с объединением ячеек:



5.3 Создайте таблицу в виде шахматной доски. Расставьте на ней фигуры использую специальные символы html.



1. **ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ:**

6.1 По предложенной литературе изучить необходимый материал и ответить на вопросы для допуска к практическому занятию;

* 1. Выполнить задание практического занятия;

6.3 Дать ответы на контрольные вопросы;

**7. СОДЕРЖАНИЕ ОТЧЕТА**:

7.1 Наименование и цель работы

7.2 Код программы

* 1. Ответы на контрольные вопросы

7.4 Вывод о проделанной работе

**8. КОНТРОЛЬНЫЕ ВОПРОСЫ:**

8.1 Назначение и виды таблиц в HTML – документе?

8.2 В чем различие тега <th> и тега <td>?

**ПРИЛОЖЕНИЕ**:

Каждая таблица должна начинаться тегом <**table**> и заканчиваться тегом <**/table**>. Все элементы таблицы должны быть вложены внутрь этого тега. Название таблицы определяется при помощи тегов **<caption> …</caption>**. Выравнивание наименования задается при помощи атрибута **align**, который может принимать значения **top** (над таблицей) и **bottom** (под таблицей).

**Создание ячеек таблицы:**

Для начала нужно создать необходимое количество строк, затем поделить строки столбцами на ячейки. Количество пар тегов <**tr**> и <**/tr**> определяет количество горизонтальных строк в вашей таблице. Количество пар тегов **<td>…</td**>, расположенных между тегами соответствующей строки, определяет количество ячеек (столбцов) в пределах строки.

Пример:

<body>

<table border=1>

<tr> <td> ячейка1 </td> <td> ячейка2</td> </tr>

<tr> <td>ячейка3 </td> <td> ячейка4 </td> </tr>

</table>

</body>

**Спецификация тега <table>**

<**table align**= “выравнивание”

**border**= “толщина рамки”

**height**= “высота таблицы”

**width**= “ширина таблицы”

**valign**= “вертикальное выравнивание”>

**Спецификация тега <tr>**

<**tr align**= “выравнивание”

**valign**= “вертикальное выравнивание”

**bgcolor**= “цвет фона”>

**Спецификация тега <td>**

**<td align**= “выравнивание”

**bgcolor**= “цвет фона”

**colspan**= “количество ячеек объединенных по горизонтали”

**rowspan**= “количество ячеек объединенных по вертикали”

**height**= “высота ячейки”

**width**= “ширина ячейки”>

**ПРАКТИЧЕСКОЕ ЗАНЯТИЕ №6**

**НАИМЕНОВАНИЕ**: Неявные таблицы на веб – страницах

**1. ЦЕЛЬ**: Применение таблиц на веб – страницах, формирование компетенций ОК 1 – ОК11; овладеть знаниями и умениями для освоения ПК 1.2, ПК 1.3, ПК 1.4

**2. ПОДГОТОВКА К ЗАНЯТИЮ**: по предложенной литературе повторить тему «Таблицы на веб – страницах» и ответить на следующие вопросы:

2.1 Перечислите основные теги создания таблиц в HTML – документе

2.2 С помощью какого тега можно подписать таблицу?

2.3 Какой атрибут отвечает за объединение ячеек в таблицу по горизонтали (вертикали)?

1. **ЛИТЕРАТУРА**:
   1. Шомас Е.А.Веб - дизайн. Учебное пособие. – Самара.: КС ПГУТИ, 2022. - 122c.

3.2 <https://html5book.ru/osnovy-html/> - Основы HTML

**4. ПЕРЕЧЕНЬ ОБОРУДОВАНИЯ И ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**:

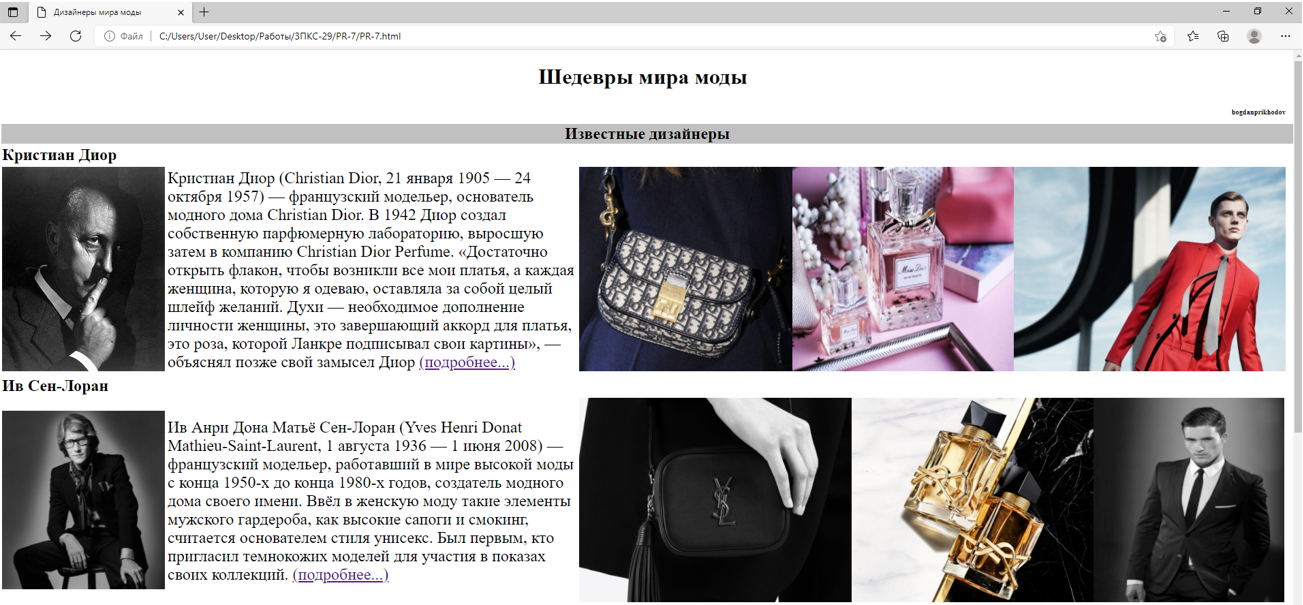
4.1 ПЭВМ, подключенные к сети Интернет

* 1. Браузеры

4.3 Текстовый редактор

**5. ЗАДАНИЕ:**

5.1 Создайте тематический сайт в табличной форме следующего вида:



Тема сайта выбирается произвольно. Должна быть представлена информация о трех – четырех персонах.

Обязательно должна быть ссылка на сайт, где можно более подробно узнать информацию о данном человеке.

1. **ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ:**

6.1 По предложенной литературе изучить необходимый материал и ответить на вопросы для допуска к практическому занятию;

* 1. Выполнить задание практического занятия;

6.3 Дать ответы на контрольные вопросы;

**7. СОДЕРЖАНИЕ ОТЧЕТА**:

7.1 Наименование и цель работы

* 1. Код программы

7.3 Ответы на контрольные вопросы

7.4 Вывод о проделанной работе

**8. КОНТРОЛЬНЫЕ ВОПРОСЫ:**

8.1 Для чего в дизайне используются неявные таблицы?

8.2 Какие теги используются для объединения ячеек в таблице?

8.3 Каким тегом установить границы в таблице?

**ПРИЛОЖЕНИЕ**:

В HTML-таблицах может содержаться информация любого типа. С их помощью можно более эффектно расположить на странице текст и графику, а также создать четкую структуру веб-страницы.

Вся таблица располагается внутри тегов <**table**>…</**table**>. Строка в таблице создается через элемент <**tr**> </**tr**>, а внутри размещаются элементы <**td**> </**td**>, определяющие ячейки внутри строки (соответствуют столбцам в таблице).

**ПРАКТИЧЕСКОЕ ЗАНЯТИЕ №7**

**НАИМЕНОВАНИЕ**: Создание форм на веб - страницах

**1. Цель**: Научится создавать формы на веб страницах и размещать в них элементы управления. Формировать компетенции ОК 1 – ОК11; овладеть знаниями и умениями для освоения ПК 1.2, ПК 1.3, ПК 1.4

**2. ПОДГОТОВКА К ЗАНЯТИЮ**: по предложенной литературе повторить тему «Создание форм» и ответить на следующие вопросы:

2.1 Перечислите основные теги организации форм?

2.2 Какие атрибуты и значения принимает тег <input>?

* 1. Как организуется текстовое поля для сбора комментариев от посетителей сайта?

**3. ЛИТЕРАТУРА**:

* 1. Шомас Е.А.Веб - дизайн. Учебное пособие. – Самара.: КС ПГУТИ, 2022. - 122c.

3.2 <https://html5book.ru/osnovy-html/> - Основы HTML

**4. ПЕРЕЧЕНЬ ОБОРУДОВАНИЯ И ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**:

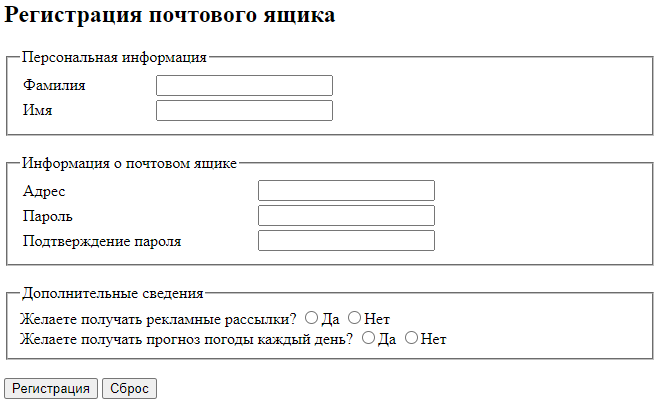
4.1 ПЭВМ, подключенные к сети Интернет

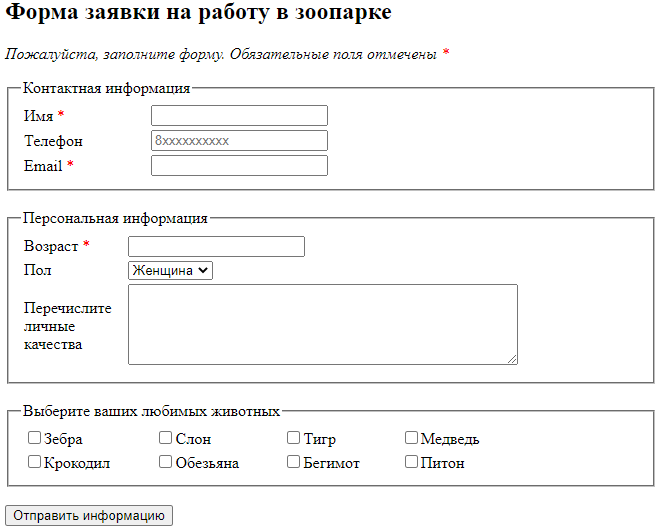
4.2 Браузеры

* 1. Текстовый редактор

**5. ЗАДАНИЕ**:

5.1 Создайте форму заявки по следующему образцу:





5.2 Продумайте и составьте собственную форму по оформлению определенного заказа товара через интернет – магазин (или форму анкетирования по определенному вопросу). Оформление продумайте самостоятельно.

**6. ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ:**

6.1 По предложенной литературе изучить необходимый материал и ответить на вопросы для допуска к практическому занятию;

6.2 Выполнить задание практического занятия;

* 1. Дать ответы на контрольные вопросы;

**7. СОДЕРЖАНИЕ ОТЧЕТА**:

7.1 Наименование и цель работы

7.2 Код программы

7.3 Ответы на контрольные вопросы

* 1. Вывод о проделанной работе

**8. КОНТРОЛЬНЫЕ ВОПРОСЫ:**

8.1 Назначение форм при создании сайта?

8.2 Основные виды форм?

8.3 В чем отличие кнопки – переключателя и кнопки - флажка?

**ПРИЛОЖЕНИЕ**:

Формы в html представляют один из способов для ввода и отправки данных. При создании форм необходимо следить за тем, чтобы предложенные поля были удобны для посетителей и приемлемы для ответов на поставленные вопросы.

Форма определяется тегами: **<form>...</form>**. Все, что заключено в этом теге, будет считаться формой.

