

Раздел 1. Frontend-разработка

Знакомство с GitHub. Основы языка разметки документов HTML. Основные понятия веб-разработки и структура HTML.

Знакомство с GitHub и установка окружения

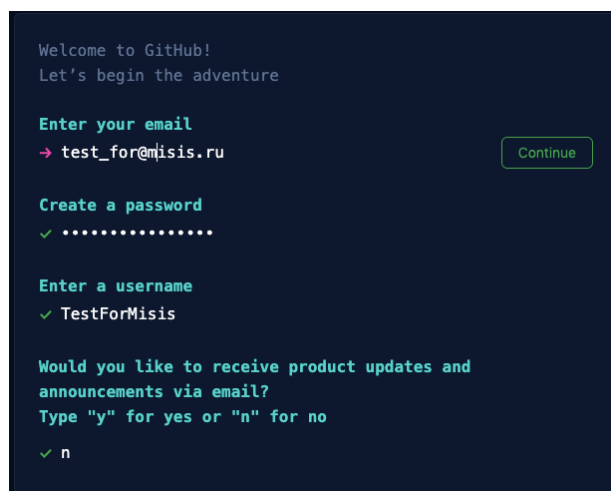
GitHub — это хостинг, т.е. сайт-хранилище, который позволяет хранить Ваши проекты удалённо на сервере и работать с ними из любой точки мира. При этом под проектом можно понимать абсолютно любые файлы или набор файлов, доступ к которым есть у всех, у кого есть ссылка.

Чаще всего GitHub используют команды разработчиков в силу того, что одной из главных функций сервиса выступает – контроль версий. Все изменения в коде можно отследить, поэтому в командной разработке это незаменимая вещь. Однако для начинающего разработчика GitHub может выступать некоторым хранилищем собственных проектов, помогающего структурировать знания и навыки в виде Repositories.

GitHub можно считать соцсетью для хранения кода.

Ход работы

1. Регистрация на хостинге GitHub (нужен доступ на электронную почту):



2. Установка визуального клиента для управления версиями SmartGit (сделать на домашнем компьютере, чтобы получить возможность выполнять задания)
 - 2.1. Перейдите по ссылке: <https://www.syntevo.com/smartgit/download/> и скачайте продукт для Вашей операционной системы.
 - 2.2. Процесс установки
3. Создание репозитория и его клонирование в SmartGit.
 - 3.1. Создайте репозиторий в GitHub, задав ему имя и описание проекта, для которого он создается.

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository](#).

Owner *

Repository name *

TestForMisis

 /

Misis-Web

Great repository names are short and memorable. Need inspiration? How about [supreme-broccoli](#)?

Description (optional)

ИТ-каникулы в НИТУ "МИСиС"

☒ Public

Anyone on the internet can see this repository. You choose who can commit.

☐ Private

You choose who can see and commit to this repository.

Initialize this repository with:

Skip this step if you're importing an existing repository.

☒ Add a README file

This is where you can write a long description for your project. [Learn more](#).

☐ Add .gitignore

Choose which files not to track from a list of templates. [Learn more](#).

☐ Choose a license

A license tells others what they can and can't do with your code. [Learn more](#).

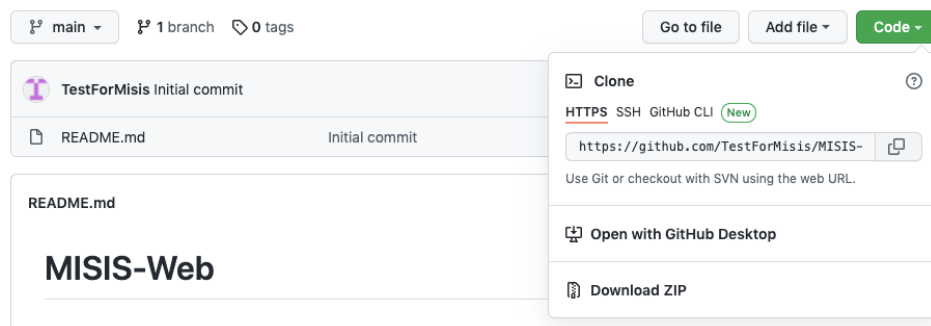
This will set

main

as the default branch. Change the default name in your [settings](#).

