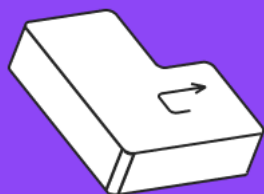


Урок 8. Bootstrap

Веб-вёрстка HTML / CSS



Оглавление

Bootstrap	2
Основные преимущества	2
Основные инструменты	3
Как начать использовать Bootstrap	4
FTP-клиент	8
Filezilla	9
Отладчик браузера	9
Валидатор	10
Как пользоваться валидатором	11
Оптимизация изображений	11
Совместимости разных версий браузеров	12
Условный комментарий	12
Как разместить сайт в интернете	13

Bootstrap

Bootstrap (также известен как Twitter Bootstrap) — свободный набор инструментов для создания сайтов и веб-приложений. Включает в себя HTML- и CSS-шаблоны оформления для типографики, веб-форм, кнопок, меток, блоков навигации и прочих компонентов веб-интерфейса, включая JavaScript-расширения. Bootstrap использует самые современные наработки в области CSS и HTML.

Основные преимущества

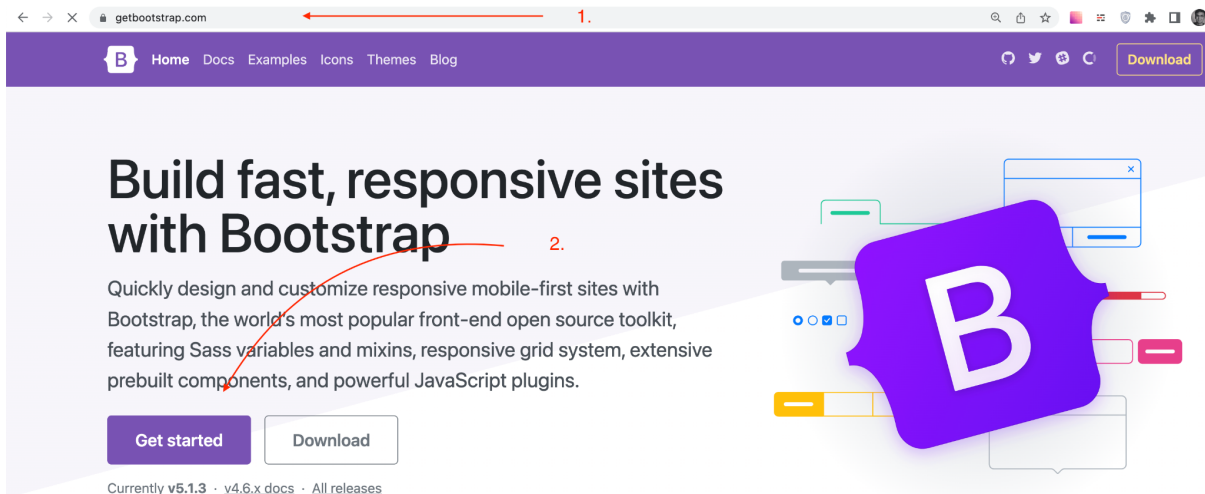
1. Экономия времени — Bootstrap позволяет сэкономить время и усилия, используя шаблоны дизайна и классы, и сконцентрироваться на других разработках.

2. Высокая скорость — динамичные макеты Bootstrap масштабируются на разные устройства и разрешения экрана без каких-либо изменений в разметке.
3. Гармоничный дизайн — все компоненты платформы Bootstrap используют единый стиль и шаблоны с помощью центральной библиотеки. Дизайн и макеты веб-страниц согласуются друг с другом.
4. Простота в использовании — пользователь с базовыми знаниями HTML и CSS может начать разработку с Twitter Bootstrap.
5. Совместимость с браузерами — Twitter Bootstrap совместим с Mozilla Firefox, Яндекс.Браузер, Google Chrome, Safari, Internet Explorer и Opera.
6. Открытое программное обеспечение — особенность Twitter Bootstrap, которая предполагает удобство использования посредством открытости исходных кодов и бесплатной загрузки.