У тега **<form>** имеются атрибуты:

**action**, который используется для того, чтобы соединить форму с программой по сбору данных и передать значение на нужную страницу.

**method** – определяется каким способом будут передаваться данные, занесенные в форму.

<form action=”url”>

Содержание формы с тегами и атрибутами

</form>

**Содержание формы** - это теги для сбора данных (они используются вместе с атрибутами):  
**<input>** - тег для сбора информации от посетителей веб- страниц. Атрибуты тега:

**name**="…"- имя переменной в которой будут переданы значения в данном элементе формы.

**type**="…"- указывает на **способ** обработки собранных данных.

Атрибуты name и type используются совместно с тегом input.

Способов **приёма и обработки данных** несколько видов - это различные значения атрибута type для тега input:

**text** – текстовое поле;

**password** – текстовое поле для ввода пароля;

**radio** – кнопка – переключатель;

**checkbox** – кнопка флажок;

**submit** – кнопка отправки формы;

**reset** – кнопка для сброса введенных данных;

**date** – поле для ввода даты;

**email** – поле для ввода адреса электронной почты;

**number** – поле для ввода чисел;

**tel** – поле для ввода телефона;

**file** – поле для выбора отправляемого файла;

**ПРАКТИЧЕСКОЕ ЗАНЯТИЕ № 8**

**НАИМЕНОВАНИЕ**: Блочная и семантическая верстка HTML – страницы

**1. Цель**: Научится подключать внешний CSS- файл и влиять на поведение блоков в HTML - документе. Формировать компетенции ОК 1 – ОК11; овладеть знаниями и умениями для освоения ПК 1.2, ПК 1.3, ПК 1.4

**2. ПОДГОТОВКА К ЗАНЯТИЮ**: по предложенной литературе повторить темы «Введение в CSS», «Семантическая верстка сайта» и ответить на следующие вопросы:

* 1. В чем заключается отличие тега <div> от <span>?

2.2 Назначение таблиц стилей CSS?

2.3 Перечислите способы подключения стилей к HTML - документу?

**3. ЛИТЕРАТУРА**:

* 1. Шомас Е.А.Веб - дизайн. Учебное пособие. – Самара.: КС ПГУТИ, 2022. - 122c.

3.2 <https://html5book.ru/osnovy-html/> - Основы HTML

**4. ПЕРЕЧЕНЬ ОБОРУДОВАНИЯ И ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**:

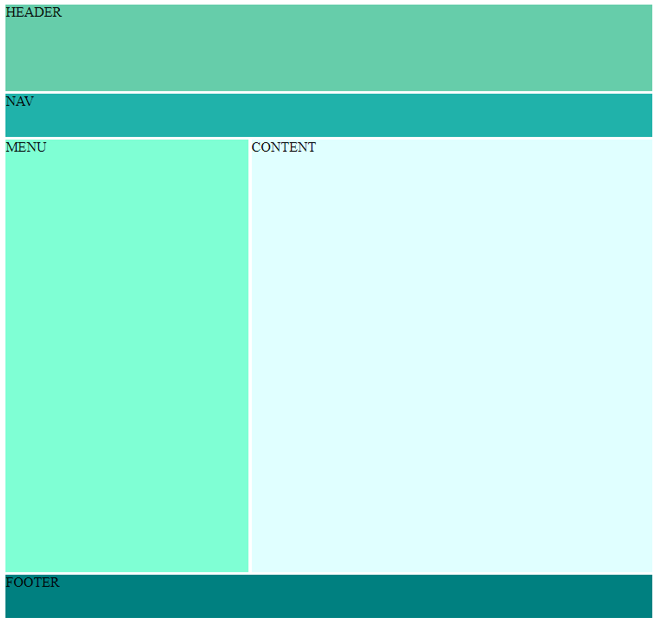
4.1 ПЭВМ, подключенные к сети Интернет

* 1. Браузеры

4.3 Текстовый редактор

**5. ЗАДАНИЕ**:

5.1 С помощью тегов <div> сверстать шаблон согласно следующего примера. При оформлении блоков использовать цвета одного оттенка. Все оформление осуществлять через внешний CSS – файл.



5.2 С помощью семантических тегов переверстать шаблон. Все оформление осуществлять через внешний CSS – файл.

**6. ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ:**

6.1 По предложенной литературе изучить необходимый материал и ответить на вопросы для допуска к практическому занятию;

* 1. Выполнить задание практического занятия;

6.3 Дать ответы на контрольные вопросы;

**7. СОДЕРЖАНИЕ ОТЧЕТА**:

7.1 Наименование и цель работы

7.2 Код программы

* 1. Ответы на контрольные вопросы

7.4 Вывод о проделанной работе

**8. КОНТРОЛЬНЫЕ ВОПРОСЫ:**

8.1 Из чего состоит правило CSS?

8.2 Назначение панели Инструменты разработчика? Как вызвать данную панель?

8.3 Какое свойство CSS позволяет сделать элемент из строчного блочным и наоборот?

8.4 Разница между селекторами id и class?

**ПРИЛОЖЕНИЕ**:

**Каскадные таблицы стилей (CSS)** предоставляют дополнительные возможности для разметки HTML – документа и свободу по созданию уникального дизайна веб – страниц.

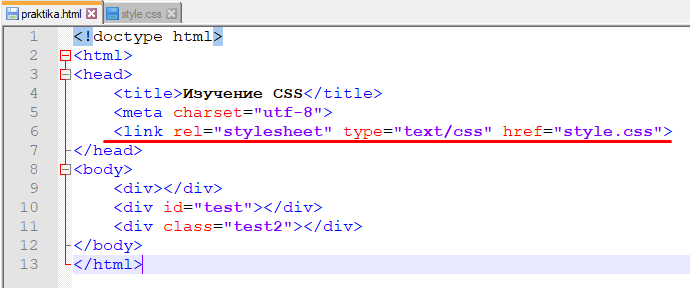
Файл с CSS – кодом сохраняется с расширением .css, имя файла может быть любым. Для подключения внешней таблицы стилей ее прикрепляют к документу HTML с помощью тега <link>, который должен располагаться внутри тега <head>.

<head>

**<link rel="stylesheet" type="text/css" href="styles.css">**

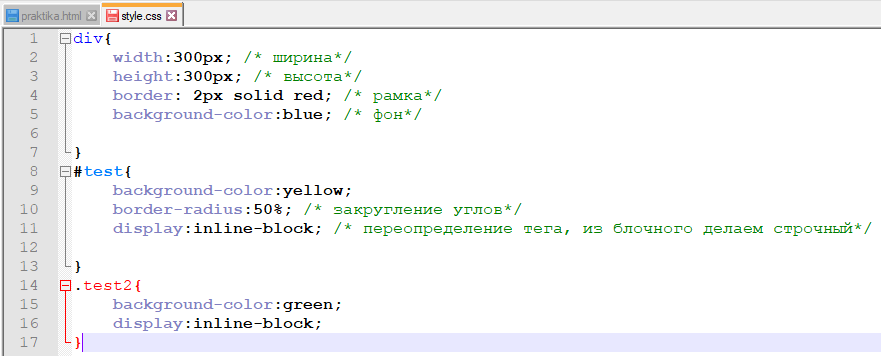
</head>

**HTML - документ**

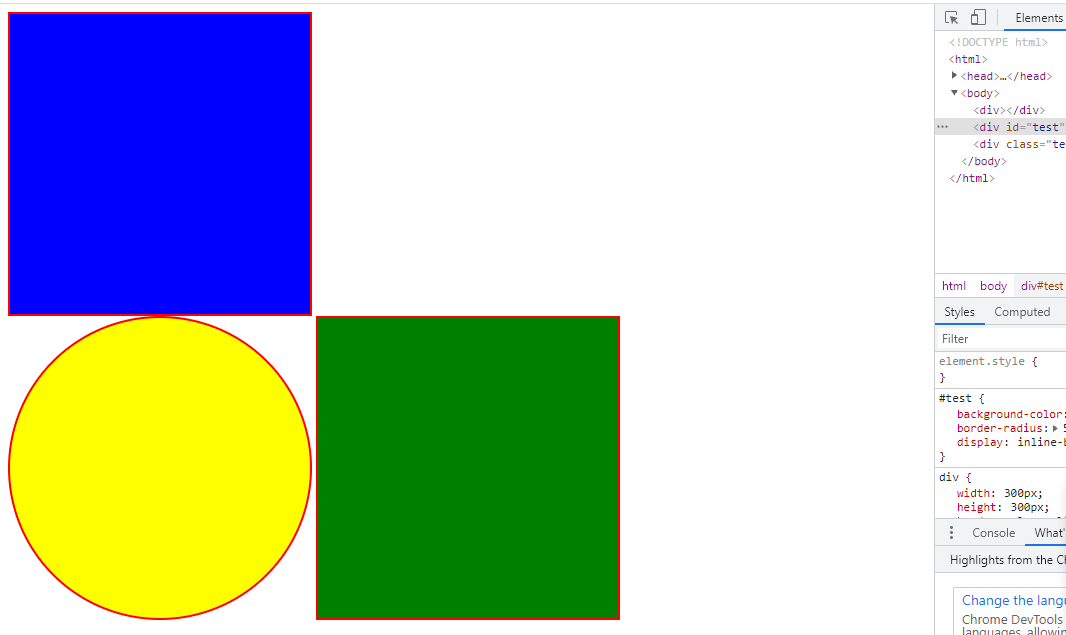


В CSS – файле прописываются все стили по оформлению того или иного документа. Обращаться к HTML – элементам можно непосредственно к тегам, или через селекторы id или class.

**CSS - файл**



**Вид в браузере**



**Инструменты разработчика – F12** – необходимы, чтобы посмотреть Блочную модель элементов в браузере.

**Семантическая верстка**

При создании верстки нужно учитывать семантику, то есть смысловое (логическое) значение элементов.

Основная цель любой HTML- вёрстки — *передача смысла блоков*. Часто, помимо пользователей, по нашим страницам ходят и роботы. Они собирают и анализируют информацию страницы. К примеру, поисковые роботы просматривают всю страницу и определяют её полезность, а также уникальность. И если для человека достаточно просто разделить страницу на части и найти шапку, основной контент, футер, то для робота это достаточно сложная задача. Робот видит исключительно вёрстку и не может «глазами» отделить части макета.

**ПРАКТИЧЕСКОЕ ЗАНЯТИЕ № 9**

**НАИМЕНОВАНИЕ**: Форматирование текста средствами CSS

1. **Цель**: Применить оформление веб – страницы средствами CSS. Формировать компетенции ОК 1 – ОК11; овладеть знаниями и умениями для освоения ПК 1.2, ПК 1.3, ПК 1.4

**ПОДГОТОВКА К ЗАНЯТИЮ**: по предложенной литературе повторить тему «Семантическая верстка сайта» и ответить на следующие вопросы:

* 1. Какие виды верстки вам известны?

2.2 Что значит семантическая верстка?

**3. ЛИТЕРАТУРА**:

* 1. Шомас Е.А.Веб - дизайн. Учебное пособие. – Самара.: КС ПГУТИ, 2022. - 122c.

3.2 <https://html5book.ru/css-css3> - Основы CSS

1. **ПЕРЕЧЕНЬ ОБОРУДОВАНИЯ И ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**:

4.1 ПЭВМ, подключенные к сети Интернет

4.2 Браузеры

* 1. Текстовый редактор

**5. ЗАДАНИЕ**:

5.1 В шапке шаблона подключить сторонний шрифт и написать тест в <h1> «Изучение CSS». К самому блоку шапки применить градиентную заливку из заранее определенного цветового диапазона.



5.2 В области контента сверстанного шаблона поместить небольшой текст (рассказ или сказку). Определить заголовок, где указать название произведения, автора. Обязательно должен присутствовать рисунок к тексту. Выполнить обтекание картинки текстом (float).

**Требования к оформлению:**

* текст должен иметь отступ (padding).
* размер шрифта 12pt,
* красная строка 40px (text-indent:40px;),
* Межстрочный интервал – 1.5 (line-height:1.5;)
* Текст выравнен – по ширине (text-align:justify;)
* Шрифт – любой, без засечек (Arial, Verdana …).
* К заголовкам можно подключить сторонний шрифт.
* Фон для области делать таким, чтобы текст был читабельным.

