

Лабораторная работа 1. «Основы HTML 5»

Теория

Структура HTML-документа определяет базовый «скелет» для будущего сайта. В HTML5 введены новые семантические элементы которые позволяют лучше структурировать страницу сайта и придать каждому блоку определенный смысл, рассмотрим структуру сайта по стандарту HTML5

```
<!DOCTYPE HTML>
<html lang=«ru»>
<head>
<meta charset=«UTF-8»>
<title>Название страницы</title>
<meta name=«description» content=«Описание страницы» />
<link rel=«stylesheet» type=«text/css»
href=«css/style.css» />
<link rel=«icon» href=«/img/logo.jpg»>
<script type=«text/javascript» src=«code.js»></script>
<!--[if lt IE 9]>
<script
src=«http://html5shiv.googlecode.com/svn/trunk/html5.js
»></script>
<![endif]-->
</head>
<body>
<header>
Шапка
<nav>
Навигация
</nav>
</header>
<main>
<section>
<h1>Заголовок страницы</h1>
Основная часть
</section>
</main>
<aside>
Боковая колонка (сайдбар)
</aside>
<footer>
Подвал
</footer>
```

```
</body>  
</html>
```

Разберем основные теги и атрибуты, примененные в примере:

<!DOCTYPE HTML> – указание браузеру, как интерпретировать данную страницу, к какой версии принадлежит HTML-документ. В нашем случае подразумевается стандарт HTML5.

<html lang="ru"> – контейнер для всего содержимого страницы с указанием принадлежности сайта к конкретному языку.

<head> – технический контейнер, направленный на составление информации о странице, подключения стилей и скриптов.

<meta charset="UTF-8"> – кодировка сайта.

<title> – название страницы. Придерживайтесь правильного заполнения тега Title.

<meta name="description" content="" /> – мета-описание документа. Опять же, существуют некоторые правила грамотного заполнения метаданных.

<link rel="stylesheet" type="text/css" href="" /> – подключение файла каскадных таблиц стилей (CSS) для стилизации и оформления элементов.

<link rel="icon" href=""> – иконка сайта, отображаемая в браузере.

<script type="text/javascript" src=""> – подключение внешней библиотеки JQuery последней версии.

<![if lt IE 9]><script src=""> – условие, при котором посетитель, зашедший с браузера Internet Explorer 9, получит поддержку новым тегами, описанным в HTML5.

<body> – тело документа, включающее теги, атрибуты, текст и изображения. Отображается в браузере.

<header> – шапка сайта.

<nav> – навигация по ресурсу непосредственно в шапке. Может быть также использован как в aside, так и в footer.

<main> – основной контент документа.

<section> – секция для одного типа содержания. Допустимо наличие несколько одноименных тегов.

<h1> – заголовок страницы.

<aside> – боковая колонка, также именуемая как «сайдбар».

<footer> – подвал сайта.

HTML5-формы

HTML-формы являются элементами управления, которые применяются для сбора информации от посетителей веб-сайта.

До появления HTML5 веб-формы представляли собой набор нескольких элементов `<input type="text">`, `<input type="password">`, завершающихся кнопкой `<input type="submit">`. Для стилизации форм в разных браузерах приходилось прилагать немало усилий. Кроме того, формы требовали применения JavaScript для проверки введенных данных, а также были лишены специфических типов полей ввода для указания повседневной информации типа дат, адресов электронной почты и URL-адресов.

HTML5-формы решили большинство этих распространенных проблем благодаря наличию новых атрибутов. С выходом **HTML 5** к элементу **<input>** добавилось *12 новых типов* (видов) полей, но к сожалению, пока не все из них имеют полную поддержку всеми передовыми браузерами.

Основу любой формы составляет элемент `<form>...</form>`. является контейнером, удерживая вместе все элементы управления формы – **поля**.

Элемент **<input>** является **основным элементом формы** и определяет пользовательское поле для ввода информации. Поле ввода принимает различный вид, в зависимости от значения атрибута **type**, применённого к данному элементу.

Самыми часто используемым видам полей относятся те, которые были введены задолго до **HTML 5**, например, такие как:

text (однострочное текстовое поле)

checkbox (флаговая кнопка)

radio (радиокнопка)

submit (кнопка отправки формы)

Для визуальной группировки элементов, размещенных внутри формы служит тег `<fieldset>`. Во всех основных браузерах, данный элемент выделяется рамкой, а содержимое элемента получает внутренние отступы со всех сторон.