Основные инструменты

1. Сетки — заранее заданные размеры колонок, которые можно сразу же использовать. Например, ширина колонки 140px относится к классу `.span2` (`.col-md-2` в третьей версии фреймворка), который можно использовать в CSS-описании документа.
2. Шаблоны — фиксированный или резиновый шаблон документа.
3. Типографика — описания шрифтов, определение некоторых классов для шрифтов, таких как код, цитаты и т. п.
4. Медиа — предоставляет некоторое управление изображениями и видео.
5. Таблицы — средства оформления таблиц вплоть до добавления функциональности сортировки.
6. Формы — классы для оформления форм и некоторых событий, происходящих с ними.
7. Навигация — классы оформления для табов, вкладок, пагинации, меню и тулбара.
8. Алерты — оформление диалоговых окон, подсказок и всплывающих окон.

Как начать использовать Bootstrap



1. Переходим на официальный сайт <https://getbootstrap.com/>
2. Выбираем онлайн подключение “Get started”

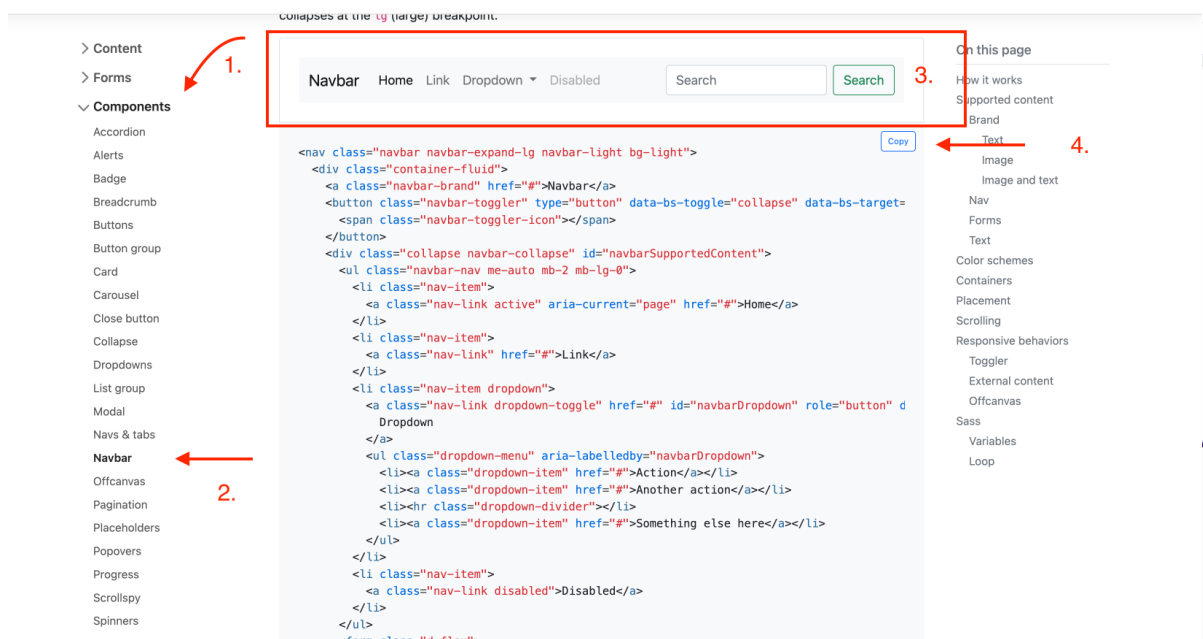


3. Внутри тега `<head>` скопировать ссылку link и добавить выше подключения вашего файла стилей
4. Скопировать подключение script и добавить перед закрывающимся тегом `</body>`

```
1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5   <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css"
6     rel="stylesheet"
7     integrity="sha384-1BmE4kWBq78iYhFldvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3"
8     crossorigin="anonymous">
9   <meta charset="UTF-8">
10  <meta http-equiv="X-UA-Compatible" content="IE=edge">
11  <meta name="viewport" content="width=device-width, initial-scale=1.0">
12  <title>Document</title>
13  <link rel="stylesheet" href="style.css">
14 </head>
15
16 <body>
17   <!-- Тут код сайта -->
18
19   <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js"
20     integrity="sha384-ka7Sk0Gln4gmtz2MlQnikT1wXgYs0g+OMhuP+IlRH9sENB00LRn5q+8nbTov4+1p"
21     crossorigin="anonymous"></script>
22
23 </body>
24 </html>
```

В результате получаем вот такой шаблон.

