**Creating a Website on GitHub Pages**

**Put your work online using GitHub Pages – no command line required!**

[GitHub Pages](https://pages.github.com/), a tool provided by [GitHub](https://github.com/), lets you easily create and deploy a website online. GitHub allows you to store all the files and code for your website in a repository. You can then use GitHub Pages to generate a personal URL and share your site with friends, family, and the world!

For this tutorial, you need a GitHub account and basic HTML knowledge. You don’t need to know Git or the command line.

**GitHub vs. GitHub Pages**

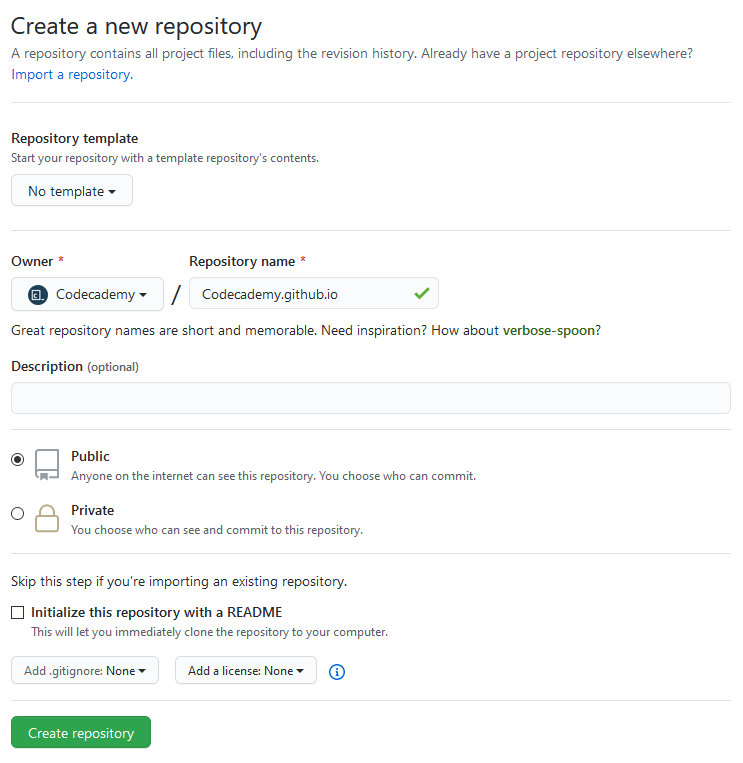
In GitHub, code is stored in a *repository*, or repo. You can think of it as a special folder that lives online. GitHub hosts that folder so that it is accessible to you and your teammates from anywhere in the world.

In our case, our repository will contain HTML and CSS code that defines a website. But you can’t SEE the website anywhere. GitHub just displays the code like any other text. GitHub Pages essentially connects your repository to a unique URL so that, when you go to that URL – say **username.github.io** – in your browser, you’ll find your website displayed.

**Create a Repository**

Let’s get started!

[Create a new repository](https://github.com/new) named **username.github.io**, where **username** is your GitHub username.

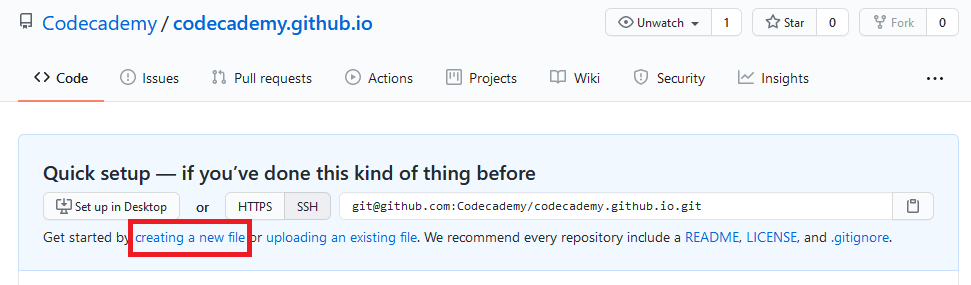


You have the option here to use a template, add a description, make your repository private, and initialize some additional files. These items make it easy for other developers to learn about your repository: what it’s about, who can access it, who can use the code, etc. For simplicity, we’re going to skip those items for now. Your site will work without them.

