

Лабораторная работа #03

3.1 Подключение CSS

Описание

[Как добавлять стили на страницу](#)

3.2 Селекторы. Вес селектора

Описание

[Про селекторы](#)

[Вес селектора](#)

3.3 Единицы измерения CSS

Описание

[W3.Org — единицы измерения CSS](#)

[Webref — Единицы измерения CSS](#)

3.4 Свойство display

Описание

Свойство display:

<https://webref.ru/css/display>

https://www.w3schools.com/cssref/pr_class_display.asp

<https://html5book.ru/svoystvo-display/>

3.5 Блочная модель

Описание

Блочная модель:

<https://webref.ru/layout/learn-html-css/box-model>

https://developer.mozilla.org/ru/docs/Web/CSS/box_model

<https://html5book.ru/css-blochnaya-model/>

3.6 Позиционирование

Описание

[Webref — Позиционирование](#)

[HTML5book — CSS-позиционирование](#)

3.7 Кодстайл CSS

Описание

Про плагин stylelint и работу с ним расскажем чуть позже, когда изучим основы html, css, js.

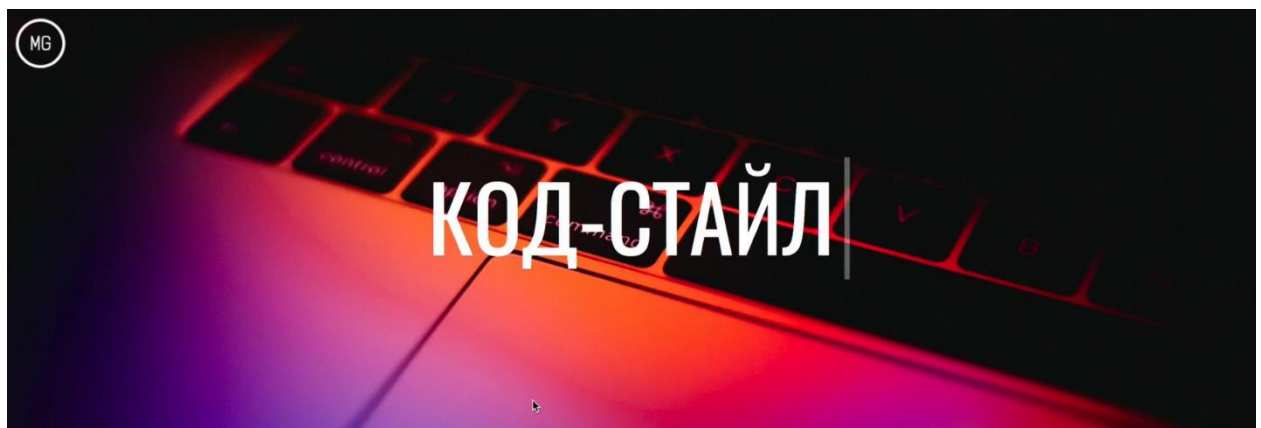
[Кодстайл](#)

ЧТО МЫ УЗНАЕМ СЕГОДНЯ?

Код-стайл — это стиль кода, его оформление. Мы узнаем, как использовать его в CSS.

ГДЕ УЗНАТЬ, КАК ПРАВИЛЬНО ОФОРМЛЯТЬ КОД?

С правилами кодстайла для данного курса можно ознакомиться на сайте [Codeguide Maxgraph](#).







ОГЛАВЛЕНИЕ

HTML

- | | | | |
|---------------------|---------------------------|--------------------------|----------------|
| 1. СИНТАКСИС | 5. КОДИРОВКА | 9. ПОРЯДОК АТТРИБУТОВ | 13. ВАЛИДНОСТЬ |
| 2. СЕЛЕКТОРЫ | 6. АТТРИБУТ ЯЗЫКА | 10. ЛОГИЧЕСКИЕ АТТРИБУТЫ | |
| 3. ЗАКРЫВАЮЩИЕ ТЕГИ | 7. РЕЖИМ СОВМЕСТИМОСТИ IE | 11. СВЯЗИ ПОЛЕЙ ВВОДА | |