Как добавить компонент на сайт



1. В боковом меню выбираем Components
2. Выбираем необходимый

3. Смотрим внешний вид, если не подходит, прокручиваем ниже и находим подходящий
4. Нажимаем сору для того, чтобы скопировать код

Итоги

1. Вы можете быстро создать простой сайт, просто скопировав несколько готовых компонентов
2. Вы можете дополнить уже существующий проект, готовыми компонентами
3. Вы можете стилизовать или менять любой добавленный компонент Bootstrap
4. Вы можете добавить слайдер на страницу, просто скопировав код компонента слайдера из библиотеки Bootstrap

Проверка сайта на ошибки

Большинство начинающих разработчиков, при создании сайта, не обращают внимание на удобство сайта для пользователя, основной задачей является реализация проекта, а удобство использования проекта для пользователя уходит на второй план. Но нам нужно помнить что разработчик с опытом в разработке сайтов отличается тем, что он создает проект к которому невозможно придраться, который будет максимально продуман для использования пользователем.

Давайте рассмотрим на что обязательно стоит обратить внимание, при создании проекта.

1. Добавление эффектов наведения и нажатия. Вам нужно помнить что кнопки без эффектов наведения и нажатия намного меньше используются, следовательно наш сайт будет меньше продавать товаров или выполнять любых действий
2. Семантические элементы, мы на прошлых уроках рассмотрели такие теги как `<header>` `<footer>` `<nav>` и тд, если мы их не будем использовать, то наш сайт будет намного сложнее продвигать в сети интернет, наш код будет менее понятен для программиста, который будет поддерживать данный сайт
3. Добавление изображений в формате SVG, мы уже умеем работать с svg графикой, но если вы перепутали и добавили иконку социальной сети в формате png, в чем может быть проблема? Конечно же на png картинку нельзя добавить эффект наведения и чтобы в ней что-то поменялось, нельзя

сделать масштабирование без потери качества, картинка в формате png будет весить больше, чем svg

4. Добавление текста изображением, если мы видим текст в макете, то его необходимо добавлять именно текстом, не стоит скачивать его как изображение, тем более делать скриншот
5. Проверить, как выглядит сайт на большем или меньшем разрешении экрана. Данный вариант мы подробно разбирали на уроке по созданию адаптивного проекта
6. Логотип должен быть ссылкой на главную страницу. Эта правда является одной из самых частых, так как об этом нигде не пишут, но постоянно требуют реализовать такой функционал
7. Стандартные эффекты наведения или фокуса в макете, тут проще всего вспомнить поле ввода с синей рамкой, стандартной. Вам нужно исключать такие стандартные эффекты в вашем проекте.
8. Проверить большого количества текста, ваш сайт не должен рушиться из-за того, что в вашем проекте станет в 2 или 3 раза больше текста
9. Проверить сайт в типографе. На первых уроках мы рассматривали подготовку текста к веб изданию, вам нужно не забыть добавить спецсимволы, если такие требуются <https://www.artlebedev.ru/typograf/>
10. Отступы для полей ввода, любое поле ввода, которое вы создаете, необходимо проверить, просто введите туда текст и посмотрите не прижимается ли введенный контент к левой или правой границе, если это так, добавляйте отступы padding слева и справа.
11. Проверка сайта на валидность <https://validator.w3.org/> (об этом инструменте вы узнаете, когда прочитаете материал полностью)

Инструменты разработчика

Какие инструменты могут понадобиться веб-разработчику:

1. FTP-клиент: FileZilla, WinSCP, Total Commander, Cyberduck.
2. Плагины: Firebug, Web Developer.
3. [Валидатор](#).
4. Оптимизация изображений: TinyPNG.
5. Совместимости разных версий браузеров: [Can I Use](#).