Create repository

- 3.2. Перейдите в созданный репозиторий и выберите кнопку Code в правой части экрана и скопируйте ссылку:



- 3.3. Откройте SmartGit и выберите вкладку *Repository* -> *Clone*, в появившемся окне появится, скопированная нами ссылка, после чего необходимо завершить процесс клонирования.

4. Создайте первую html-страницу в свободном редакторе Brackets.

- 4.1. Создайте новый файл с помощью комбинации «горячих клавиш» - Ctrl+N и сохраните его (ctrl+shift+s) с именем index.html – это главная страница сайта.

Основы языка разметки документов HTML

Структура HTML-документа

Каждая страница в HTML-документе состоит из трех обязательных элементов:

Раздел `<head></head>` с технической информацией о странице: заголовок, описание, ключевые слова для поисковых машин, кодировка. Введенная в нем информация в основном не отображается в окне браузера, однако содержит данные, которые указывают браузеру, как следует обрабатывать страницу.

Раздел `<body></body>`, где располагаются все элементы, которые видит пользователь. Именно с этим разделом мы в основном и будем работать.

DOCTYPE отвечает за корректное отображение веб-страницы браузером. DOCTYPE определяет не только версию HTML, но и соответствующий DTD-файл в интернете.

Пример структуры HTML 5:

```
<!doctype html>
<html>
<head>
  <title>Hello HTML</title>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
</head>
<body>
...
</body>
</html>
```

Теги и атрибуты

Язык HTML — язык тегов. Теги описывают структуру HTML-документа. Теги оформляются угловыми скобками, между которыми прописывается имя тега. Теги HTML документа предназначены для управления конструкциями разметки — заголовками, абзацами, списками, таблицами и картинками.

Теги бывают парные и одиночные.

Парные теги сначала открываются, потом закрываются. Все, что описывают парные теги, находится внутри них. Так, тег делает текст полужирным:

```
<b>Какое прекрасное утро</b>
```

Одиночные теги сами по себе, и как правило не меняют контент вокруг себя. Пример:

```
<br/> - тег переноса строки
```

```
<hr/> - тег горизонтальной линии
```

Атрибуты тегов определяют какие-либо дополнительные, уточняющие параметры того или иного тега. Вы можете указать внутри тега столько атрибутов, сколько вам необходимо. Атрибуты разделяются пробелами.

`<название_тега атрибут1="значение_атрибута1"> текст внутри тега </ название_тега >`

```
<a href="contacts.htm">Контакты</a>
```

```

```

Основные теги оформления текста

Заголовки

В HTML предусмотрено 6 уровней заголовков, первый из которых будет отображен самым крупным шрифтом, а далее, чем выше уровень, тем размер шрифта будет уменьшаться.

Для отображения заголовков существует тег, и указывается цифра от 1 до 6, которая соответствует уровню заголовка.

```
<h1>Заголовок первого уровня</h1>
```

```
<h2>Заголовок второго уровня</h2>
```

...

Перечень некоторых полезных тегов:

```
<p>Здесь мы напишем параграф</p>
```

```
<hr> Горизонтальная черта
```

```
<i>Текст курсивом</i>
```

```
<small> Уменьшает размер шрифта на единицу по отношению к обычному тексту.
```

```
<sub> Используется для создания нижних индексов. Сдвигает текст ниже уровня
строки, уменьшая его размер.
```

```
<sup> Используется для создания степеней и верхних индексов. Сдвигает текст выше
уровня строки, уменьшая его размер.
```

Виды ссылок

Относительные ссылки используются для перемещения внутри документа или сайта.

```
<a href="file_name.html"> текст ссылки, который видит пользователь</a>
```

Абсолютные ссылки используются для перехода на страницы внешнего сайта, для этого в значении атрибута href нужно указать полный путь до той страницы, на которую мы хотим перейти, включая тип протокола

```
<a href="http://yandex.ru" target="_blank"> страница yandex.ru откроется в новой
вкладке</a>
```

У тега ссылки существует атрибут target. Если указать target="_blank" , то страница откроется в новой вкладке браузера

Списки

Списки в html-документах бывают двух видов: маркированные и нумерованные.