ОБЩИЙ ПРИМЕР КОДСТАЙЛА

Всегда используется отступ два пробела. Стили пишутся друг за другом по категориям — сначала позиционирование, затем блочная модель, типографика и далее.

```
# style.css >  .element
1  .element {
2      /* Позиционирование */
3      position: relative;
4      left: 0;
5      right: 0;
6      top: 0;
7      bottom: 0;
8
9      /* Блочная модель */
10     display: flex;
11     align-items: center;
12     margin: 10px;
13     padding: 10px 20px;
14     border: 1px solid  red;
15     width: 200px;
16     height: 100px;
17     box-sizing: border-box;
18
19     /* Типографика */
20     font-family: Arial;
21     font-size: 25px;
22     font-style: italic;
23     text-decoration: none;
24     color:  red;
25
26     /* Оформление */
27     background:  red;
28     opacity: 1;
29
30     /* Анимация */
```

На этом всё!

3.8 Задание к лабораторной работе

Задача

Цель задания

Стилизовать страницу из предыдущего модуля и научиться подключать фоновые изображения.

Внимание: 17 пунктов задания являются обязательными.

2 последние пункта являются необязательными.

Что важно учесть в процессе работы над заданием

На нашем курсе есть свой [код-стайл](#). При работе над проектом вам нужно строго придерживаться код стайла курса, код должен быть написан единообразно.

Также очень важно, помимо валидации HTML, валидировать и CSS. Код должен быть правильным, валидным, чистым.

Осмысленная и интуитивно понятная структура проекта является одной из важнейших частей разработки. Необходимо изначально приучить себя к тому, что проект должен быть структурирован и понятен не только вам, но и другим разработчикам.

Все исходники проекта (кроме html файлов) нужно хранить в соответствующих папках.

Папка “css” для стилей, папка “img” для изображений, папка “fonts” для шрифтов и т.д.

Что нужно сделать

1. В своем рабочем пространстве создайте папку, например, “Lab03” и скопируйте туда файлы из предыдущей лабораторной работы.

Скачайте приложенные к домашнему заданию файлы (папку fonts, style.css, return.png) и вложите их в созданную папку.

2. Перенесите файл style.css в папку css, а все изображения в папку img.

Подсказка: Как вы могли заметить, структура нашего проекта изменилась. В следствии этих изменений вам нужно поправить пути к файлам.

Если потребуется, пересмотрите урок, посвященный [путям](#).

3. Подключите style.css к вашим двум HTML-файлам и убедитесь, что он сработал (внешний вид страниц немного изменится).

Пояснение: В style.css написан код для старта стилизации html документов. Данный код подключает шрифты и производит “настройку” некоторых css свойств.

Для дальнейшей работы над стилями важно не просто скопировать css файл в папку проекта, а разобрать каждое css правило и свойство, дабы понять, как и на что они влияют.

Стилизируйте index.html

1. Оберните всё содержимое элемента body, в <div> с классом main-content.

2. Напишите css правило по селектору класса main-content, в этом правиле нужно описать следующие css свойства:

display: flex;

flex-direction: column;

justify-content: center;

align-items: center;

height: 100vh;

background-color: #222020;

Затем оберните ссылки «скачать» и «подробнее» в <div> - обертку.

3. Используйте для стилизации элементов только селекторы по классу. Так мы будем делать на протяжении всего курса. Исключения будут четко указаны в описании к соответствующим заданиям.

Важно: Именованное классов должно соответствовать следующим критериям:

- имена классов должны быть краткими и информативными,
- при чтении селектора, должно быть сходу понятно, что этот элемент делает, о чем он.
- в имени класса старайтесь описывать не внешний вид(red, big) и не расположение(left, top...), а суть, смысл, назначение. - ".page-header", "nav-list", "news-item" и т.п.

[Примеры.](#)

4. Задайте для заголовка:

- белый цвет,
- font-family: 'Graphik LCG', sans-serif,
- внешний отступ снизу, равный 30 пикселям.

5. Задайте ссылке «скачать»:

- внутренний отступ 20 пикселей,
- цвет — белый,
- цвет фона — #2f1ce4 (далее будем называть его “цвет WebD”),
- бордер толщиной 3 пикселя тоже цвета WebD.