FTP-клиент

FTP — это протокол передачи файлов. Через него файлы можно скопировать с компьютера на сервер и обратно. Ещё одна из часто используемых возможностей — редактирование файлов на сервере.

Для использования FTP необходима специальная программа. Их существует большое количество, некоторые из них бесплатные, другие могут быть коммерческими. Практически все программы FTP работают таким образом: появляются два окна, где первое — ваши файлы на сервере в интернете, а второе — файлы на компьютере. Можно легко передавать файлы на компьютер или сервер.

Предположим, программа FTP уже выбрана и установлена на компьютер, но для работы необходимо знать FTP-адрес вашего веб-сервера, имя пользователя и пароль. Это важная информация, её можно получить на хостинге и нужно обязательно записать. Если все данные есть, можно попробовать получить доступ к файлам FTP.

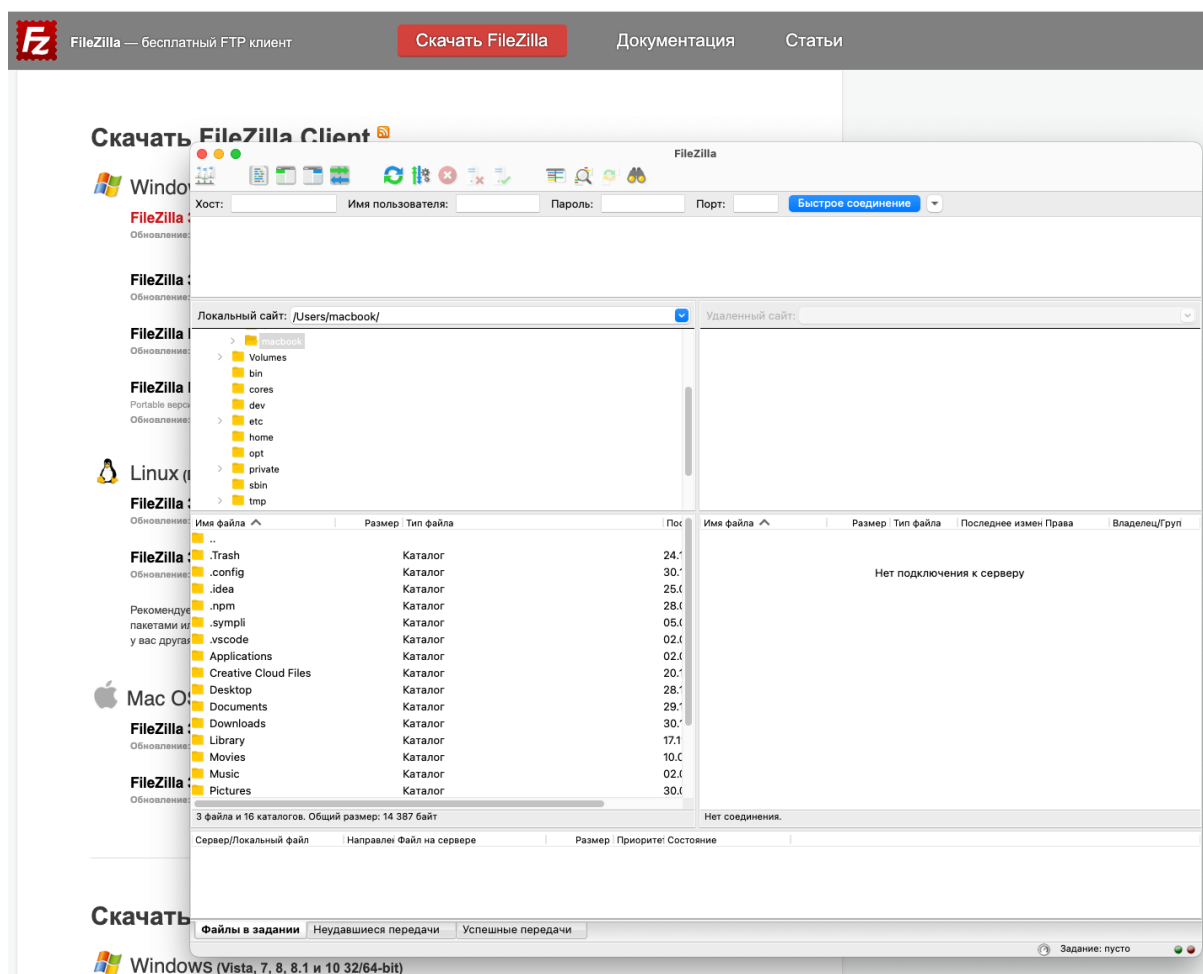
Действия могут быть разными в зависимости от программы, но в любом случае нужно создать новое соединение с именем пользователя и паролем. Это позволит сохранить настройки и исключить необходимость ввода данных каждый раз при входе в программу.