Если вдруг блок начинает выпадать из общего потока примените к нему правило -  overflow:hidden; Оно влияет на переполнение блока невидимыми отступами.

Пример оформления веб – страницы представлен в Приложении к практическому занятию.

1. **ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ:**

6.1 По предложенной литературе изучить необходимый материал и ответить на вопросы для допуска к практическому занятию;

6.2 Выполнить задание практического занятия;

* 1. Дать ответы на контрольные вопросы;

**7. СОДЕРЖАНИЕ ОТЧЕТА**:

7.1 Наименование и цель работы

7.2 Код таблицы CSS

7.3 Ответы на контрольные вопросы

* 1. Вывод о проделанной работе

**8. КОНТРОЛЬНЫЕ ВОПРОСЫ:**

8.1 Какие виды градиента вам известны?

8.2 Какое правило позволяет повлиять на переполнение блока?

8.3 Какая директива и правило CSS позволяет подключить сторонний шрифт?

8.4 Как оформить красную строку через CSS?

**ПРИЛОЖЕНИЕ**:

Пример оформления веб – страницы:



**ПРАКТИЧЕСКОЕ ЗАНЯТИЕ № 10**

**НАИМЕНОВАНИЕ**: Создание меню навигации

1. **Цель**: Применить оформление веб – страницы средствами CSS. Формировать компетенции ОК 1 – ОК11; овладеть знаниями и умениями для освоения ПК 1.2, ПК 1.3, ПК 1.4

**ПОДГОТОВКА К ЗАНЯТИЮ**: по предложенной литературе повторить тему «Создание панели навигации средствами CSS» и ответить на следующие вопросы:

* 1. Что лежит в основе панели навигации?

2.2 С помощью каких элементов можно реализовать панель навигации?

**3. ЛИТЕРАТУРА**:

* 1. Шомас Е.А.Веб - дизайн. Учебное пособие. – Самара.: КС ПГУТИ, 2022. - 122c.

3.2 <https://html5book.ru/css-css3> - Основы CSS

1. **ПЕРЕЧЕНЬ ОБОРУДОВАНИЯ И ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**:

4.1 ПЭВМ, подключенные к сети Интернет

* 1. Браузеры

4.3 Текстовый редактор

**5. ЗАДАНИЕ**:

5.1 Открыть предыдущую работу, продолжать будем в ней

5.2 В блоке NAV сверстанного шаблона разместить горизонтальное меню навигации

5.3 В блоке MENU разместить вертикальное меню навигации.

**Требования к оформлению:**

* Оформить меню через маркерный список. Для элементов списка отменить маркер;
* Для различных видов ссылок (пункты меню) применить различное оформление;
* Обязательно продумать **hover** – эффект при нажатии на ссылку (пункт меню).

5.4 В подвал сайта (раздел FOOTER) поместить свои фамилию и имя, группу

1. **ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ:**

6.1 По предложенной литературе изучить необходимый материал и ответить на вопросы для допуска к практическому занятию;

* 1. Выполнить задание практического занятия;

6.3 Дать ответы на контрольные вопросы;

**7. СОДЕРЖАНИЕ ОТЧЕТА**:

7.1 Наименование и цель работы

* 1. Код таблицы CSS

7.3 Ответы на контрольные вопросы

7.4 Вывод о проделанной работе

**8. КОНТРОЛЬНЫЕ ВОПРОСЫ:**

8.1 Каким образом оформляется панель навигации на сайте?

8.2 Какие виды меню существуют?

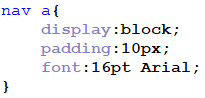
**ПРИЛОЖЕНИЕ**:

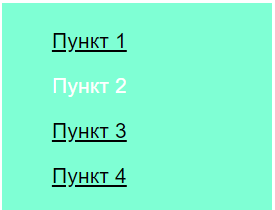
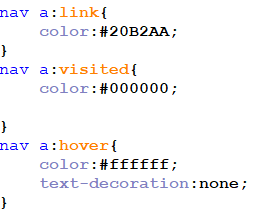
**Панель навигации** – это набор ссылок, часто выполненный в виде маркированного списка. Бывают вертикальными, горизонтальными, одноуровневыми и многоуровневыми, но в центре каждой навигации находится элемент <**a**>.

Важно помнить, что элемент <a> является строчным элементом, для которого нельзя указать ширину, высоту и отступы.

**Вертикальное меню**

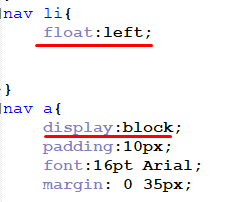
Для создания вертикального меню необходимо сделать ссылку блочным элементом.



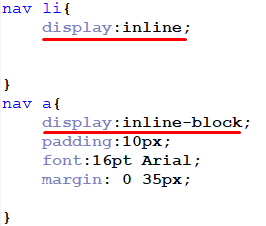


**Горизонтальное меню:**

**1 способ:** через свойство **float**



**2 способ:** через свойства **inline** и **inline-block**

**ПРАКТИЧЕСКОЕ ЗАНЯТИЕ № 11**

**НАИМЕНОВАНИЕ**: Позиционирование элементов

* 1. **Цель**: Применить различные виды позиционирования блоков на веб странице. Формировать компетенции ОК 1 – ОК11; овладеть знаниями и умениями для освоения ПК 1.2, ПК 1.3, ПК 1.4
  2. **ПОДГОТОВКА К ЗАНЯТИЮ**: по предложенной литературе повторить тему «Позиционирование элементов через CSS» и ответить на следующие вопросы:

2.1 Какие виды позиционирования элементов вы знаете?

2.2 Какой вид позиционирования принят по умолчанию?

**3. ЛИТЕРАТУРА**:

* 1. Шомас Е.А.Веб - дизайн. Учебное пособие. – Самара.: КС ПГУТИ, 2022. - 122c.

3.2 <https://html5book.ru/css-css3> - Основы CSS

1. **ПЕРЕЧЕНЬ ОБОРУДОВАНИЯ И ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**:
   1. ПЭВМ, подключенные к сети Интернет

4.2 Браузеры

4.3 Текстовый редактор

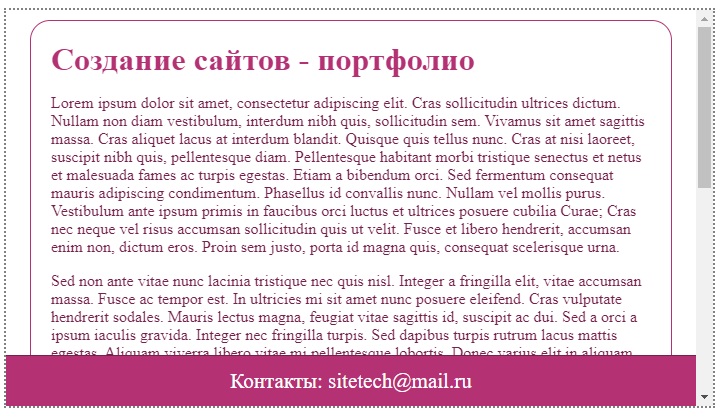
**5. ЗАДАНИЕ**:

***Вариант 1***

5.1 Абсолютное позиционирование повторить по образцу:

5.2 Фиксированное позиционирование повторить по образцу:

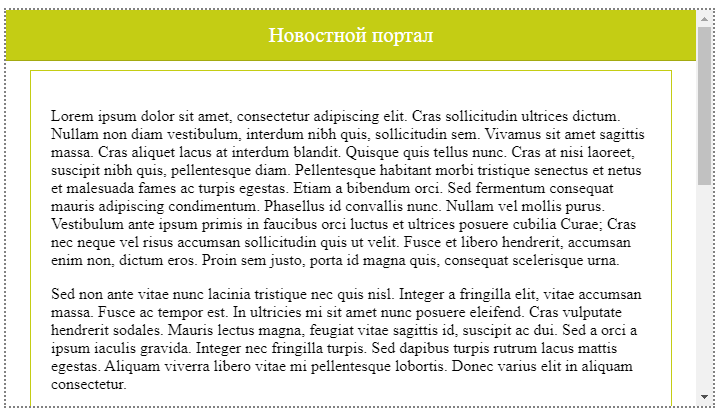


***Вариант 2***

5.1 Абсолютное позиционирование повторить по образцу:



5.2 Фиксированное позиционирование повторить по образцу:



1. **ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ:**

6.1 По предложенной литературе изучить необходимый материал и ответить на вопросы для допуска к практическому занятию;

* 1. Выполнить задание практического занятия;

6.3 Дать ответы на контрольные вопросы;

**7. СОДЕРЖАНИЕ ОТЧЕТА**:

* 1. Наименование и цель работы

7.2 Код таблицы CSS

7.3 Ответы на контрольные вопросы

7.4 Вывод о проделанной работе

**8. КОНТРОЛЬНЫЕ ВОПРОСЫ:**

8.1 Какое правило позволяет разместить элемент на новый слой?

8.2 Пример применения фиксированного позиционирования на веб – странице?

**ПРИЛОЖЕНИЕ**:

CSS предоставляет возможности по позиционированию элементов. Свойства **position** определяют алгоритм позиционирования, используемый для определения положения блока. Принимает следующие значения:

**static** – (значение по умолчанию) – является обычным блоком, позиционируемый в соответствии с нормальным потоком. Свойства left и top не применяются;

**absolute** – назначается тегу абсолютное позиционирование, при котором тег позиционируется относительно экрана либо относительно первого родительского тега, которому указано позиционирование;

**relative** – назначает тегу относительное позиционирование, при котором тег позиционируется относительно его текущего расположения;

**fixed** – назначает тегу фиксированное позиционирование, при котором тег зафиксирован в определенном месте окна браузера и не покидает своего положения при прокрутке.

**ПРАКТИЧЕСКОЕ ЗАНЯТИЕ № 12**

**НАИМЕНОВАНИЕ**: Трансформация, переходы и анимация

**1. ЦЕЛЬ**: Научиться средствами CSS3 «оживлять» веб – страницы, с применением трансформирования, переходов и анимации. Формировать компетенции ОК 1 – ОК11; овладеть знаниями и умениями для освоения ПК 1.2, ПК 1.3, ПК 1.4

**2. ПОДГОТОВКА К ЗАНЯТИЮ**: По указанной литературе повторить тему «Трансформация, переходы и анимации», ответить на следующие вопросы:

2.1 Что такое CSS? Роль каскадных таблиц стилей при создании веб страниц.

2.2 Из каких частей состоит CSS - правило?

2.3 Какими способами осуществляется подключение CSS к HTML - документу?

**3. ЛИТЕРАТУРА**:

3.1 Шомас Е.А.Веб - дизайн. Учебное пособие. – Самара.: КС ПГУТИ, 2022. - 122c.

3.2 <http://uroki-css.ru/> - Уроки CSS3

**4. ПЕРЕЧЕНЬ ОБОРУДОВАНИЯ И ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**:

4.1 ПЭВМ, подключенные к сети Интернет

4.2 Браузеры

4.3 Текстовый редактор

**5. ЗАДАНИЕ**:

5.1 Применяя свойства трансформации, функцию **rotate**(), создать следующий эффект:



5.2 Применяя различные эффекты CSS3 выполните следующие переходы:

* Изменение цвета объекта (transition: background-color);
* Расширения и сужения объекта (transition: scale() );
* Вращение объекта (transition: rotate() );
* Сглаживания углов у объекта (transition: border-radius);
* Покачивание объекта (transition: translate(x, y)).



5.3 Создайте анимацию переключения светофора или индикатор прогрессии, который постепенно увеличивается.

**6. ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ:**

6.1 По предложенной литературе изучить необходимый материал и ответить на вопросы для допуска к практическому занятию;

6.2 Выполнить задание практического занятия;

6.3 Дать ответы на контрольные вопросы;

**7. СОДЕРЖАНИЕ ОТЧЕТА**:

7.1 Наименование и цель работы

7.2 Ответы на контрольные вопросы

7.3 Код программы и скрин ее выполнения

7.4 Вывод о проделанной работе

8.1 Что подразумевается под трансформацией объекта?

8.2 Какое свойство применяется для создания эффекта перехода в CSS3?

