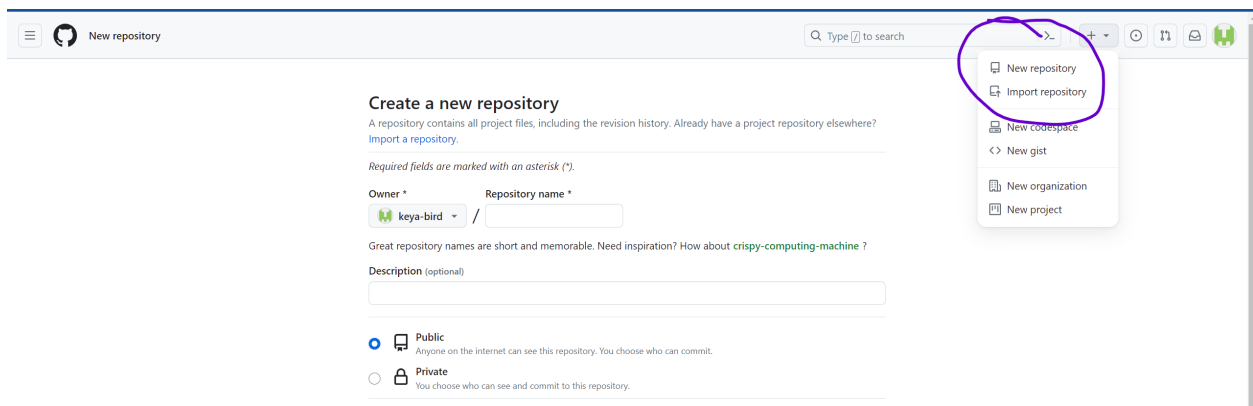# Creating a Personal Website on GitHub Step by Step Instructions

For first time coders needing a step-by-step guide in creating a simple website or page on GitHub.

## Step 1) Creating the Repository

Create a new repository by going to Github.io and selecting the drop down in the upper right-hand corner see figure 1.0 below:



**Figure 1.0**

Next create a *new* repository and the domain name must include your username as shown below in figure 1.1 on the next page:

# Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? Import a repository.

*Required fields are marked with an asterisk (\*).*

**Owner \***                  **Repository name \***

[🐦 keya-bird ▾] / [ keya-bird.github.io ]

✅ keya-bird.github.io is available.

Great repository names are short and memorable. Need inspiration? How about **verbose-guacamole** ?

**Description** (optional)

[                                                                ]

⦿ 📖 **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:**
☑ **Add a README file**
   This is where you can write a long description for your project. Learn more about READMEs.

**Add .gitignore**
[ .gitignore template: None ▾ ]
Choose which files not to track from a list of templates. Learn more about ignoring files.

**Choose a license**
[ License: None ▾ ]
A license tells others what they can and can't do with your code. Learn more about licenses.

This will set ⑂ main as the default branch. Change the default name in your settings.

ⓘ You are creating a public repository in your personal account.

[ **Create repository** ]

**Figure 1.1**

**Repository Fields:**

-Make sure the repository is set to public.

-Add the readme section.