После создания профиля и ввода всех данных нажмите «Соединиться», после чего программа начнёт соединяться с сервером. Если все настройки верны, через несколько секунд (в зависимости от скорости интернета) можно будет увидеть все свои файлы, находящиеся на сайте.

На веб-сервере находится набор папок и файлов. Подключаясь к серверу, программа отправляет запрос через интернет на специальный сервер и получает данные. Существуют протоколы и с более расширенным функционалом, нежели обычный веб-сёрфинг.

FTP-программа даёт возможность создавать, редактировать, удалять файлы и папки на веб-сервере, а также передавать файлы на сервер и обратно.

Filezilla



У программы очень простой внешний вид. Вам необходимо на хостинге скопировать данные хоста, имя пользователя и пароль, чтобы справа отобразилось содержимое сервера.

Программа бесплатна и идеально подходит для большинства современных операционных систем.

Отладчик браузера

На первых уроках мы изучали инструмент разработчика для просмотра кода элемента. Чаще всего на первых занятиях ученики не понимают, пригодится им эта программа или нет, но уже на занятии по созданию адаптивного сайта ученики понимают что без данного инструмента, разработка невозможна.

В отладчике браузера можно:

- просматривать HTML-код элемента;
- инспектировать любые сайты в интернете;
- точно определять структуру сайта;
- вносить изменения в код;
- многое другое.

Валидатор

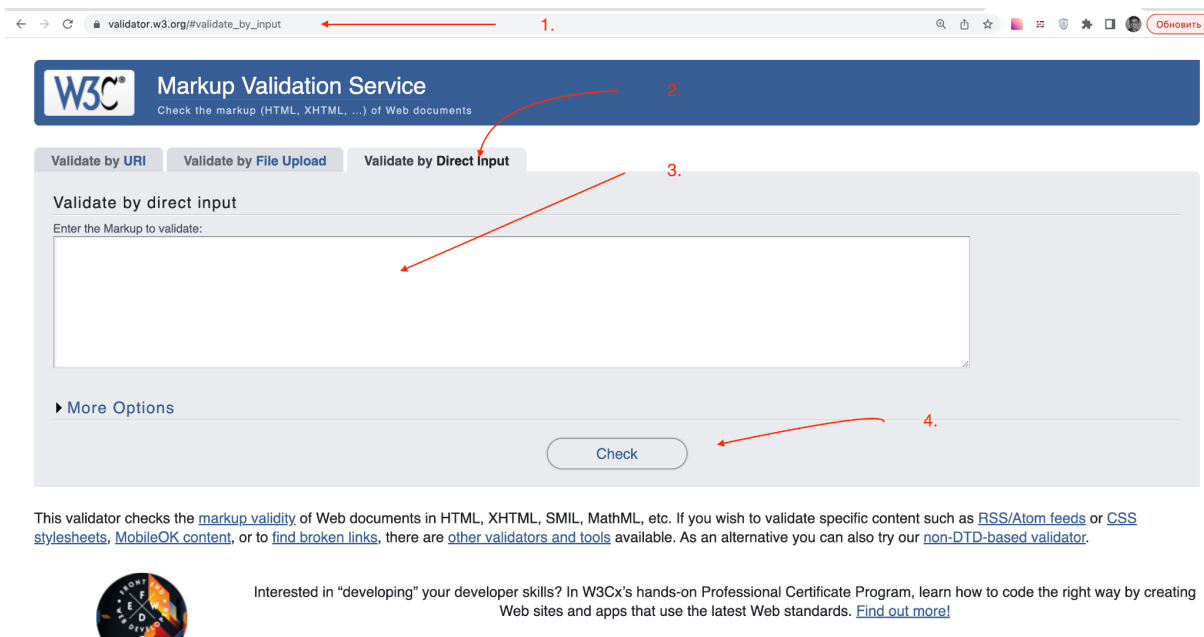
Чистый валидный код в HTML и CSS повышает скорость загрузки и шансы, что на всех без исключения устройствах и браузерах ваша вёрстка будет выглядеть так, как вы хотите. Поэтому любой вёрстке нужно проводить валидацию. Валидаторы W3C — два сервиса, которые помогают держать код в чистоте и порядке.