8.3 С помощью какого свойства определяется анимация в CSS3?

8.4 В чем отличие между переходом и анимацией?

**ПРИЛОЖЕНИЕ**:

Для создания **трансформации** в CSS применяется свойство **transform**.

Для **поворота** элемента свойство transform использует функцию **rotate**:

transform: rotate(угол\_поворота deg);

**Масштабирование:** transform:scale(величина\_масштабирования);

Для **перемещения** элемента используется функция **translate**:

transform: translate(offset\_X, offset\_Y);

Для **наклона** элемента применяется функция **skew().** transform: skew(X, Y);

**Переход** (**transition**) представляет анимацию от одного стиля к другому в течении определенного периода времени.

div{

                background-color: #ccc;

                transition-property: background-color;

                transition-duration: 2s;

            }

            div:hover{

                background-color: red;

            }

Чтобы указать свойство как анимируемое, его название передается свойству **transition-property**. Далее устанавливается время перехода в секундах (можно в миллисекундах – ms) с помощью свойства **transition-duration**.

Свойства **transition-timing-function** позволяет контролировать скорость хода и выполнения анимации. В качестве значения свойства может применяться одна из следующих функций:

**linear** – плавный переход, равномерный по времени;

**ease** – анимация ускоряется к середине и замедляется в конце;

**ease-in** – происходит ускорение в начале;

**ease-out** – происходит ускорение в конце;

**ease-in-out** - анимация ускоряется к середине и замедляется в конце;

**cubic-bezier**- к анимации применяется кубическая функция Безье. Для ее использования необходимо передать ряд значений.

Свойство transition можно написать сокращенно: transition: background-color 3s ease-in-out 500ms;

**Анимация** опирается на последовательную смену ключевых кадров (**keyframes**). Объявление ключевого кадра в CSS3 имеет следующую форму:

@keyframes название\_анимации {

    from {

        /\* начальные значения свойств CSS \*/   }

    to {

        /\* конечные значения свойств CSS \*/   }

}

**ПРАКТИЧЕСКОЕ ЗАНЯТИЕ № 13**

**НАИМЕНОВАНИЕ**: Применение модуля Flexbox

**1. ЦЕЛЬ**: Научиться применять элементы Flexbox на веб страницах. Формировать компетенции ОК 1 – ОК11; овладеть знаниями и умениями для освоения ПК 1.2, ПК 1.3, ПК 1.4

**2. ПОДГОТОВКА К ЗАНЯТИЮ**: по предложенной литературе повторить тему «CSS flexbox» и ответить на следующие вопросы:

2.1 Что такое Flexbox? Из чего состоит Flexbox?

2.2 Какие задачи позволяет решить Flexbox?

**3. ЛИТЕРАТУРА**:

3.1 Шомас Е.А.Веб - дизайн. Учебное пособие. – Самара.: КС ПГУТИ, 2022. - 122c.

3.2 <https://html5book.ru/css3-flexbox/> - Flexbox

**4. ПЕРЕЧЕНЬ ОБОРУДОВАНИЯ И ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**:

4.1 ПЭВМ, подключенные к сети Интернет

4.2 Браузеры

4.3 Текстовый редактор

**5. ЗАДАНИЕ**:

5.1 Создать фотогалерею из 10 изображений на flexbox.



5.2 Создать карточки (сетка с ценами) следующего образца:



**6. ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ:**

6.1 По предложенной литературе изучить необходимый материал и ответить на вопросы для допуска к практическому занятию;

6.2 Выполнить задание практического занятия;

6.3 Дать ответы на контрольные вопросы;

**7. СОДЕРЖАНИЕ ОТЧЕТА**:

7.1 Наименование и цель работы

7.2 Ответы на контрольные вопросы

7.3 Код программы и скрин ее выполнения

7.4 Вывод о проделанной работе

**8. КОНТРОЛЬНЫЕ ВОПРОСЫ:**

8.1 Какие задачи можно решить, используя модель flexbox при верстке веб - страниц?

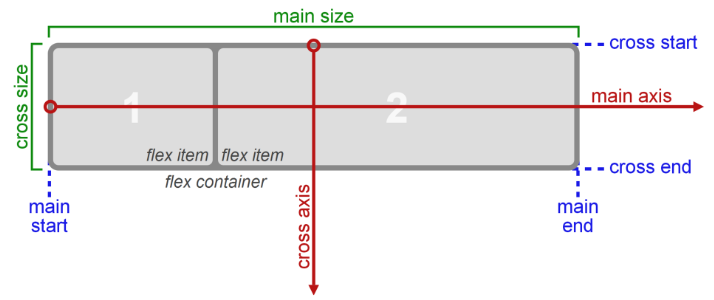
8.2 Что обозначает свойство со следующим значением **flex-flow: row nowrap**?

**ПРИЛОЖЕНИЕ**:

**CSS flexbox –** модуль макета гибкого контейнера – представляет собой способ компоновки элементов, в основе которой лежит идея оси.

Flexbox состоит из гибкого контейнера (**flex container**) и гибких элементов (**flax items**). Гибкие элементы могут выстраиваться в строку или столбец, а оставшееся между ними пространство распределяться различным образом.

**Модель flexbox** выглядит следующим образом:



**Главная ось (main axis)** – ось вдоль которой выкладываются flex – элементы.

**Main start** и **main end** – линии, которые определяют начальную и конечные стороны flex контейнера, относительно которых выкладываются flex – элементы.

**Основной размер** (**main size**) – ширина и высота flax – контейнера или элементов;

**Flex – контейнер**

Flex – контейнер устанавливает новый гибкий контекст форматирования для его содержимого. Он не является блочным контейнером, не поддерживает свойства float, clear, vertical-align.

Модель flexbox – разметки связана с определенным значение CSS – свойства **display** родительского html – элемента, содержащего внутри себя дочерние блоки. Для управления элементами с помощью этой модели нужно установить свойство **display** следующим образом:

.flex-container{

display: flex;

background-color: antiquewhite;

}



.flex-container{

display: inline-flex;

background-color: antiquewhite;

}



После установки данных значение свойства каждый дочерний элемент автоматически становится flex – элементом, выстраиваясь в один ряд.

**Flex – элементы**

Flex – элементы – блоки, представляющие содержимое flex – контейнера в потоке. Содержимое flex – контейнера можно разложить в любом направлении и в любом порядке.

**Свойство flex-direction**

Свойство относиться к flex – контейнеру. Управляет направлением главной оси, вдоль которой укладываются flex – элементы, в соответствии с текущим режимом записи. Значения свойства:

**flex-direction: row** // значение по умолчанию, слева направо

**flex-direction: row-revers** // справа налево

**flex-direction: column** // сверху вниз

**flex-direction: column** – **revers** // снизу вверх

**Свойство flex-wrap**

Свойство определяет, будет ли контейнер однострочным или многострочным, а также задает направление поперечной оси, определяющее направление укладки новой линии flex – контейнера. По умолчанию элементы укладываются в одну строку, вдоль главной оси. При переполнении они будут выходить за пределы рамки контейнера. Значения свойства:

**nowrap** – значение по умолчанию, элементы не переносятся, располагаются в одну линию

**wrap** – элементы переносятся, располагаясь в несколько горизонтальных рядов

**wrap – revers** – элементы переносятся на новые линии, располагаясь в обратном порядке

**initial** – устанавливает значение свойства по умолчанию

**inherit** – наследует значение от родительского элемента

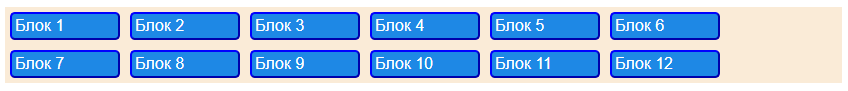
.flex-container{

display: flex;

flex-wrap: wrap;

background-color: antiquewhite;

}



**Свойство flex-flow**

Свойство определяет краткую запись направления и многострочности (заменяет свойства flex-direction и flex-wrap). Значение по умолчанию **flex-flow: row nowrap**. Перепишем предыдущий пример:

.flex-container{

display: inline-flex;

flex-flow: row wrap;

background-color: antiquewhite;

}

**Гибкость flex – элементов**

За гибкость макета отвечают свойства:

**flex-grow** – коэффициент растяжения

**flex-shrink** - коэффициент сужения

**flex-basis** – базовая ширина элемента

Все свойства можно задать через сокращенную запись: **flex: 0 1 auto;** (это значение по умолчанию).

.flex-container{

display: inline-flex;

flex-flow: row wrap;

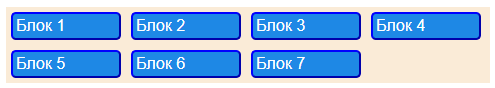
}

.flex-item{

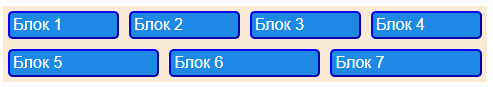
…

flex: 0 1 100px;

}



Если **flex: 1 1 100px;**



**ПРАКТИЧЕСКОЕ ЗАНЯТИЕ № 14**

**НАИМЕНОВАНИЕ**: Создание многоколоночного макета

**1. ЦЕЛЬ**: Научиться верстать многоколоночные макеты. Формировать компетенции ОК 1 – ОК11; овладеть знаниями и умениями для освоения ПК 1.2, ПК 1.3, ПК 1.4

**2. ПОДГОТОВКА К ЗАНЯТИЮ**: По указанной литературе повторить тему «CSS columns», ответить на следующие вопросы:

2.1 Где можно увидеть многоколоночную верстку?

2.2 Что можно помещать в колонки?

**3. ЛИТЕРАТУРА**:

* 1. Шомас Е.А. Веб – дизайн, учебное пособие. Самара, КС ПГУТИ, 2022.
  2. <https://html5book.ru/css3-columns/> - Уроки CSS3

**4. ПЕРЕЧЕНЬ ОБОРУДОВАНИЯ И ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**:

4.1 ПЭВМ, подключенные к сети Интернет

4.2 Текстовый редактор, браузер

**5. ЗАДАНИЕ**:

5.1 Сверстать страницу по следующему образцу:



1. **ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ:**

6.1 По предложенной литературе изучить необходимый материал и ответить на вопросы для допуска к практическому занятию;

6.2 Выполнить задание практического занятия;

6.3 Дать ответы на контрольные вопросы.

**7. СОДЕРЖАНИЕ ОТЧЕТА**:

7.1 Наименование и цель работы;

* 1. Код – программы и скрин выполненного задания;

7.3 Ответы на контрольные вопросы;

7.4 Вывод о проделанной работе;

**8. КОНТРОЛЬНЫЕ ВОПРОСЫ:**

8.1 Какое свойство используется для определения количества колонок?

8.2 Какое свойство применяется для определения ширины колонок?

8.3 Какие свойства можно применить к разделительной линии между колонками?

**ПРИЛОЖЕНИЕ**:

Модель CSS **columns** описывает многоколоночный макет, который позволяет организовывать содержимое таким образом, чтобы оно занимало несколько вертикальных контейнеров, подобно газете или журналу.

Колонки могут содержать заголовки, текст, таблицы, картинки и любые другие **inline** – элементы.

**Количество и ширина колонок**

Свойства **column-count** и **column-width** используется для установки количества и ширины колонок. Свойство **columns**, является сокращенным свойством, посредством которого можно одновременно задать количество и ширину колонок.

**Ширина колонок (column-width)** указывает минимальную ширину, которую должен занимать каждый столбец. Значения свойства:

**длина** – задается в **px**, **em**. Проценты не допускаются, а также и отрицательные значения;

**auto** – значение по умолчанию, зависит и от других свойств, например количества колонок;

**initial** – устанавливает значение свойства в значение по умолчанию;

**inherit** – наследуется от родительского элемента.

Синтаксис:

column-width: auto;

column-width: 100px;

column-width: 10em;

**Количество колонок (column-count)** описывает количество колонок, а их ширина будет рассчитываться, исходя из ширины доступного пространства. Свойство не наследуется. Может принимать следующие значения:

**число** – задается целым числом, которое должно быть больше 0;

**auto** – значение по умолчанию, зависит и от других свойств, например ширины колонок;

**initial** – устанавливает значение свойства в значение по умолчанию;

**inherit** – наследуется от родительского элемента.

column-count: auto;

column-count: 2;

column-count: inherit;

column-count: initial;

Можно задать оба свойства одновременно, например,

columns: auto 12em; /\* column-width: 12em; column-count: auto \*/

columns: 2 auto; /\* column-width: auto; column-count: 2 \*/

**Промежутки между колонками и разделительные линии**

Промежутки между колонками и разделительные линии помещаются между колонками в одном многоколоночном контейнере. Длина промежутков и разделительных линий равна высоте колонки. Промежутки в колонках занимают место, т.е. раздвигают содержимое в соседних колонках.