# Step 2) Check the Repository Settings

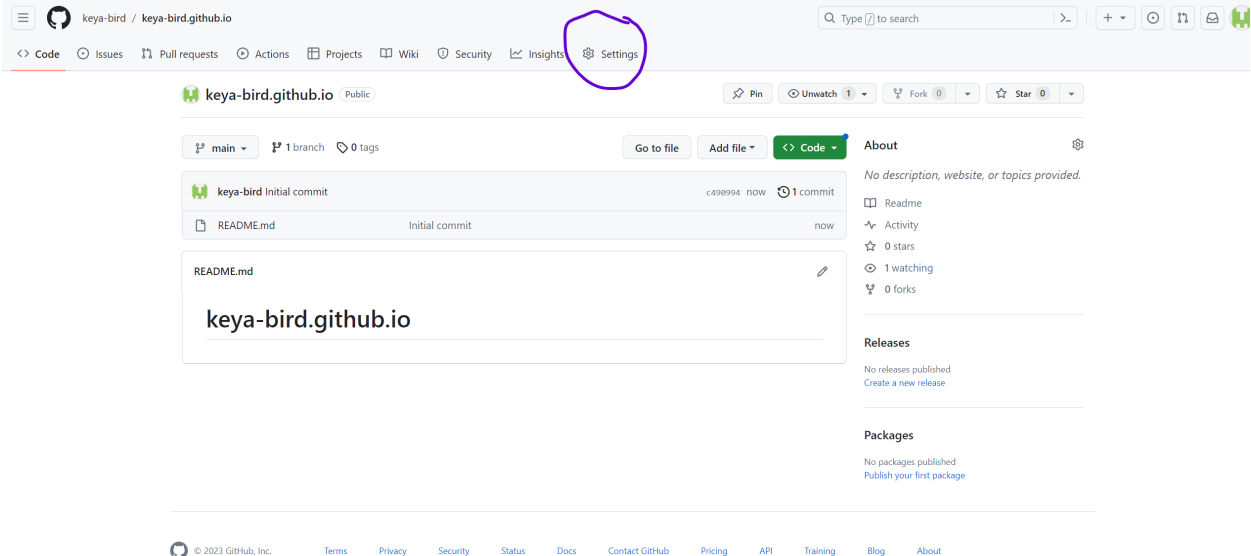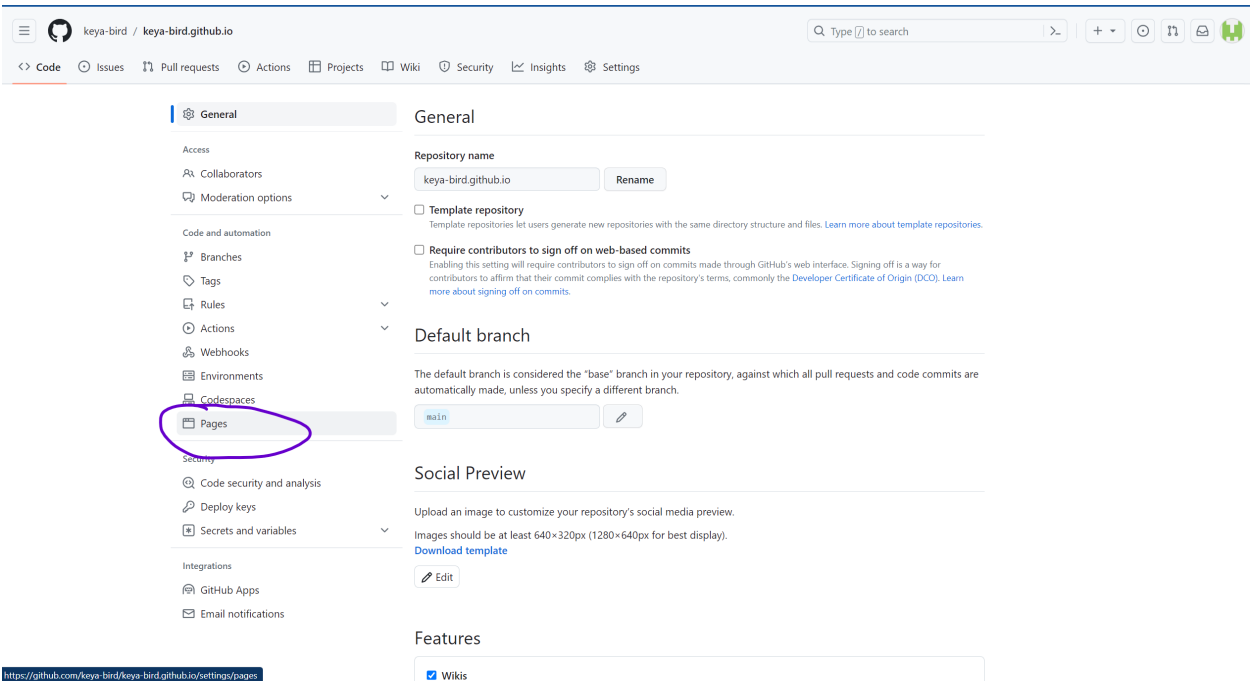