

Лабораторная работа #02

2.1 Ускорение работы. Emmet

Описание

Плагин Emmet уже установлен в VS Code. Чтобы он корректно работал, сперва нужно создать файл с нужным расширением (html, css).

[Описание плагина Emmet на английском](#)

[Сокращения emmet](#)

2.2 Элементы HTML-разметки. Базовые

теги

Описание

Полезные ссылки:

[Базовые теги](#)

[Справочник html](#)

[Основы CSS](#)

2.3 Теги картинок и ссылок. Кнопки

Описание

Полезные ссылки:

- [тег img](#)
- [тег ссылки](#)
- [тег кнопки](#)
- [Когда использовать кнопки и ссылки](#)

2.4 Теги таблиц

Описание

[Таблицы](#)

2.5 Служебные теги

Описание

[Элемент мета](#)

2.6 Кодстайл HTML

Описание

[Кодстайл](#)

[Простые правила разметки](#)

2.9 Задание к лабораторной работе

Задача

Цель задания

В процессе верстки вашего первого сайта из двух страниц закрепить на практике работу с HTML-тегами.

Внимание: 9 пунктов задания являются обязательными для выполнения. Пункт 10 - является дополнительным заданием. Также обращайте внимание на “Подсказки” и рекомендации.

Что вам нужно сделать

Перед началом работы

В прикрепленных материалах вы увидите архив с файлом .editorconfig (это файл без названия, но с расширением). Данный файл отвечает за настройки форматирования (например, размер табуляции) кода, благодаря расширению. Разархивируйте файл .editorconfig и вставьте его в корень проекта. Для VS Code используйте расширение “[EditorConfig for VS Code](#)”. Конфиг-файл (.editorconfig) вам нужен для того, чтобы от проекта в проект у вас было одно и то же форматирование.

Важно: Если у вас не получается сделать файл без имени в системе — откройте его с помощью VS Code и уже там, внутри, переименуйте.

Что важно учесть в процессе работы над заданием

При работе над проектом вам нужно учитывать код-стайл проекта, чтобы писать единообразный код по правилам. Мы будем использовать [код-стайл](#), который необходимо соблюдать в этой и всех последующих лабораторных работах.

Контент и внешний вид страниц должен соответствовать изображениям, прикрепленным к текущей лабораторной работе (посмотрите файлы “как должно выглядеть”).

1. Внимательно изучите макеты страниц (изображения) и самостоятельно определите, из каких элементов (тегов) они состоят. Например, на макете “как должно выглядеть 2” отчётливо видны заголовки “Описание курса” и “Программа курса”. Они выглядят одинаково, значит это будет одинаковый уровень заголовка.

А вот текст “Информация о курсе” отличается, следовательно, это уже будет другой уровень заголовка. В таком ключе вам нужно проанализировать все элементы на первом и втором макетах и только после этого приступить к разметке в редакторе кода.

2. Сделайте разметку первой страницы в соответствии с тем, что изображено на макете “как должно выглядеть 1”. Для этого создайте папку, вставьте туда `editorconfig`, создайте файл `index.html`.

Подсказка: в html документе должен быть заголовок, внутри него ссылка на страницу нашего курса в Moodle, открытие ссылки должно происходить в новой вкладке.

3. К заданию приложен документ в формате docx. Добавьте в разметку ссылку, по нажатию на которую будет скачиваться данный документ.

Подсказка: для ссылки укажите атрибут `download`.

4. Добавьте в документ еще одну ссылку, которая будет вести на страницу `program.html` (открытие страницы в новой вкладке делать не нужно).
5. Сделайте разметку второй страницы в соответствии с тем, что изображено на макете “как должно выглядеть 2”. Создайте файл `program.html`.
6. Добавьте в документ ссылку с текстом “назад”. Эта ссылка должна вести на страницу `index.html`.
7. В макете есть ссылка “Изучить теги”. При клике на эту ссылку происходит переход к таблице, которая находится внизу макета. Реализуйте данный функционал.
8. **Важно учесть:** Для разметки картинок и подписей к ним используйте элементы `img`, `figure` и `figcaption`.
9. В макете “как должно выглядеть 2” в самом низу находится таблица с именами элементов и их описанием. Сделайте разметку этой таблицы. Для реализации примеров html-кода используйте элемент `code`, а также [спецсимволы](#), которые заменят вам `<>` (угловые скобки у тега).

Подсказка: Разницу в отображении документа со спецсимволами и без них можно увидеть [по ссылке](#).

Подсказка: для реализации заголовка таблицы используйте тег `caption`

Подсказка: Используйте подходящие элементы для разметки таблицы (`thead`, `tbody`, `tfoot`) — о них мы говорили на занятии про таблицы.

Дополнительное задание

10. Чтобы привести таблицу в соответствие с картинкой, используйте CSS. Как его подключать мы говорили на занятии. Также рекомендую прочитать эту [статью](#).
В css-файле для тегов `table`, `td` и `th` напишите следующий код:
`border: 1px solid #000;` (обратите внимание на то, что запятые отсутствуют, иначе `border` не сработает);
Также, отдельно для тега `table` напишите - `border-collapse: collapse;`
И отдельно для ячеек таблицы (`th`, `td`) напишите `padding: 5px`.

Важно! Перед отправкой работы на проверку **всегда** проверяйте код в [валидаторе](#) и HTMLHint. Краткая инструкция по нему по [ссылке](#).

Критерии оценки

«Зачёт»

1. Вы прислали архив с выполненной разметкой страниц index.html и program.html.
2. Архив содержит все необходимые файлы.
3. Вы выполнили все требования к макету в соответствии со списком задач выше (9 обязательных пунктов).
4. Валидатор и HTMLHint показывают, что ошибки отсутствуют.
5. Успешность выполнения задания 10 не влияет на зачет/незачет.

«Незачет»

1. Вы не выполнили 9 обязательных пунктов 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
Две картинки с сайтами
Документ в формате .docx

Формат сдачи

Архив с версткой (html-файл и приложенные картинки + doc).