Разделительные линии рисуются в середине промежутка между колонками, не занимая места.

За промежутки между колонками отвечает свойство **column-gap**. Значение свойства задается следующими значениями:

**длина** – промежуток задается в единицах длины. Значение не отрицательное;

**normal** – значение по умолчанию, соответствует значению 1em;

**initial** – устанавливает значение свойства в значение по умолчанию;

**inherit** – наследуется от родительского элемента.

**Разделительная линия** определяется с помощью свойства **column-rule**.

Можно определить **цвет** разделительной линии - **column**-**rule-color**.

Можно задать **стиль** разделительной линии - **column-rule-style**. В этом случае принимаются следующие значения (перечислены не все):

**none** – значение по умолчанию, равно 0;

**hidden** – линия отсутствует (скрыта);

**dotted** – квадратные точки;

**dashed** – тире;

**solid**- сплошная линия;

**double** – двойная линия;

**groove**- объемная линия.

**Ширина** разделительной линии: **column-rule-width**. Значения:

**длина** – задается в единицах длины;

**thin** – тонкая линия;

**medium** – средняя толщина (по умолчанию);

**thick** – утолченная линия.

Можно все свойства разделительной линии записать кратко:

**column-rule: solid thick blue**;

**Разрыв колонок**

Когда содержимое размещено в нескольких колонках, браузер должен определить, где размещаются разрывы колонок. Для решения данной задачи введено три свойства: **break**-**before, break- after** и **break-inside**.

**Охват колонок (свойство column-span)**

Свойство позволяет элементу охватить несколько столбцов. Указывается не для блока – контейнера, а для конкретного элемента внутри, например, заголовка. Принимает значения: **none** и **all**.

**Заполнение колонок содержимым (свойство column – fill)**

Свойство **column – fill** контролирует заполнение колонок содержимым. Различают две стратегии заполнения колонок: колонки могут быть выравнены по высоте или нет. Если они выравнены, браузеры пытаются минимизировать изменения высоты колонки, учитывая при этом вынужденные разрывы. Если не выравнены, они заполняются последовательно, некоторые из них могут быть частично заполнены или вообще не заполнены. Значения:

**auto** - колонки заполняются последовательно;

**balance** – отображает содержимое одинаково во всех колонках (значение по умолчанию);

**balance-all** – равномерное выравнивание содержимого.

**Переполнение внутри многоколоночного контейнера** может привести к разрыву колонки, содержимое, которое выходит за границы колонки, выходит за ее границы и не обрезается. Это в первую очередь касается изображений. Для решения этой задачи, для изображений следует прописать свойства:

img {

display: block; /\*убираем нижний отступ под картинкой\*/

width: 100%; /\*растягиваем изображение на всю ширину блока-контейнера\*/

}

**Разбивка на страницы и переполнение вне многоколоночного контейнера**

Содержимое и разделительные линии, которые выходят за рамки колонок по краям, обрезаются в соответствии со свойством **overflow**.

Многоколоночный контейнер может иметь больше колонок, чем у него есть для этого места из-за ограничения высоты колонок (например, с помощью **height** или **max-height**) и явных разрывов колонок. В этом случае дополнительные колонки создаются в направлении строки, перемещаясь на следующие страницы.

**ПРАКТИЧЕСКОЕ ЗАНЯТИЕ № 15**

**НАИМЕНОВАНИЕ**: Использование Grid - сетки

**1. ЦЕЛЬ**: Научиться выполнять разметку страницы средствами сетки Grid. Формировать компетенции ОК 1 – ОК11; овладеть знаниями и умениями для освоения ПК 1.2, ПК 1.3, ПК 1.4

**2. ПОДГОТОВКА К ЗАНЯТИЮ**: По указанной литературе повторить тему «CSS Grid», ответить на следующие вопросы:

2.1 Где можно использовать модуль Grid?

2.2 Из каких компонентов состоит модуль Grid?

**3. ЛИТЕРАТУРА**:

* 1. Шомас Е.А. Веб – дизайн, учебное пособие. Самара, КС ПГУТИ, 2022.
  2. <https://html5book.ru/css3-columns/> - Уроки CSS3

**4. ПЕРЕЧЕНЬ ОБОРУДОВАНИЯ И ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**:

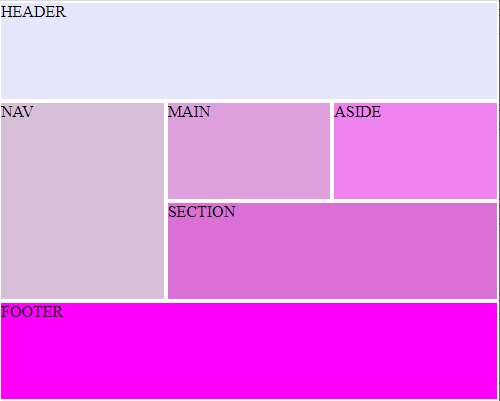
4.1 ПЭВМ, подключенные к сети Интернет

4.2 Текстовый редактор, браузер

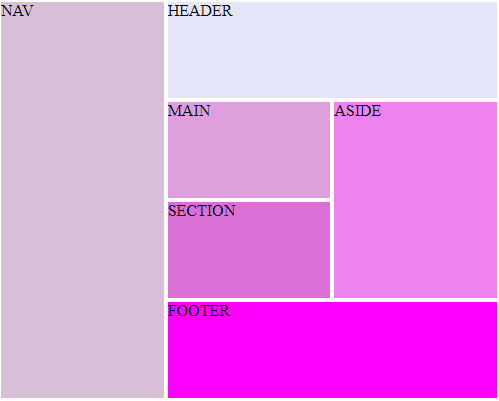
**5. ЗАДАНИЕ**:

5.1 Сверстать страницу по следующему образцу:

***Вариант 1***



***Вариант 2***



**6. ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ:**

6.1 По предложенной литературе изучить необходимый материал и ответить на вопросы для допуска к практическому занятию;

6.2 Выполнить задание практического занятия;

6.3 Дать ответы на контрольные вопросы.

**7. СОДЕРЖАНИЕ ОТЧЕТА**:

* 1. Наименование и цель работы;

7.2 Код – программы и скрин выполненного задания;

7.3 Ответы на контрольные вопросы;

7.4 Вывод о проделанной работе;

**8. КОНТРОЛЬНЫЕ ВОПРОСЫ:**

8.1 Какие свойства позволяют разбить сетку на строки и столбцы?

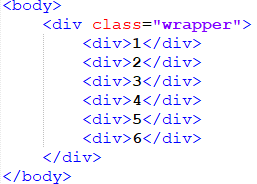
8.2 Как объединяют строки и столбцы внутри сетки?

**ПРИЛОЖЕНИЕ**:

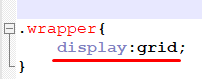
**Модуль Grid** – мощный и простой инструмент для создания макета веб – сайта.

Двумя основными компонентами CSS Grid являются ***обертка*** (Родитель) и ***элементы*** (Дети). Обертка – это фактически сетка, а элементы – это содержимое внутри сетки.

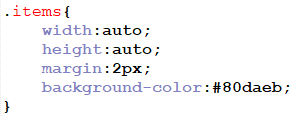
Разметка для обертки с шестью элементами:



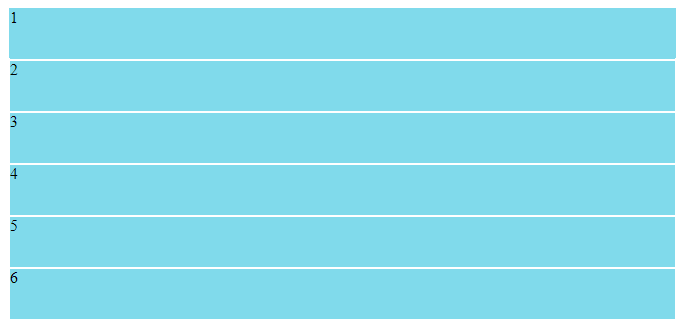
Чтобы превратить обертку div в сетку, дадим ей отображение grid:



Добавим для вложенных div немного стилей, не связанных с Grid.

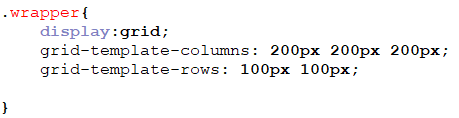


В настоящий момент сетка ничего не делает, она только складывает 6 div друг в друга.

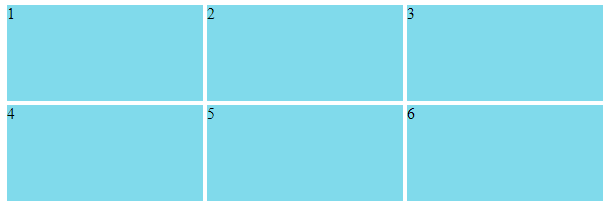


**Столбцы и строки**

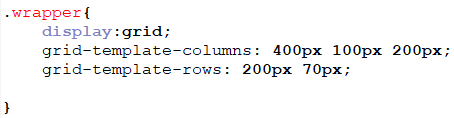
Чтобы сделать макет двумерным, нужно определить столбцы и строки. Создадим три столбца и две строки. Будем использовать свойства grid-template-rows и grid-template-columns.

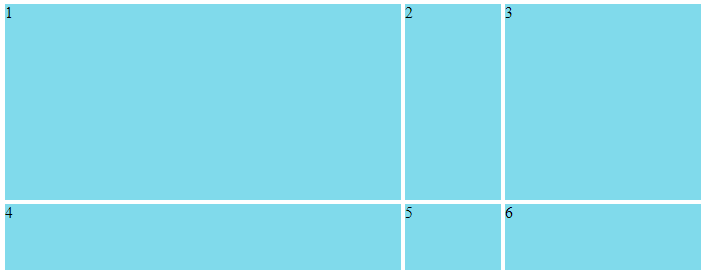


Так как мы написали три значения для свойства grid-template-columns, то мы получим три столбца. Для свойства grid-template-rows указали два значения, значит будет две строки. Значения определяют ширину столбцов (200px) и высоту строк (100px). Результат:



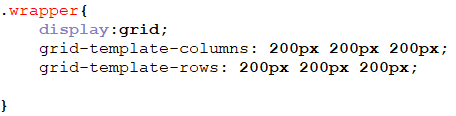
Переопределим размеры столбцов и строк, чтобы уловить взаимосвязь:



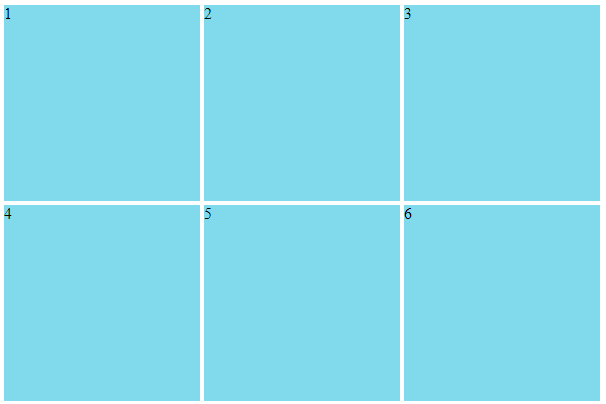


**Расположение элементов**

Создадим сетку 3х3, используя нашу разметку:

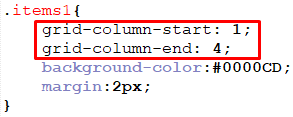


Результат:

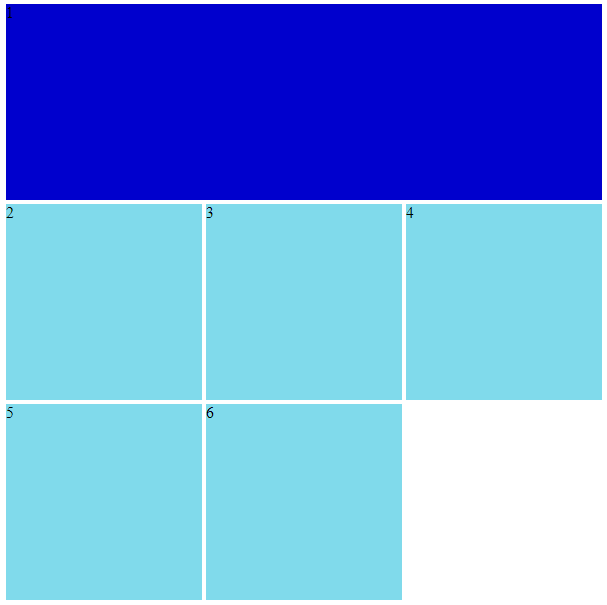


В окне браузера мы видим только сетку 3х2, в то время как мы определили ее как 3х3. У нас есть только 6 элементов, чтобы заполнить эту сетку. Если бы у нас было еще три элемента, то и нижняя строка была бы заполнена.