Структура маркированного списка

```
<ul>
<li>Первый элемент списка</li>
<li>Второй элемент списка</li>
<li>Третий элемент списка</li>
</ul>
```

Структура нумерованного списка

```
<ol>
<li>Первый элемент списка</li>
<li>Второй элемент списка</li>
<li>Третий элемент списка</li>
</ol>
```

Спецсимволы

Используя спецсимволы в HTML, можно заменять уже имеющиеся на клавиатуре компьютера символы или указать те, что отсутствуют на клавиатуре. Например, значок копирайта.

Любые теги никак не показываются в окне браузера, поскольку воспринимаются им как команды для вставки элементов и изменения их свойств. Но иногда требуется на веб-странице вывести теги, например, для демонстрации HTML-кода. В этом случае используйте спецсимволы < и > для замены угловых скобок < и >.

Некоторые варианты спецсимволов:

Код в HTML	Внешний вид	Описание
 		неразрывный пробел
©	©	знак copyright
" и »	« и »	двойные кавычки
< и >	< и >	символы
′	'	одиночный штрих

Комментарии

В HTML-документах можно оставить какой-либо комментарий, например, название части страницы. Это не отобразится в браузере. Но любой желающий сможет увидеть этот текст, если воспользуется функцией «просмотр кода страницы» в любом браузере.

Комментарии применяются для того, чтобы не отображать теги в браузере. Это может пригодиться при редактировании веб-сайт целые блоки кода.

Пример:

```
<!--Текст, который не будет отображаться на сайте-->  
<p>Текст основной части сайта</p>
```

Формы и их элементы

Формы — один из важных элементов любого сайта. Они нужны, чтобы получать от пользователя информацию, отправлять эти данные на сервер, а там уже их обрабатывать. Документ может содержать любое количество форм. Но одновременно на сервер может быть отправлена только одна. По этой причине данные форм должны не зависеть друг от друга.

Допускается внутри контейнера `<form>` помещать другие теги. При этом сама форма никак не отображается на веб-странице, видны только её элементы и результаты вложенных тегов.

Элементы форм

В HTML существует три основных тега элементов форм:

1. Тег `<input>` в зависимости от значения атрибута `type` будет выглядеть по-разному и иметь различные значения. Тег `<input>` одиночный.
2. `<textarea>` используется, чтобы сформировать многострочное поле ввода для информации. С помощью атрибута `cols` можно задать ширину этого поля, а с помощью атрибута `rows` — указать число строк. Тег `<textarea>` парный.
3. `<select>` представляет собой выпадающий список, элементы которого указываются в теге `<option>`. У этого тега есть несколько атрибутов. Чтобы сделать возможность выбора нескольких пунктов, нужно задать атрибут `multiple`, в значении которого указывается `multiple`. В атрибуте `size` определяется число строк выпадающего списка. Их увидит пользователь в браузере. Если необходимо, чтобы какой-то из элементов выпадающего списка был выбран, нужно этому элементу в теге `<option>` задать атрибут `selected`. Тег `<select>` также парный.

Рассмотрим более подробно тег `<input>`.

```
<input type="text" size="30" placeholder="Ваше Имя">
<input type="password" size="8">
<input type="checkbox" checked="checked">
<input type="radio" name="radio">
<input type="file">
<input type="submit" value="Сохранить">
<input type="reset" value="Очистить">
<input type="button" value="просто кнопка">
```

У первого тега `<input>` значение атрибута `type="text"`, значит, это обычное однострочное поле ввода. В значении атрибута `size` указывается размер этого поля в символах. Атрибут `placeholder` выводит текст внутри текстового поля, который исчезает при начале ввода символов.