Тег `<legend>` определяет заголовок для `<fieldset>` элемента, с помощью которого группируются элементы формы.

Пример использования:

```
<form>
  <fieldset>
    <legend>Пожалуйста, представьтесь</legend>
    Имя: <input type = "text" name = "firstname"
    value = "Введите имя"> <br>
    Фамилия: <input type = "text" name = "lastname" Required > <br>
    Мужчина <input type = "radio" name = "sex"
    value = "male" checked><br>
    Женщина <input type = "radio" name = "sex" value = "female"><br>
    <input type="submit" name = "submit" value = "отправить">
  </fieldset>
</form>
```

В этом примере мы:

- Разместили внутри формы (парный тег `<form>`) элемент `<fieldset>`, благодаря которому мы группируем элементы формы, вокруг формы появляется рамка (внешний вид рамки может отличаться в зависимости от браузера). Тегом `<legend>` мы определяем для нашей рамки заголовок "Пожалуйста, представьтесь".
- Разместили два однострочных текстовых поля (`<input type = "text">`) для ввода пользователем своего имени и фамилии. Присвоили этим полям уникальные имена атрибутом **name** (для корректной передачи и обработки формы обязательно указывайте для каждого элемента `<input>` этот атрибут). Кроме того атрибутом **value** установили для этих полей значения по умолчанию (при заполнении полей значение этого атрибута будет соответствовать пользовательскому вводу).
В поле «Фамилия» задали атрибут **Required** который определяет, что поле обязательно для заполнения. Если в текстовой области не будет содержимого, при попытке отправки формы на сервер веб-браузер выведет сообщение о необходимости ее заполнения, а форма отправлена не будет. Содержание сообщения зависит от браузера и не может быть изменено.
- Разместили две радиокнопки (`<input type = "radio">`) для выбора одного из ограниченного числа вариантов. Обратите внимание, что для радиокнопок необходимо указывать одинаковое имя, чтобы была возможность выбрать только один вариант из предложенных.
- Заключительный элемент это кнопка, которая служит для отправки формы (`<input type = "submit">`).

Для работы с датами и временем в HTML5 предназначено несколько типов элементов input:

- `datetime-local`: устанавливает дату и время

- date: устанавливает дату
- month: устанавливает текущий месяц и год
- time: устанавливает время
- week: устанавливает текущую неделю

Пример:

```
<form>
<p>
<label for="week">Неделя: </label>
<input type="week" name="week" id="week" />
</p>
<p>
<label for="localdate">Дата и время: </label>
<input type="datetime-local" id="localdate" name="date"/>
</p>
<p>
<label for="month">Месяц: </label>
<input type="month" id="month" name="month"/>
</p>
<p>
<label for="time">Время: </label>
<input type="time" id="time" name="time"/>
</p>
<p>
<button type="submit">Отправить</button>
</p>
</form>
```

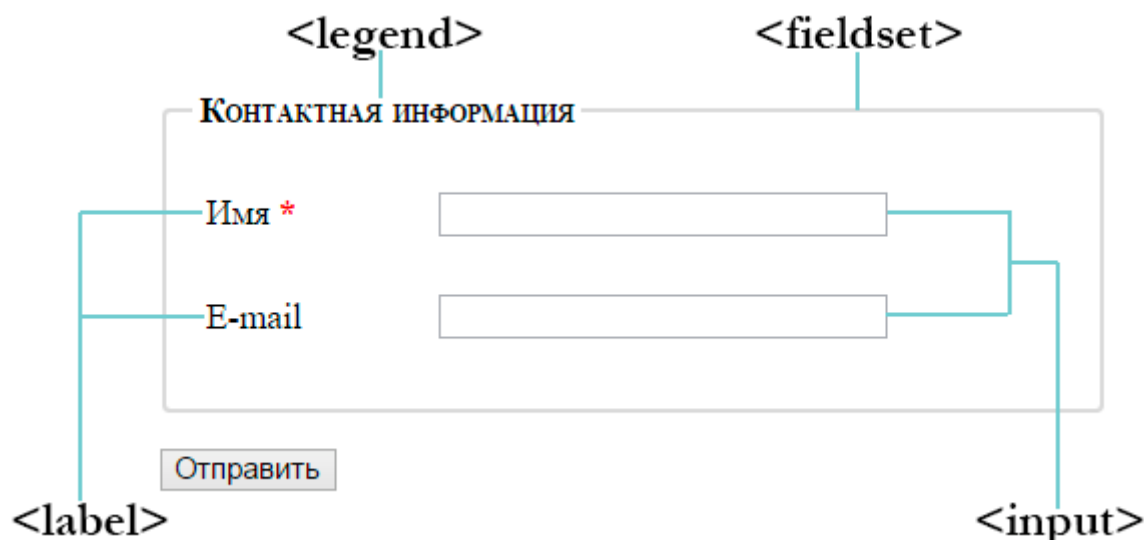
Внешний вид этих элементов может отличаться в разных браузерах