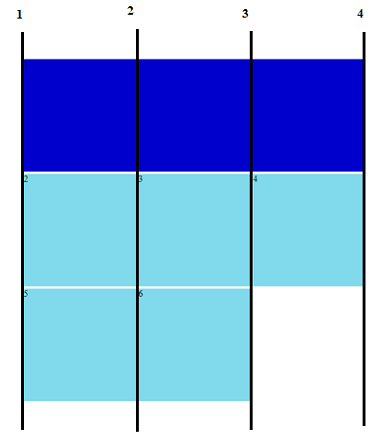
Для позиционирования и изменения размера элементов мы будем указывать их и использовать свойства grid-column и grid-row:



В свойстве мы указали, что items1 начинается у нас с первой линии сетки и заканчивается на четвертом столбце. Другими словами, займет всю строку.

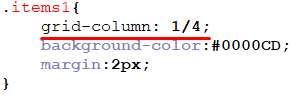


Почему оказалось 4 столбца? На рисунке проведем линии столбцов (линии строк проводятся аналогично):

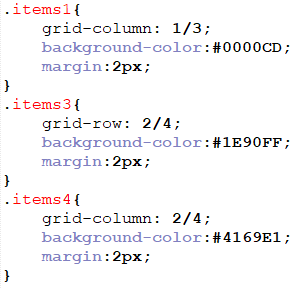


Теперь мы используем все строки в сетке. Когда мы сделали так, чтобы первый элемент занимал весь первый ряд, он сдвинул все остальные элементы вниз.

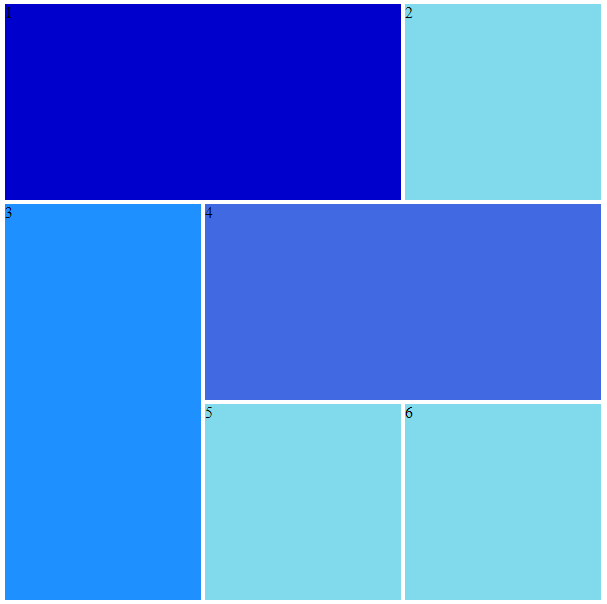
Более простой синтаксис для примера выше:



Перестроим элементы следующим образом:



Получим следующий результат:



**ПРАКТИЧЕСКОЕ ЗАНЯТИЕ № 16**

**НАИМЕНОВАНИЕ**: Адаптивность страницы под различные разрешения экранов

**1. ЦЕЛЬ**: Научиться верстать адаптивные веб – страницы с применением media - правил. Формировать компетенции ОК 1 – ОК11; овладеть знаниями и умениями для освоения ПК 1.2, ПК 1.3, ПК 1.4

**2. ПОДГОТОВКА К ЗАНЯТИЮ**: По указанной литературе повторить тему «Введение в адаптивный дизайн», ответить на следующие вопросы:

2.1 Что значит адаптивный веб - сайт?

2.2 За счет чего может достигаться адаптивность?

**3. ЛИТЕРАТУРА**:

* 1. Шомас Е.А. Веб – дизайн, учебное пособие. Самара, КС ПГУТИ, 2022
  2. https://html5book.ru/css3 - Уроки CSS3

**4. ПЕРЕЧЕНЬ ОБОРУДОВАНИЯ И ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**:

4.1 ПЭВМ, подключенные к сети Интернет

4.2 Текстовый редактор, браузер

**5. ЗАДАНИЕ**:

5.1 Создать адаптивную веб-страницу, которая позволяет на экране ПК просматривать три блока с изображением и текстом. При изменении расширения экрана до 768 и меньше, блоки смещаются вниз (друг под друга).



5.2 Для мобильной версии изменитт цвет фона документа и скрыть рисунок

1. **ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ:**

6.1 По предложенной литературе изучить необходимый материал и ответить на вопросы для допуска к практическому занятию;

6.2 Выполнить задание практического занятия;

6.3 Дать ответы на контрольные вопросы.

**7. СОДЕРЖАНИЕ ОТЧЕТА**:

7.1 Наименование и цель работы;

7.2 Код – программы и скрин выполненного задания;

* 1. Ответы на контрольные вопросы;

7.4 Вывод о проделанной работе;

**8. КОНТРОЛЬНЫЕ ВОПРОСЫ:**

8.1 Как тестируются адаптивные сайты для различных расширений экрана на ПК?

8.2 С какого тега начинается адаптивность любого сайта?

**ПРИЛОЖЕНИЕ**:

Адаптивный веб-дизайн позволяет хорошо выглядеть веб - странице на всех устройствах (компьютерах, планшетах и телефонах).

**Адаптивность достигается:**

1. Удалением (скрытием) блоков и/или перераспределением их между собой:
2. Изменение размеров блоков в зависимости от ширины экрана.

**Тестирование адаптивного сайта**

Современные браузеры позволяют эмулировать запуск веб – страницы на том или ином устройстве с различной шириной экрана. Например, в Google Chrome надо перейти в меню **Дополнительные инструменты – Инструменты разработчика (F12)**. После открытия панели разработчика в начале меню самой панели нажать на иконку мобильного телефона, и после этого можно будет эмулировать отображение страницы на различных устройствах.

**Мета-тег viewport**

Адаптивный дизайн начинается с мета-тега **viewport**, который размещается в разделе **head**, например:

**<meta name="viewport" content="width=device-width, initial-scale=1.0">**

Вся видимая часть на экране браузера описывается понятием **viewport**, по сути - это область в которую веб – браузер пытается «впихнуть» веб – страницу. По умолчанию браузер считает, что каждая веб – страница предназначена для стандартных экранов ПК. При этом для каждого браузера устанавливается своя ширина области viewport. Если устройство имеет меньшую ширину экрана, то страница сжимается на коэффициент масштабирования. Чтобы избежать подобной картины следует использовать мета-тег **viewport**.

**Правила Media Query**

Важным элементом в построении адаптивного дизайна являются правила **Media Query**, которые позволяют определить стиль в зависимости от размера браузера пользователя.

<head>

<title>Адаптивная веб-страница</title>

 <meta name="viewport" content="width=device-width">

**<link rel="stylesheet" type="text/css" href="style.css" />**

 </head>

Используя правила **CSS3 Media Query** в одном файле css можно совместить стили для различных устройств:

body {

    background - color: red;}

/\*Далее остальные стили\*/

**@media (max-device-width:480px){**

    body {

        background -color: blue;

    }

    /\*Далее остальные стили\*/}

**ПРАКТИЧЕСКОЕ ЗАНЯТИЕ № 17**

**НАИМЕНОВАНИЕ**: Внедрение компонентов Bootstrap на веб - страницу

**1. ЦЕЛЬ**: Научиться использовать сетку Bootstrap для создания шаблона сайта, внедрять компоненты Bootstrap на созданную страницу. Формировать компетенции ОК 1 – ОК11; овладеть знаниями и умениями для освоения ПК 1.2, ПК 1.3, ПК 1.4

**2. ПОДГОТОВКА К ЗАНЯТИЮ**: по предложенной литературе повторить тему «Фреймворк Bootstrap» и ответить на следующие вопросы:

2.1 Назначение фреймворка?

2.2 Какие компоненты представлены во фреймворке?

**3. ЛИТЕРАТУРА**:

3.1 Шомас Е.А.Веб - дизайн. Учебное пособие. – Самара.: КС ПГУТИ, 2022. - 122c.

3.2 <http://getbootstrap.com> – фреймворк Bootstrap

**4. ПЕРЕЧЕНЬ ОБОРУДОВАНИЯ И ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**:

4.1 ПЭВМ, подключенные к сети Интернет

4.2 Браузеры

4.3 Текстовый редактор

**5. ЗАДАНИЕ**:

5.1 Создать собственный профиль по аналогии социальных сетей. Пример профиля должен выглядеть следующим образом:



**6. ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ:**

6.1 По предложенной литературе изучить необходимый материал и ответить на вопросы для допуска к практическому занятию;

6.2 Выполнить задание практического занятия;

6.3 Дать ответы на контрольные вопросы;

**7. СОДЕРЖАНИЕ ОТЧЕТА**:

7.1 Наименование и цель работы

7.2 Ответы на контрольные вопросы

7.3 Код программы и скрин ее выполнения

7.4 Вывод о проделанной работе

**8. Контрольные вопросы:**

8.1 Сколько колонок идет в стандартной сетке Bootstrap?

8.2 Как сверстать фиксированный макет на основе сетки Bootstrap?

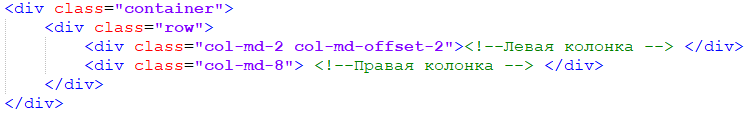
8.3 Как сверстать резиновый макет на основе сетки Bootstrap?

8.4 Какие стандартные классы для разметки макеты представлены в Bootstrap?

**ПРИЛОЖЕНИЕ:**

**Ход выполнения задания:**

1. Для выполнения задания нужно взять шаблон, к которому подключен Bootstrap. Можно взять стартовый шаблон (**Starter template**) на сайте <https://getbootstrap.com/docs/5.0/getting-started/introduction/>
2. Основа профиля (его сетка) состоит из двух колонок:



1. Левая колонка включает в себя следующие элементы:

Фотография – необходимо разместить изображение шириной 200px, высота произвольная. Пример:

<img src="" class="img-thumbnail" alt="" width="200" height="250">

Файл с изображением должен лежать в той же папке, что и все остальные файлы. Изображение нужно поместить в левую колонку с применением компонента **Image thumbnails**:

***Компонент***: <https://getbootstrap.com/docs/5.0/content/images/#image-thumbnails>

Пример:



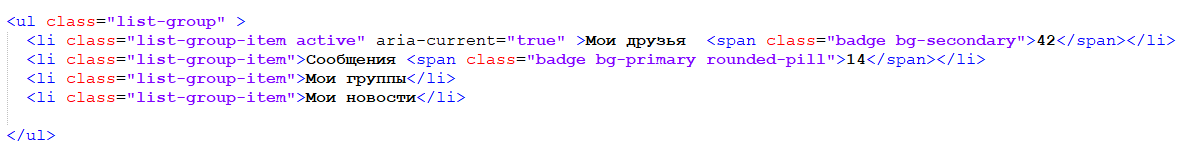
1. Список ссылок реализован с помощью компонента **Active items**:

***Компонент***: <https://getbootstrap.com/docs/5.0/components/list-group/>

Внутри списка используется компонент **badge**:

***Компонент***: <https://getbootstrap.com/docs/5.0/components/badge/>

Пример:



1. Правая колонка состоит из следующих компонентов:

**Заголовок:** <h1> Иванов Иван <small class="text-muted">Самара</small></h1>

Тег <small class="text-muted"> размещенный внутри заголовка, делает текст размером меньше и более блеклого оттенка.