Есть три варианта проверки: проверить код в файле по указанной ссылке, загрузить файл с десктопа или скопировать код прямо в форму на сайте.

HTML-валидатор производит несколько проверок кода:

1. Валидация синтаксиса — проверка на наличие синтаксических ошибок.
2. Проверка вложенности тегов — они должны быть закрыты в обратном порядке относительно их открытия.
3. Валидация DTD — проверка соответствия вашего кода указанному Document Type Definition. Она включает проверку названий тегов, атрибутов и «встраивания» тегов.
4. Проверка на посторонние элементы выявляет всё, что есть в коде, но отсутствует в DTD.

Как пользоваться валидатором



1. Переходим на официальный сайт https://validator.w3.org/#validate_by_input
2. Выбираем раздел ввода кода
3. Копируем весь код html, вашего сайта
4. Нажимаем check для проверки
5. Исправляем ошибки, если валидатор их определил

Оптимизация изображений

Оптимизация графики — не менее важный аспект вёрстки, чем чистота кода. Из третьего урока мы помним, что изображения бывают разных форматов. Для вёрстки самое важное — это размер файла. При скачивании изображений в необходимом формате размер файла может быть слишком большим, поэтому после экспорта PNG-файлов рекомендуется оптимизировать графику в TinyPNG. После обработки файл теряет 30–70% веса, но на качество это никак не влияет.

Совместимости разных версий браузеров

В процессе работы любой веб-разработчик сталкивается с таким понятием, как кроссбраузерность. Оно означает, что любой сайт должен выглядеть одинаково во всех браузерах. Главный виновник этой проблемы — Internet Explorer (IE).

Пути решения проблемы кроссбраузерности:

1. Необходимо проверять, как выглядит сайт во всех популярных браузерах, с некоторой периодичностью во время вёрстки. Тогда вы сможете оперативно устранять несоответствия. (Для этого вам потребуется установить все современные версии браузеров, какие именно браузеры устанавливать, определяет работодатель)
2. Некоторые теги HTML, например заголовки, параграфы, списки, изначально имеют определённый набор свойств и значений. Они могут определяться каждым браузером по-разному. Чтобы вид HTML-страницы не зависел от того, с помощью какого браузера её просматривают, используется сброс стилей CSS. Обычно, если используется сброс стилей, их определяют в отдельном файле, который подключают на веб-страницу при помощи тега `<link>`.
3. Воспользоваться хаками. Хак — исправление ошибки или добавление новой функции посредством использования другой недокументированной или некорректно реализованной особенности. Необходимость добавлять хаки возникает в основном для IE, но ими пользоваться не рекомендуется, так как это противоречит спецификации CSS. Чтобы исправить несоответствия отображения в IE, лучше воспользоваться решением из следующего пункта.
4. Объявить условный комментарий. Он представляет собой обычный HTML-комментарий, где в квадратных скобках указывается условие, которое понимает только браузер IE, а все остальные браузеры игнорируют.