Задание 1

Создать страницу Вашего резюме и форму обратной связи согласно стандарту html5.

1. Обязательные параметры резюме: использование списков,

- Ваша фотография;
- ФИО полностью;
- группа;
- образование;
- опыт работы;
- интересы;
- контактная информация: электронная почта, телефон (Использовать группу элементов присвоив название с помощью элемента <legend>)



2. Создать страницу «Давайте познакомимся», для оформления формы используйте группу элементов с заголовком.

В форме «Давайте познакомимся» предусмотреть поля:

- ФИО (обязательное поле);
- Телефон;
- Электронная почта (обязательное поле);
- Дата рождения;
- образование (предопределенные поля с выбором радиокнопка: основное общее (5—9 класс), среднее общее (10 и 11 классы), среднее профессиональное, бакалавриат, специалитет, магистратура)
- Сообщение (поле для ввода текста)
- Кнопка «Отправить»

3. Обязательные параметры стилевых таблиц:

- Заголовок (задать цвет, размер шрифта),
- Пункты «Опыт работы», «Образование», «Интересы» должны содержать не менее 3 предложений.
- Для всего текста должны быть заданы цвет, выравнивание и размер шрифта.

Задание 2

Разместите сайт на хостинге в сети Интернет. Можно использовать любой хостинг. Рассмотрим как разместить полноценный сайт на примере GitHub с бесплатным хостингом

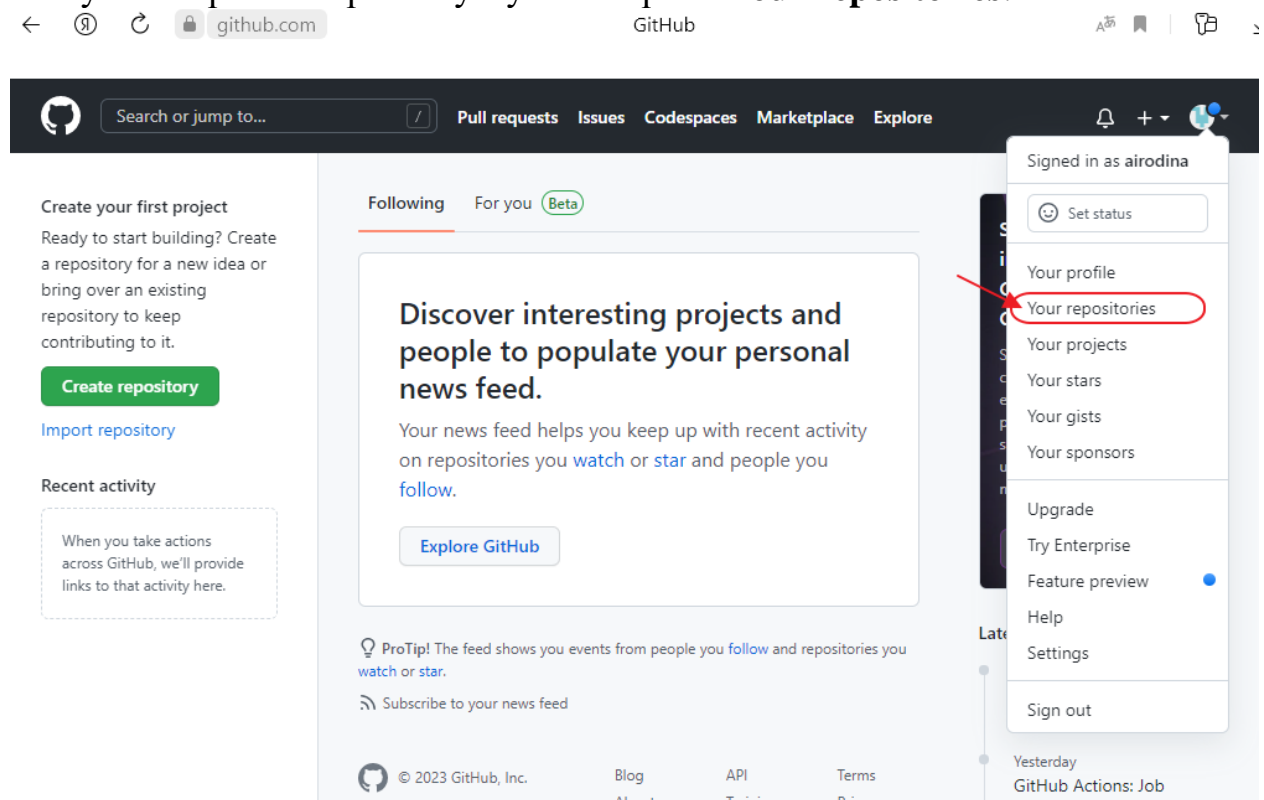
Ход выполнения работы:

Регистрируемся на GitHub (<https://github.com/>) или заходим под своим логином и паролем.

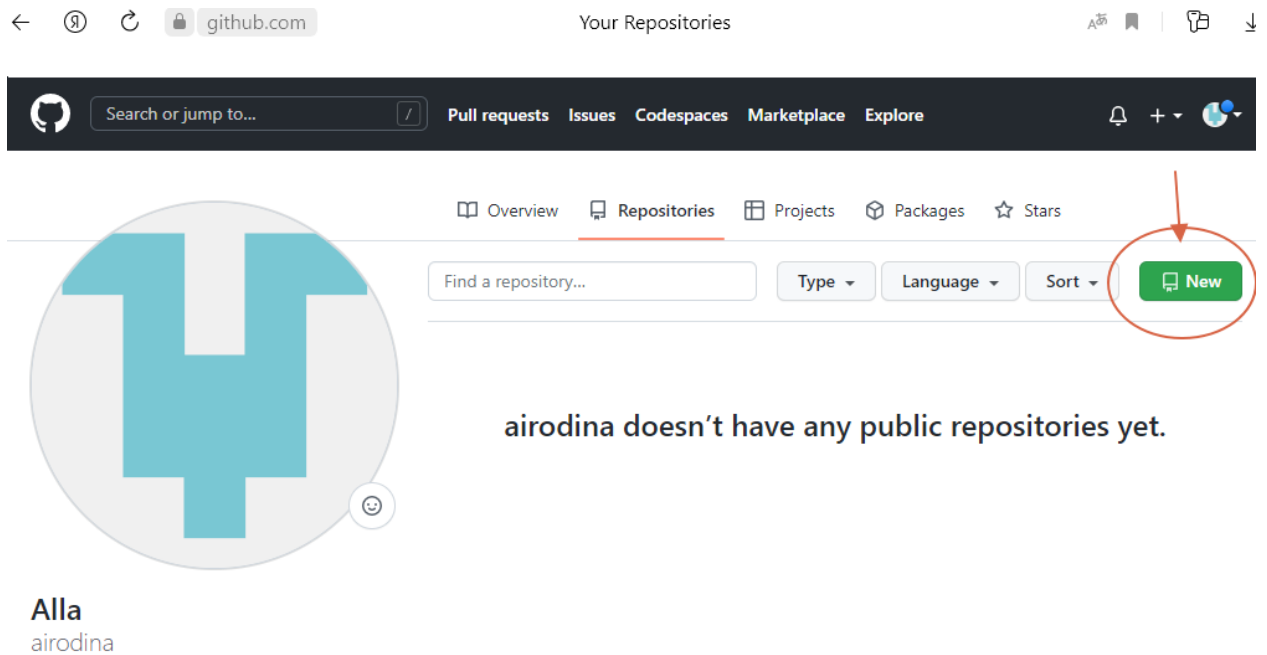
Создаём репозиторий.