Click “Create Repository”.

**Creating Your First Page**

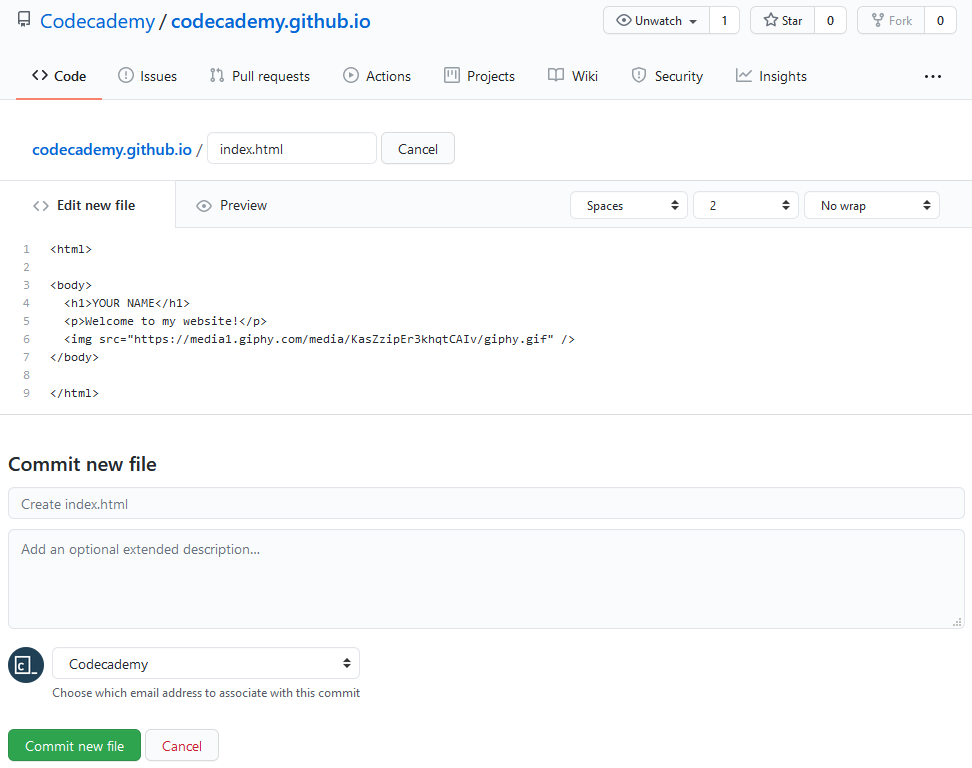
You should be redirected to the repository page, which shows an empty repository. It’s time to add a file! On that page, click the “creating a new file” link.



Name your file **index.html**. Now copy and paste the following code into the file.

<html>  
   
<body>  
  <h1>YOUR NAME</h1>  
  <p>Welcome to my website!</p>  
  <img src="https://content.codecademy.com/articles/github-pages-via-web-app/happy-ice-cream.gif" />  
</body>  
   
</html>

You can customize the text that says **YOUR NAME** and **Welcome to my website!**.



Changes to your repository are called *commits*. You can imagine each commit as a snapshot of your repository at different times.

Time to create our first commit! In the first textbox, name your commit “Create index.html”. Commits usually start with a verb describing changes made to the repository.

Press the “Commit new file” button.

**Viewing Your Website**

Your website is now accessible at **https://username.github.io**! (Remember to replace **username** with your GitHub username.)



If you’d like, you can develop your website by adding more HTML and CSS! Your site should update a few minutes after each commit.

**Troubleshooting**

If your site isn’t appearing after 10 minutes, try these additional steps:

1. Within your **username.github.io** repo, go to Settings and scroll down to GitHub Pages. Under Source, make sure that your repo is linked to your web page. If you see **None**, then select the **main** branch.
2. Within the same GitHub Pages section, go to Theme Chooser and select any theme.