***Компонент***: <https://bootstrap-4.ru/docs/5.1/content/typography/?#headings>

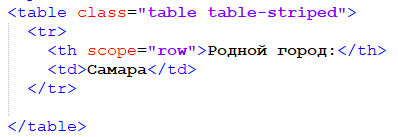
**Лид:** выделяющийся, отстоящий от основной массы параграф (абзац), формируется добавлением класса .**lead**

***Компонент***: <https://bootstrap-4.ru/docs/5.1/content/typography/?#headings>

1. Список характеристик пользователя, реализовывается с помощью таблиц:

***Компонент***: <https://bootstrap-4.ru/docs/5.1/content/tables/>

Пример таблицы с одной строкой:



**ПРАКТИЧЕСКОЕ ЗАНЯТИЕ №18**

**НАИМЕНОВАНИЕ**: Разработка шаблона для будущего сайта

**1 ЦЕЛЬ**: Разработать шаблон сайта для интернета - магазина в соответствии с предоставленными прототипами страниц. Формировать компетенции ОК 1 – ОК11; овладеть знаниями и умениями для освоения ПК 1.2, ПК 1.3, ПК 1.4

**2 ПОДГОТОВКА К ЗАНЯТИЮ**: По предложенной литературе повторить материал на тему «Разработка адаптивных веб – сайтов» и ответить на следующие вопросы:

2.1 Какие программные средства используются при подготовке шаблона сайта?

2.2 Что означает слово «адаптивность». Как добиться адаптивности сайта?

**3 ЛИТЕРАТУРА:**

* 1. Шомас Е.А.Веб - дизайн. Учебное пособие. – Самара.: КС ПГУТИ, 2022. - 122c.

3.2 Материал об отзывчивом и адаптивном дизайне сайта [Электронный ресурс]. Режим доступа - <https://html5book.ru/otzyvchivyj-dizayn-saita/> , свободный

**4. ПЕРЕЧЕНЬ ОБОРУДОВАНИЯ И ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ:**

4.1 ПЭВМ, подключенные к сети Интернет

4.2 Браузеры

* 1. Графический редактор Photoshop

**5. ЗАДАНИЕ:**

* 1. Сверстать веб – приложение (две основные страницы) по продаже продуктов питания и напитков, например, пицца, роллы и напитки или чай, кофе и сопутствующие товары) по предоставленным шаблонам;
  2. В дизайне должен использоваться принцип единообразия в структуре страниц: единый размер элементов, высота кнопок, оформление заголовков, основного текста, ссылок и т.д.;
  3. Страницы сайта должны быть сверстаны с учетом просмотра их на различных устройствах с различными разрешениями (360 х 640, 768 х 1024, 1200 и более);

1. **ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ:**

6.1 По предложенной литературе изучить необходимый материал и ответить на вопросы для допуска к практическому занятию;

6.2 Выполнить задание практического занятия;

6.3 Дать ответы на контрольные вопросы;

1. **СОДЕРЖАНИЕ ОТЧЕТА:**

7.1 Наименование и цель работы;

7.2 Ответы на контрольные вопросы;

7.3 Выводы по работе (сравнить два метода верстки);

7.4 Прикреплённые к отчету архивные файлы с выполненным заданием.

**8 КОНТРОЛЬНЫЕ ВОПРОСЫ:**

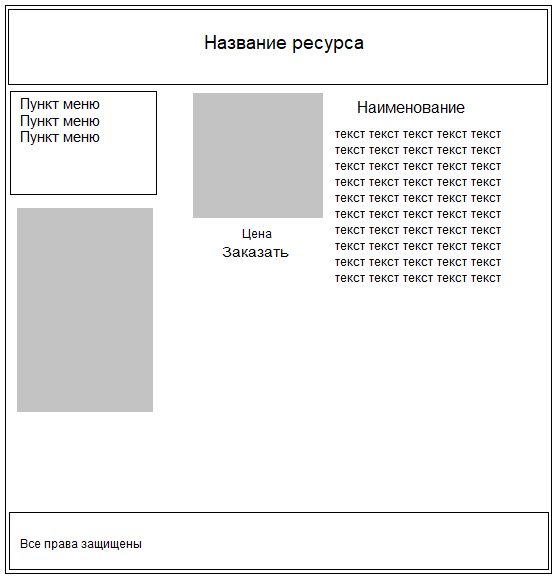
8.1 Каким образом можно сделать сайт «отзывчивым»?

8.2 Для чего создается шаблон (макет) будущего сайта?

**ПРИЛОЖЕНИЕ**:

Прототипы главной и дочерней страниц:



****

**ПРАКТИЧЕСКОЕ ЗАНЯТИЕ №19**

**НАИМЕНОВАНИЕ**: Верстка сайта по шаблону

**1 ЦЕЛЬ**: Разработать шаблон сайта для интернета - магазина в соответствии с предоставленными прототипами страниц. Формировать компетенции ОК 1 – ОК11; овладеть знаниями и умениями для освоения ПК 1.2, ПК 1.3, ПК 1.4

**2 ПОДГОТОВКА К ЗАНЯТИЮ**: По предложенной литературе повторить материал на тему «Разработка адаптивных веб – сайтов» и ответить на следующие вопросы:

2.1 Какие веб - технологии используются при верстке сайта по готовому шаблону?

2.2 С чего начинается верстка сайта?

**3 ЛИТЕРАТУРА:**

* 1. Шомас Е.А.Веб - дизайн. Учебное пособие. – Самара.: КС ПГУТИ, 2022. - 122c.

3.2 Официальный сайт с документацией на русском языке Bootstrap [Электронный ресурс]. Режим доступа - <https://getbootstrap.ru/docs/v4-alpha/> , свободный

**4. ПЕРЕЧЕНЬ ОБОРУДОВАНИЯ И ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ:**

4.1 ПЭВМ, подключенные к сети Интернет

4.2 Браузеры

4.3 Текстовый редактор

**5. ЗАДАНИЕ:**

5.1 Требуется заполнить приложение контентом в соответствии с подготовленным шаблоном (2 страницы).

5.2 Страницы сайта должны быть сверстаны с учетом просмотра их на различных устройствах с различными разрешениями (360 х 640, 768 х 1024, 1200 и более);

* 1. Верстку осуществить одним из двух способов: с помощью медиа – запросов и с использованием фреймворка, например, Bootstrap.

**6. ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ:**

6.1 По предложенной литературе изучить необходимый материал и ответить на вопросы для допуска к практическому занятию;

6.2 Выполнить задание практического занятия;

6.3 Дать ответы на контрольные вопросы;

1. **СОДЕРЖАНИЕ ОТЧЕТА:**

7.1 Наименование и цель работы;

7.2 Ответы на контрольные вопросы;

7.3 Выводы по работе (сравнить два метода верстки);

7.4 Прикреплённые к отчету архивные файлы с выполненным заданием.

**8 КОНТРОЛЬНЫЕ ВОПРОСЫ:**

8.1 Что такое медиа – запросы? Синтаксис медиа – запроса.

8.2 Что такое фреймворк? Его назначение при верстке веб – страниц?

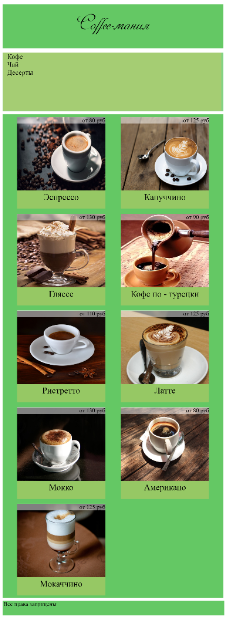
**ПРИЛОЖЕНИЕ**:

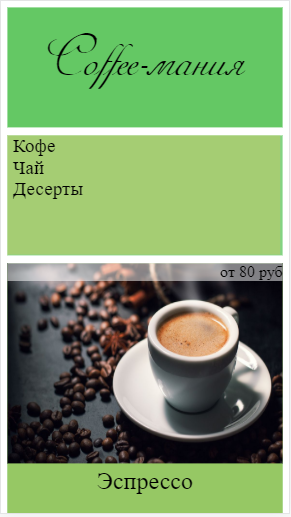
Пример заполненных страниц для версии просмотра на ПК:





Пример заполненных страниц для версии просмотра на планшете (смартфоне):





**ПРАКТИЧЕСКОЕ ЗАНЯТИЕ №20**

**НАИМЕНОВАНИЕ**: Размещение сайта в интернете на бесплатном хостинге

**1 ЦЕЛЬ**: Разместить сверстанный сайт в сети интернет. Формировать компетенции ОК 1 – ОК11; овладеть знаниями и умениями для освоения ПК 1.2, ПК 1.3, ПК 1.4

**2 ПОДГОТОВКА К ЗАНЯТИЮ**: По предложенной литературе повторить материал на тему «Размещение сайта на бесплатном хостинге» и ответить на следующие вопросы:

2.1 Что такое хостинг?

1. **ЛИТЕРАТУРА:**

3.1 Шомас Е.А.Веб - дизайн. Учебное пособие. – Самара.: КС ПГУТИ, 2022. - 122c.

3.2 Официальный сайт с документацией на русском языке Bootstrap [Электронный ресурс]. Режим доступа - <https://getbootstrap.ru/docs/v4-alpha/> , свободный

**4. ПЕРЕЧЕНЬ ОБОРУДОВАНИЯ И ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ:**

4.1 ПЭВМ, подключенные к сети Интернет

4.2 Браузеры

4.3 Текстовый редактор

**5. ЗАДАНИЕ:**

5.1 Разместите сайт на любом бесплатном хостинге. Пример размещения на сервисе GitHub показан в приложении

**6. ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ:**

6.1 По предложенной литературе изучить необходимый материал и ответить на вопросы для допуска к практическому занятию;

6.2 Выполнить задание практического занятия;

* 1. Дать ответы на контрольные вопросы;

**7. СОДЕРЖАНИЕ ОТЧЕТА:**

7.1 Наименование и цель работы;

7.2 Ответы на контрольные вопросы;

7.3 Выводы по работе (сравнить два метода верстки);

7.4 Ссылка на веб - ресурс

1. **КОНТРОЛЬНЫЕ ВОПРОСЫ:**

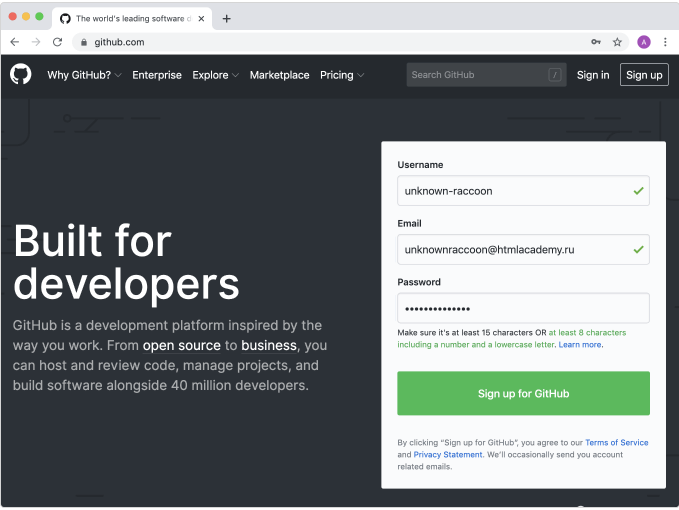
8.1 Опишите кратко алгоритм размещения сайта на хостинге

**ПРИЛОЖЕНИЕ**:

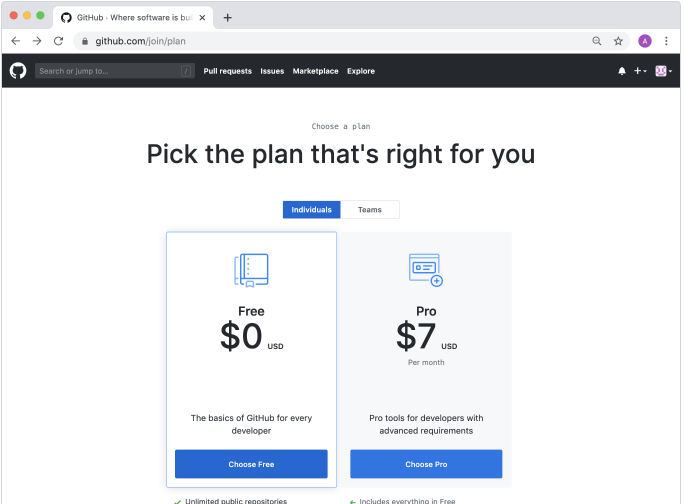
**Хостинг** – это сервер (или часть его), на котором можно разместить файлы своего веб – ресурса или приложения.Сервер работает без перерыва, чтобы обеспечить пользователям круглосуточный доступ к веб – ресурсам.