Репозиторий — это место, где будет лежать исходный код вашего сайта. Можно сравнить его с файловой системой на компьютере, только вместо личных данных — HTML-разметка, скрипты, картинки и другое содержимое веб-страницы.

Для начала — откройте раздел с репозиториями. Нажмите на иконку аккаунта в правом верхнем углу и выберите **Your repositories**:



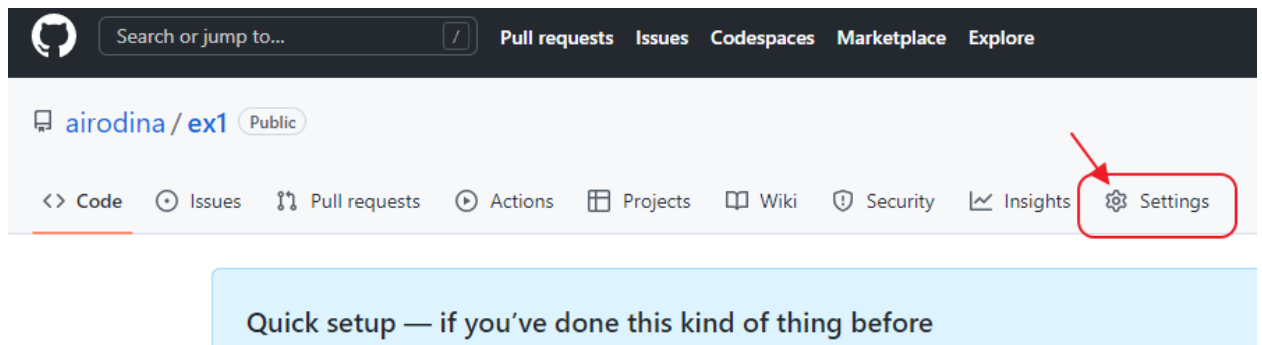
Чтобы добавить новый репозиторий, нажмите кнопку **New** в правом верхнем углу:



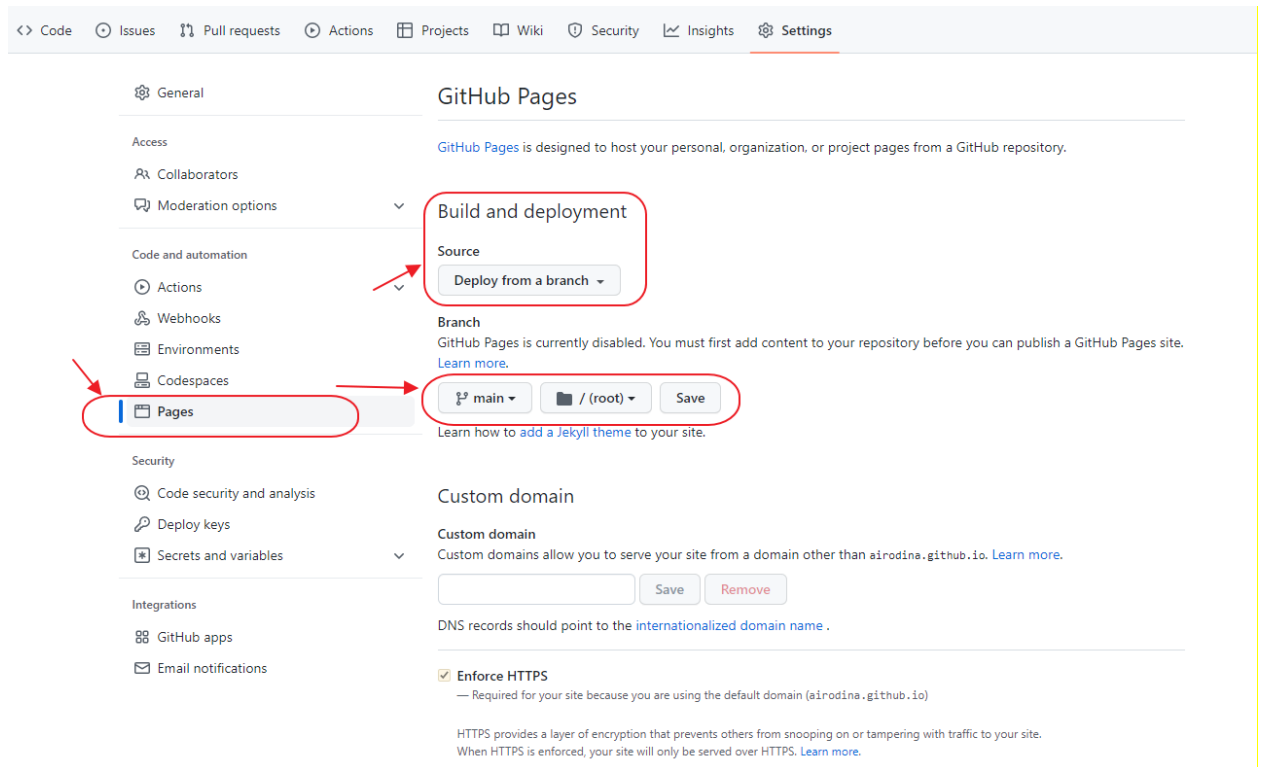
Задайте базовые настройки будущего сайта: имя, описание и видимость в интернете.