Условный комментарий

```
<!--[if gte IE 6]>
  <link href="css/style_ie.css" rel="stylesheet" type="text/css" />
<![endif]-->
```

То, что находится между `if` и `endif`, IE выполнит. В данном случае подключится файл стилей CSS. В нём можно писать только те стили CSS, которые будут относиться к IE.

Чтобы точно определять, к каким версиям IE применять стили, можно использовать дополнительные логические операторы:

- lt — версия меньше указанной;
- lte — версия меньше или равна указанной;
- gt — версия больше указанной;
- gte — версия больше или равна указанной.

При помощи других логических операторов (& — «и», | — «или», ! — «не») выполняется группировка условных комментариев для различных версий IE:

- [if (IE 6) & (IE 7)] — 6-я версия **и** 7-я версия;
- [if (IE 6) | (IE 7)] — 6-я версия **или** 7-я версия;
- [if !(IE 8)] — **не** 8-я версия.

Как разместить сайт в интернете

Технологии не стоят на месте, поэтому размещение сайта в интернете — несложная задача. Для этого не обязательно скачивать сторонние программы, так как подключение к хостингу может происходить с помощью браузера.



Внимание! Последовательность действий может отличаться, так как бесплатные хостинги могут становиться платными, могут появляться новые хостинги.

Разберём примерную последовательность действий, необходимых, чтобы ваш проект оказался в интернете:

1. Выбираем любой бесплатный хостинг.
2. Выбираем бесплатный тариф.
3. Регистрируемся (при необходимости).
4. Нажимаем «Создать сайт» (придумываем имя сайта).
5. Нажимаем кнопку «Управлять сайтом».
6. Выбираем файловый менеджер.
7. Загружаем всё, что создали, в файловый менеджер.

Итоги курса

Задачей данного курса было не просто познакомить вас с технологиями html/css и показать как они работают, хотелось собрать самый актуальный и качественный материал, чтобы курс был не перегружен информацией, одновременно с этим чтобы в нём было абсолютно всё что нужно для создания сайта с помощью html/css. Давайте вспомним с чего всё начиналось? Первым делом мы узнали основные особенности веб-разработки, узнали что такое интернет и для чего нам понадобятся технологии html и css. Узнали как добавить контент сайта, как создать простую форму регистрации. На втором уроке мы детально рассмотрели технологию css, узнали как правильно добавить стили, как правильно обращаться к селекторам, почему называется каскадная таблица стилей. Третье наше занятие было посвящено работе с макетом, так как чем раньше мы приступим к реальной практике, тем лучше, мы рассмотрели все необходимые особенности работы с макетом и узнали насколько важен макет для верстки сайта. Четвертый урок был посвящен основам позиционирования элементов, естественно такую объемную тему невозможно рассмотреть за один урок, поэтому мы рассматривали огромное количество способов позиционирования, смогли рассмотреть такую популярную технологию как flexbox и разобрали набирающую популярность технологию Grid layout. На пятом уроке мы перешли к созданию адаптивного проекта и рассмотрели все возможные способы и технологии для создания адаптивного проекта. На шестом занятии мы рассмотрели самую актуальную информацию про html5 и узнали как можно добавить аудио и видео на страницу. На седьмом уроке мы рассмотрели самые последние новшества css3, узнали как можно создать и добавить анимацию на сайт. В финальном нашем уроке мы научились использовать фреймворк Bootstrap, узнали как проверить свой сайт на ошибки и как можно оптимизировать работу сайта.

После прохождения курса вы обладаете всеми необходимыми знаниями и навыками чтобы справиться с созданием макета, вас не пугает адаптив и вы с уверенностью можете сказать что знакомы с технологиями html/css на отлично.