`<input>` со значением `password` выглядит так же, как и предыдущий. Но он используется для ввода пароля, и все символы будут показываться как точки, чтобы пароль нельзя было подсмотреть.

Следующие два варианта — `<input type="checkbox">` и `<input type="radio">` — предоставляют пользователю возможность выбора некоторых пунктов.

В первом случае есть возможность множественного выбора нескольких пунктов в виде флажков, а во втором — единственного выбора в виде переключателей. В обоих случаях можно отметить любой пункт по умолчанию, если задать атрибут `checked="checked"`.

`<input type="file">` предоставляет пользователю возможность выбора файла на своём компьютере, чтобы загрузить его на сервер.

Оставшиеся 3 тега `<input>` — кнопки. `submit` отправляет данные на сервер, `reset` предназначена для очистки всех полей в форме. А третья `type="button"` — просто кнопка, при нажатии на которую ничего не произойдёт. Но её, например, можно привязать к событию `javascript`. У всех кнопок в значении атрибута `value` указывается текст, который будет отображаться на кнопке.

Тег `<label>`

Тег `<label>` устанавливает связь между определенной меткой, в качестве которой обычно выступает текст, и элементом формы (`<input>`, `<select>`, `<textarea>`). Такая связь нужна, чтобы изменять значения элементов формы при нажатии курсором мыши на текст.

С помощью `<label>` можно устанавливать горячие клавиши на клавиатуре и переходить на активный элемент подобно ссылкам.

Существует два способа связывания объекта и метки. Первый заключается в использовании идентификатора `id` внутри элемента формы и указании его имени в качестве атрибута `for` тега `<label>`. При втором способе элемент формы помещается внутрь контейнера `<label>`.

```
<!-- Первый способ -->
<input id="идентификатор" type="checkbox">
<label for="идентификатор">Текст</label>
<!-- Второй способ -->
<label><input type="checkbox">Текст</label>
```

Задание:

Выбрать тематику интернет-магазина.

Создание основного наполнения сайта, главной страницы, каталога и товара каталога:

Создать файл `index.html` — это будет главная страница. На этой странице сделайте ссылки в виде маркированного списка на следующие страницы.

1. Меню (должно располагаться на каждой странице сайта в виде списка ссылок):
 - Главная
 - Каталог
 - Регистрация
2. Отделить горизонтальной чертой шапку сайта.
3. Добавить название вашего магазина.
4. Напишите приветственное сообщение.
5. Горизонтальная черта.
6. Подвал (должен располагаться на каждой странице сайта, содержать значок копирайта и текст «Все права защищены»).

На странице «Каталог»

1. Меню сайта.
2. Горизонтальная черта.
3. Заголовок «Каталог».
4. Уменьшенные копии изображений товаров из вашего магазина (оформить списком).
5. Ссылки под картинками, для перехода в подробное описание продукта.

6. Подвал.

На странице «Товар каталога»

На странице должны присутствовать:

- Меню сайта
- Горизонтальная черта
- Заголовок «Название продукта»
- Подзаголовок «Описание продукта». Текст подробного описания продукта.
- Подзаголовок «Характеристики продукта». Текст характеристик продукта.

1. Добавить картинку продукта с возможностью нажатия на нее. Картинка должна открыться в полном размере, в новом окне.

2. В подразделе «Характеристики продукта» необходимо добавить маркированный или нумерованный список характеристик товара.

На странице «Регистрация»

На странице должны присутствовать:

- Меню сайта
- Горизонтальная черта
- Заголовок «Регистрация на сайте»
- Форма с полями:
 - a. ФИО;
 - b. телефон;
 - c. электронная почта;
 - d. логин;
 - e. пароль
 - f. кнопки «Отправить», «Отменить».