Подключаем GitHub Pages

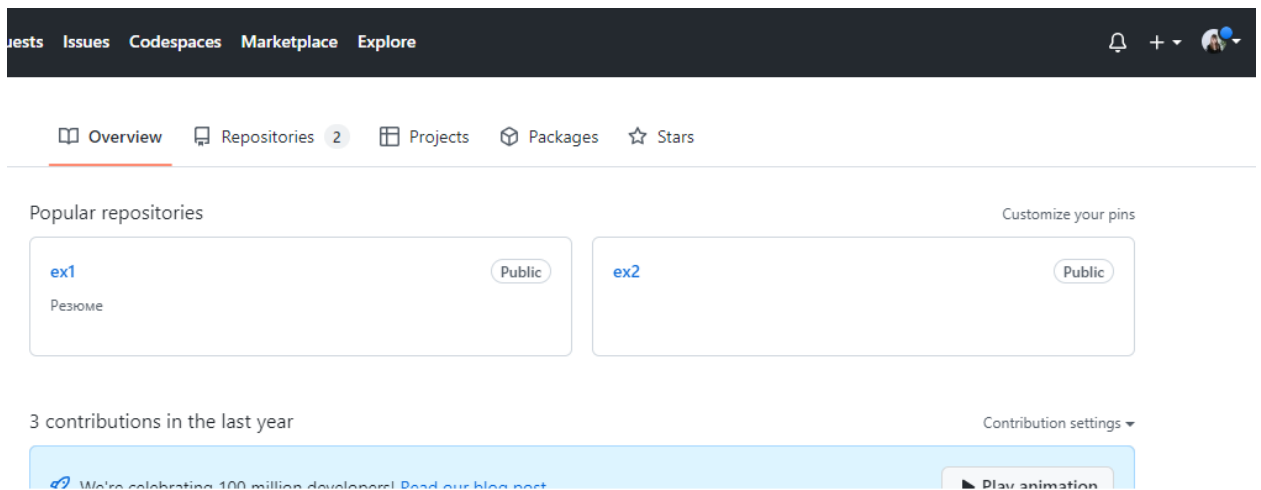
Откройте проект и зайдите в настройки:



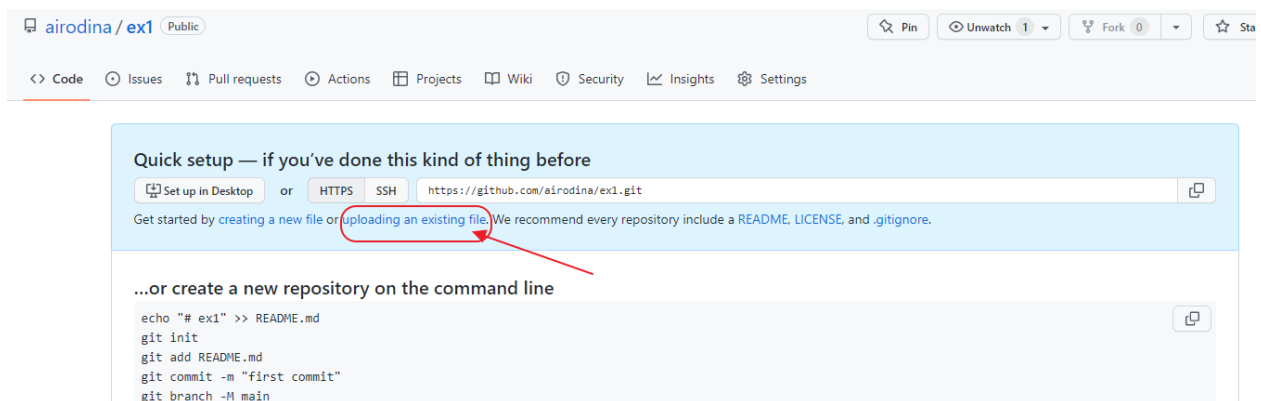
В меню слева выбираем раздел **Pages**. Затем переходим к пункту **Branch** и меняем значение **None** на **Main** — это значит, что сайт будет собираться из главной ветки репозитория. После этого нажмите **Save**.