Подробнее про свойство border прочитайте по [ссылке](#).

6. Задайте ссылке «подробнее»:

- внутренний отступ 20 пикселей,
- цвет — белый,
- цвет фона — прозрачный,
- бордер толщиной 3 пикселя белого цвета.

Стилизируйте program.html

1. В файле program.html — оберните все содержимое body в <div> с классом container.

2. Задайте <div class="container">:

- ширину 1170 пикселей,
- margin: 0 auto,
- внутренние отступы сверху и снизу по 30 пикселей.

3. Для ссылки «назад» реализуйте иконку стрелки.

Подсказка: поскольку это декоративное изображение, нужно использовать свойства из группы background.

Познаем «силу» классов и выравниваем заголовки

1. Для главного заголовка страницы `program.html` напишите следующие стили:

- отступ сверху 100 пикселей,
- внутренний отступ слева — 100 пикселей,
- размер шрифта — 48 пикселей,
- цвет шрифта: `#222020`.

2. На странице есть “одинаковые” (стили для нескольких элементов совпадают) элементы. Чтоб не писать несколько раз один и тот-же `css` код (дублировать):

- напишите одно `css` правило по селектору на определенный класс
- задайте этот класс всем “одинаковым” `html` элементам.

Итог: в документе, к которому подключен наш `css` файл, стили, написанные в `css` правиле по селектору на “определенный класс”, применяются ко всем элементам с данным классом.

Подсказка: На картинке есть одинаковые заголовки цвета `WebD`. Задайте им один класс, напишите `css` правило на селектор по этому классу, в правиле нужны следующие свойства:

- внутренний отступ слева — 180 пикселей,
- размер шрифта — 36 пикселей,
- цвет — цвет “`WebD`”,
- отступ сверху — 100 пикселей,
- отступ снизу — 25 пикселей.

3. Задайте одинаковые классы для списков. Стили, которые должны применяться к спискам:

- внутренний отступ слева — 200 пикселей,
- размер шрифта — 18 пикселей.

4. Задайте одинаковые классы для всех параграфов. Стили для них:

- внутренний отступ слева — 180 пикселей,
- отступ снизу 20 пикселей.

Финальные штрихи

1. Во второй л/р у вас было необязательное задание. В этой работе оно является уже обязательным. Если вы еще не сделали его - вам нужно сделать следующее:

В css-файле для таблицы и ее ячеек напишите следующие стили:

отступ сверху 100 пикселей

Бордер толщиной 1 пиксель и черного цвета

Отдельно для таблицы напишите - `border-collapse: collapse;`

Отдельно для ячеек таблицы (th, td) напишите внутренний отступ 5px;

Важно: для ячеек можно не задавать множество классов. При использовании селектора по тегу пользуйтесь вложенностью селекторов: `.table th, .table td {}`, чтобы стили ячеек применились только к конкретной таблице с классом `table`.

Необязательные задания этого модуля:

1. Для figure задайте:

- внутренний отступ слева — 200 пикселей,
- текст — по центру,
- строчно-блочное отображение,
- отступы сверху и снизу по 50 пикселей.

Важно: селекторы мы используем только по классу!

Готово! Сравните работу с приложенной картинкой.

Важно: Если вы не выполняли необязательное задание — сравнивайте страницу `program.html` с картинкой “Как должно выглядеть (без доп задач)”.

Критерии оценки

«Зачёт»

1. Вы прислали архив с выполненной разметкой страниц `index.html` и `program.html`, а также их стили в папке `css`.
2. Архив содержит все необходимые файлы.
3. Вы выполнили все требования к макету в соответствии со списком обязательных задач.

4. Валидатор HTML, Валидатор CSS и HTMLHint показывают, что ошибки отсутствуют.

5. Успешность выполнения необязательных заданий не влияет на зачет/незачет.

«Незачёт»

1. Вы не выполнили обязательные пункты задания и/или допустили ошибки в их решении.

2. В прикрепленном архиве отсутствуют требуемые файлы:

index.html - первая страница

program.html - вторая страница

.editorconfig - конфиг-файл с настройками

Папка img со всеми изображениями проекта

Папка css с файлом style.css

Папка fonts со шрифтами

Документ в формате .docx