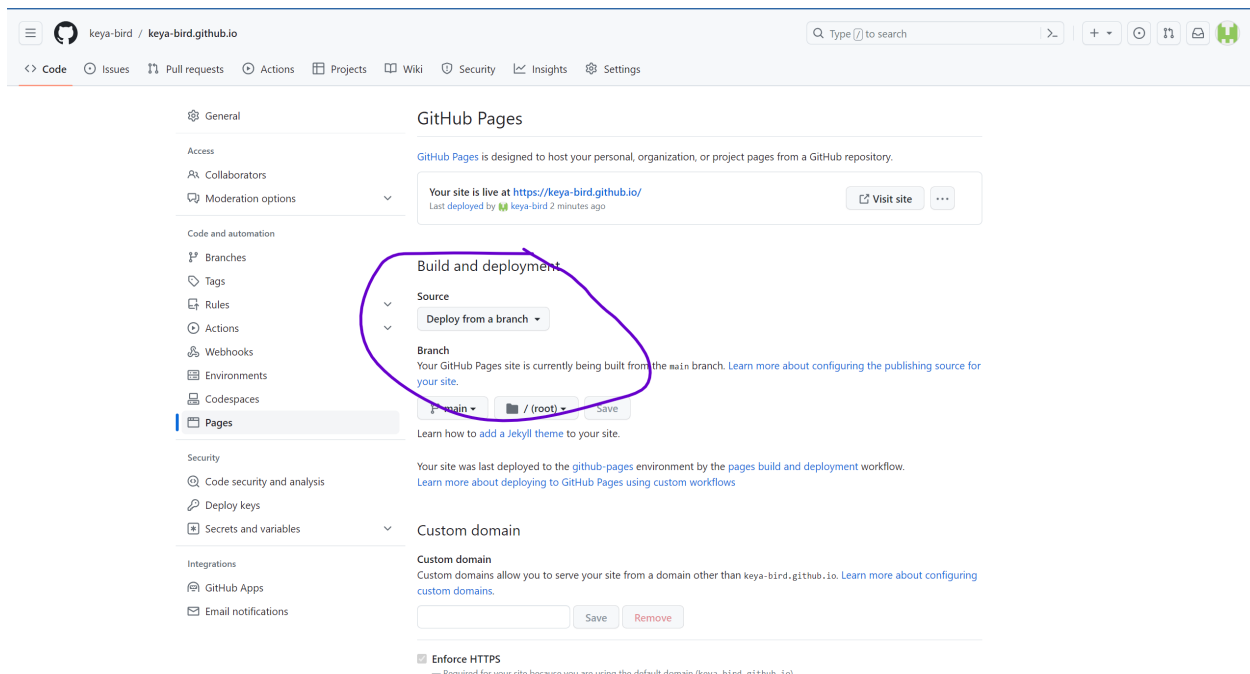Make sure to check the settings by selecting the settings sections see below:



**Figure 2.0**

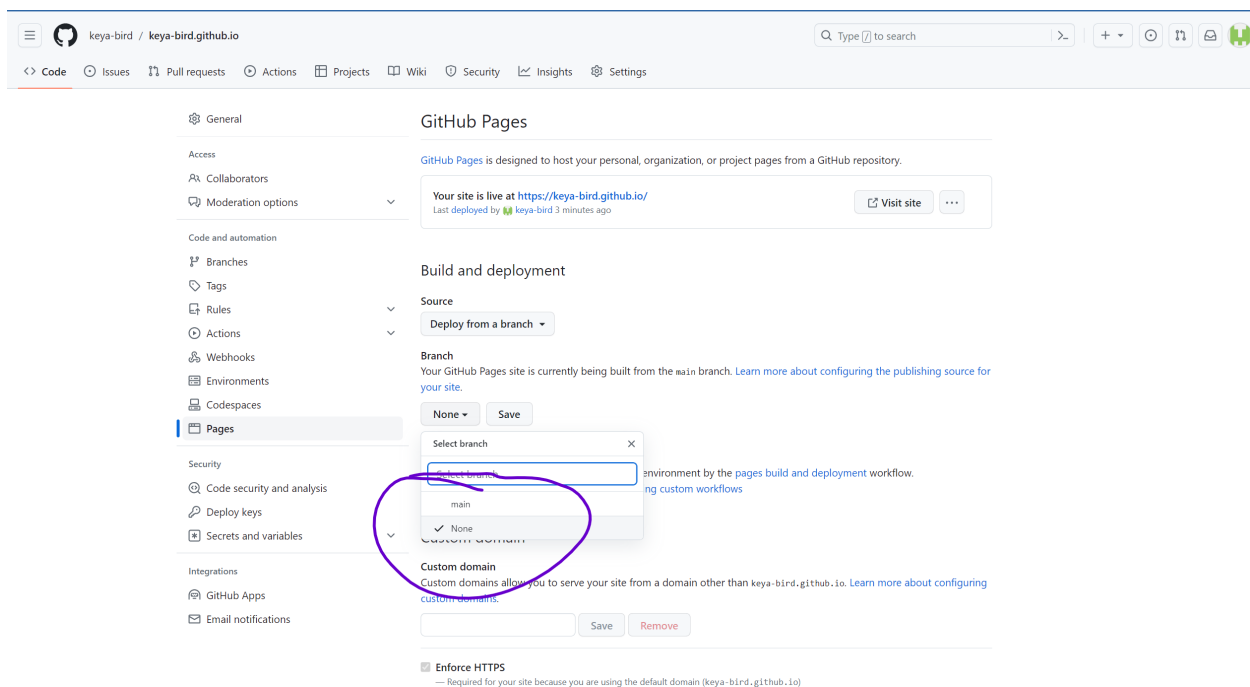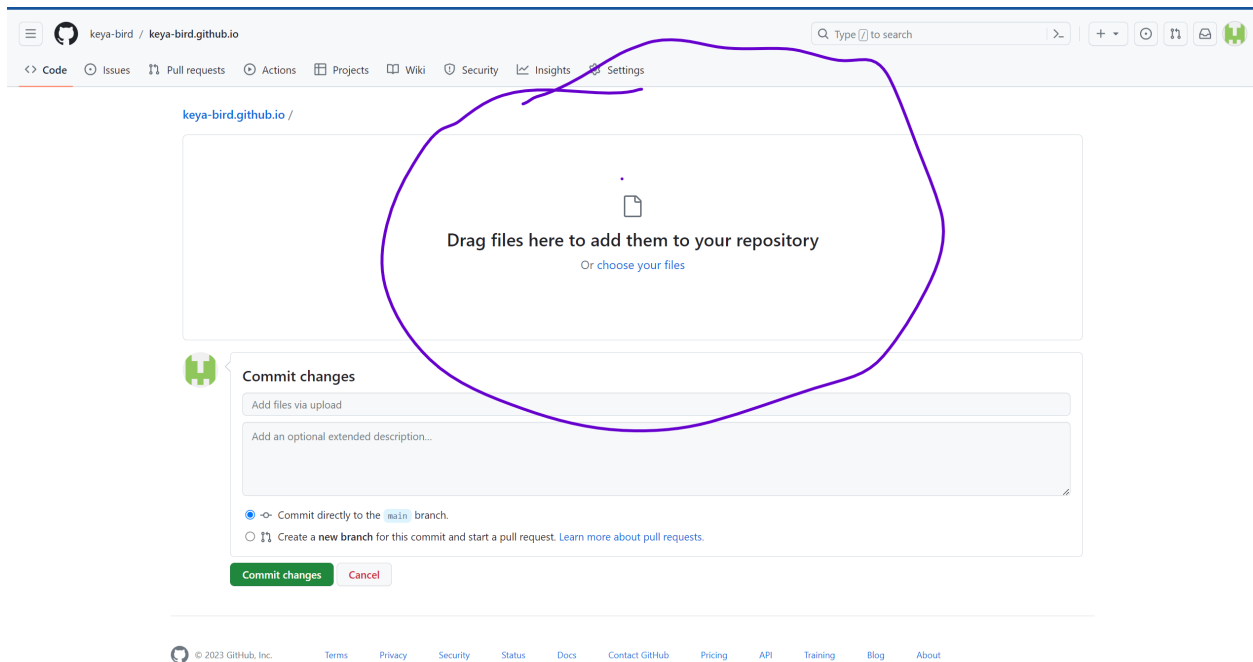Next select the pages section in the lower left see below:



**Figure 2.1**

Figure 2.2 under Build and deployment make sure to select the drop down and choose the 'none' option. See figure 2.3 below.

Figure 2.3

## Step 3) Add your Project Files to the Repository

The simplest way to achieve this would be to create your website or page locally and then drag and drop your project files.  See figure 2.0 below:



**Figure 2.0 Once you've dragged and dropped your files you can select the Commit changes button.**

## Step 4) Test Your Site by Navigating to your Domain

Finally, you can view your changes by going to your domain link which in my case is

https://keya-bird.github.io/ but yours will be whatever you named your repository in the first step.