Для примера разберем публикацию сайта на платформе GitHub, она служит для разработчиков и позволяет бесплатно опубликовать простые сайты.

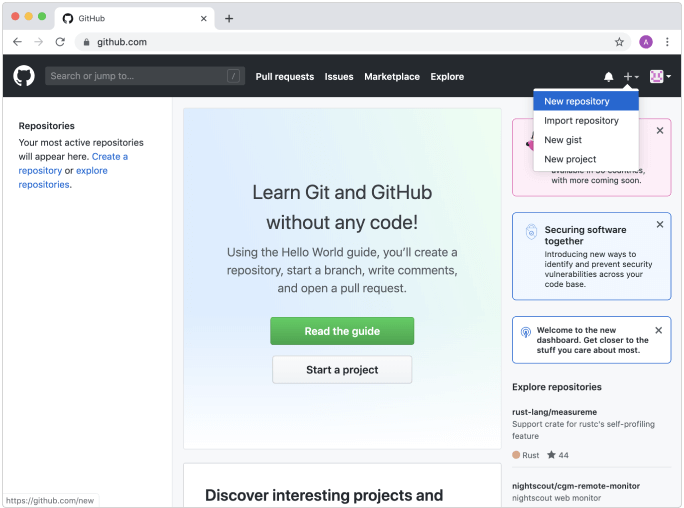
Для начала проходят процедуру регистрации на официальном сайте GitHub. На главной странице в специальной форме вводится логин, электронная почта и пароль для создания учетной записи.



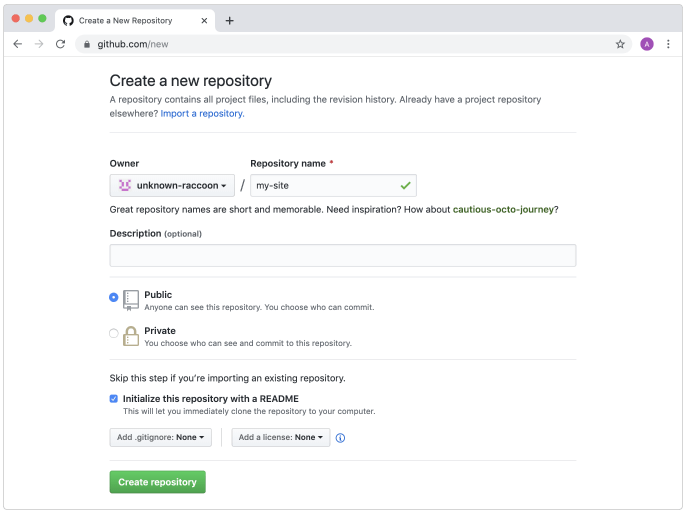
Далее подтверждаем свой аккаунт и выбираем тарифный план (бесплатный) и подтверждаем электронную почту.



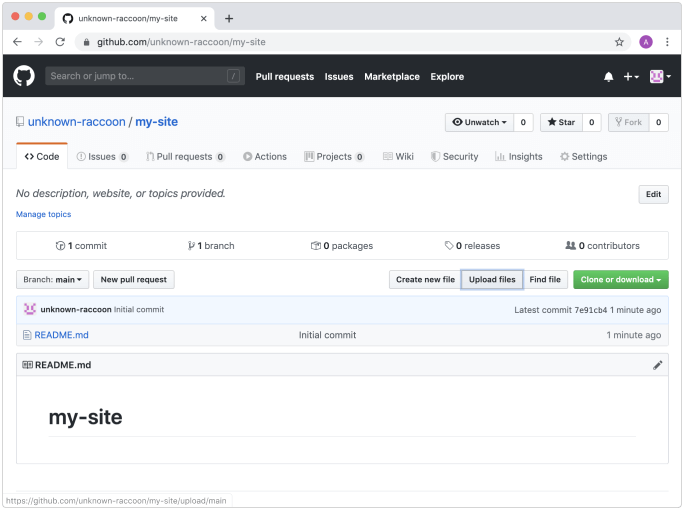
Далее создается хранилище, где будет лежать код сайта. Нажимаем кнопку **New repository**.



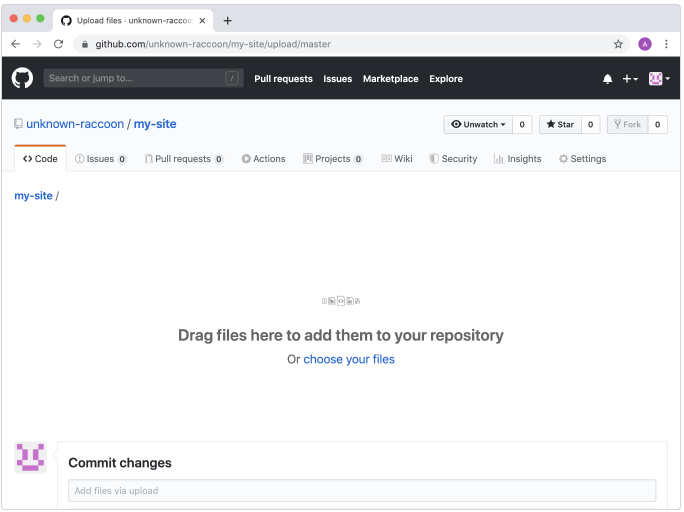
Выбираем название для хранилища, например, **my-site**, и указываем тип репозитория (например, публичный). Обязательно устанавливаем галочку **initialize this repository with a README**, а после нажимаем кнопку **Create repository**.



Репозиторий создан, теперь загружаем файлы с кодом, для этого воспользуемся интерфейсом Гитхаба. Нажимаем кнопку **Unload files**.

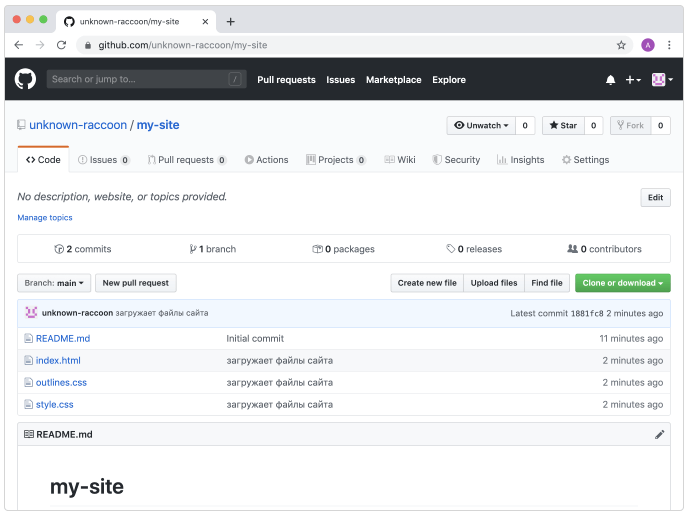


Перетаскиваем файлы в появившееся поле для загрузки.

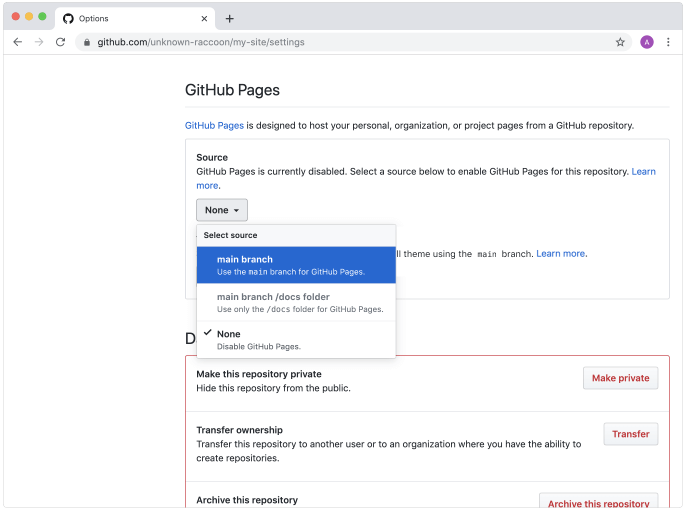


По правилам, каждое изменение в репозитории должно сопровождаться кратким описанием, что делает это изменение, например, «загружает файлы сайта». Добавляем описание и нажимаем кнопку **Commit changes**.

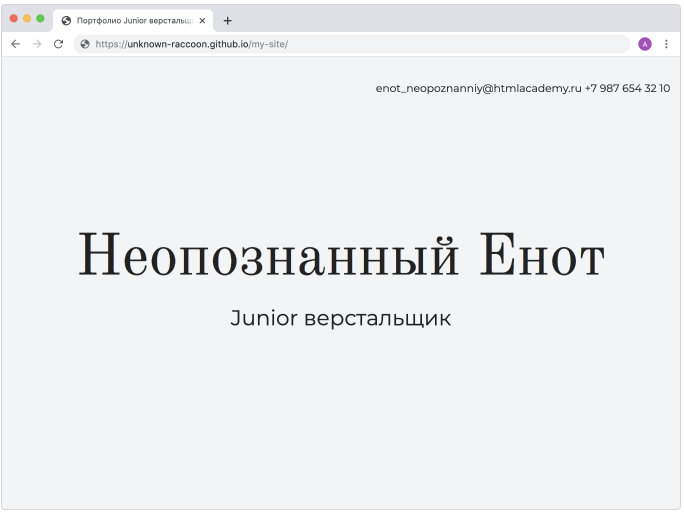
После того, как сайт будет залит на Гитхаб появится следующее окно



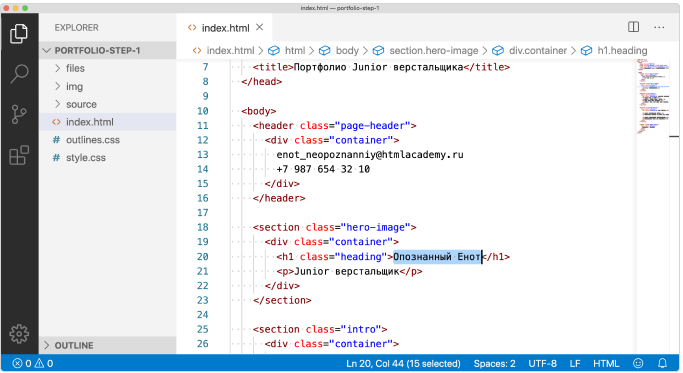
Производим настройку, чтобы сайт можно было открыть в браузере. Для этого переходим в настройки **Settings**, пролистываем настройки вниз до раздела **GitHub Pages** и в выпадающем списке выбираем ветку **main**. Изменения сохраняются автоматически.



Пробуем открыть сайт в браузере. Он будет доступен по адресу **[ваш логин].github.io/[название репозитория]**.



Если будет необходимость в доработке сайта и публикации изменений, то действуют следующим образом. Сначала вносятся изменения, допустим на гласной странице.



Далее нужно заново загрузить эту страницу на Гитхаб, добавив описание, что делается изменение, например, «меняем текст на главной странице». Снова нажимаем кнопку **Commit changes**. Через несколько секунд изменения появятся на сайте.