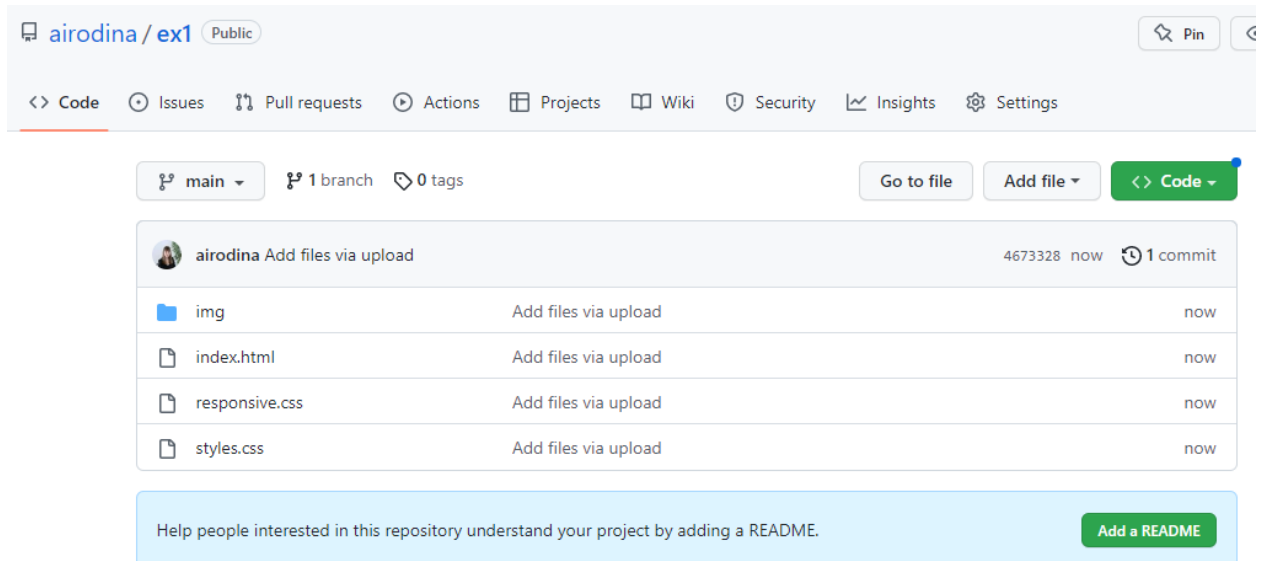
Ссылка на проект отображается на главной странице.



Загрузите все файлы вашего проекта.



Добавьте все файлы и папки вашего проекта. При этом файл index.html должен находиться в корневой директории проекта:



Подтвердите публикацию на GitHub Pages. В настройках репозитория: Settings -> Секция Pages.

airodina / airodina.github.io Public

Code Issues Pull requests Actions Projects Wiki Security Insights **Settings**

General

Access

Collaborators

Moderation options

Code and automation

Branches

Tags

Actions

Webhooks

Environments

Codespaces

Pages

GitHub Pages

GitHub Pages is designed to host your personal, organization, or project pages from a GitHub repository

Your site is live at <https://airodina.github.io/>
Last deployed by airodina 29 minutes ago

Visit site

Build and deployment

Source

Deploy from a branch

Branch

Your GitHub Pages site is currently being built from the main branch. [Learn more.](#)

main / (root) Save

Learn how to [add a Jekyll theme](#) to your site.