



Getting Started with GitHub: Part 1 -- Best practices for setting up your GitHub account



MOD

beardofedu

Moderator



Creating a username

To begin using GitHub, you should create an account so you can participate in open source projects, work with friends on a school project, or just explore a new tool that you overheard some people talking about at work. Creating an account requires three things:

- A unique username, or handle
- An email address
- A password

Your name is important. It differentiates you from millions of other people. So too, is your GitHub handle. That might sound a little heavy, but, think about this: Your GitHub handle can be used to view your contributions to public projects hosted on GitHub. Your

interactions with other people on GitHub are introduced with your avatar and handle. Selecting a handle that identifies you for the amazing person you are but can potentially survive the test of time can be hard. Thankfully, that is what I'm here for!

Be you

When selecting your handle, it is very important to be true to yourself and who you are as an individual. Keep in mind, you might be listing your handle on your resume in the future and recruiters might be contacting you because of the work you do on GitHub. Having a handle that doesn't raise questions about your personality or how you mesh with different members of a team is a great way to gauge if you are selecting a great handle.

Another way to view your handle is that you are going to be growing with it, creating a handle like "NoobCoderForever" might be funny as you are just starting out, but as you gain experience you might look back and wish you hadn't picked that handle. Perhaps you really like a band right now, and want to create a handle that reflects your love for their music. While you might honestly love Cardi B right now, you might not in 2 years ... or 3 months.

I would recommend avoiding trending topics for a user handle. Sometimes a trend will last, but, "FidgetMaster" or "SupremeSpinner", might not make sense in the future. Think about things that make you unique, maybe a hobby you really enjoy, or something that so far has weathered the test of time and try and go with that.

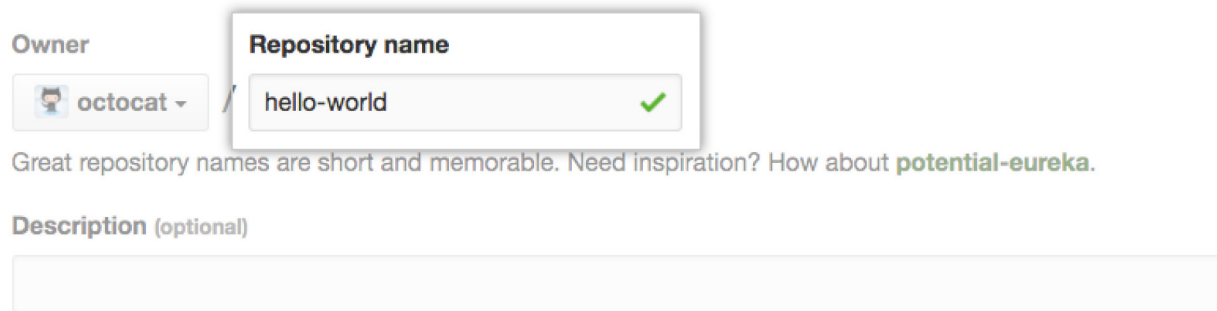
At the end of the day, your GitHub handle will become your calling card when interacting with repositories. Other contributors will have to reference your name when asking you a question, assigning you a task, or commenting on the contribution you made. So, think about what you want your calling card to be, not necessarily for the next 15 weeks but for the next 15 years.

Creating a repository

With a username selected, you can now create a repository. Repositories require a project or repository name, and you can enter an optional description of the repository if you want. The description and the repository name make it easier for people to find your project, so if you are working on a JavaScript emulation of Tetris, naming it something like "Javascript-Tetris" and providing a description like "A Javascript emulation of Tetris", would make it easier for other people interested in creating Javascript emulations of Tetris to find your project.

Create a new repository

A repository contains all the files for your project, including the revision history.



Owner

octocat /

Repository name

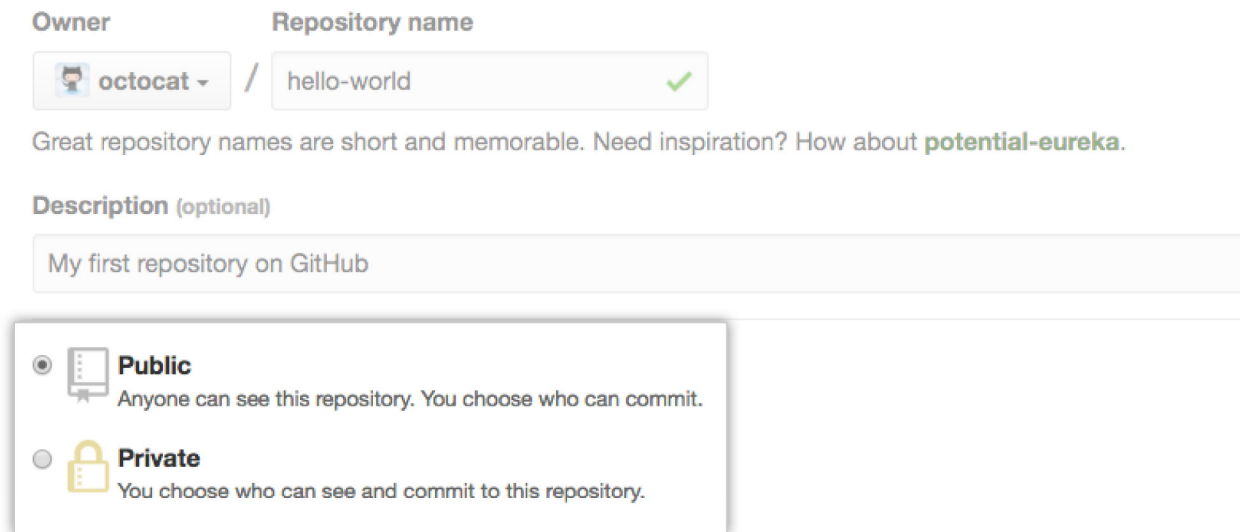
hello-world ✓

Great repository names are short and memorable. Need inspiration? How about **potential-eureka**.

Description (optional)

Public or private repositories?

After entering a repository name and providing the optional description, you are presented with two repository publication methods: **Public** and **Private**. Public repositories, are just that, public. They are open and can be viewed by other contributors on GitHub. This allows other individuals interested in your project to hopefully find your project and either use it or hopefully, contribute to the future development of it. Private repositories on the other hand are hidden from the general public of GitHub and future contributors will need to be invited to the repository before they can see it, let alone contribute to it.



The screenshot shows the GitHub repository creation interface. At the top, there are two input fields: 'Owner' with a dropdown menu showing 'octocat' and a 'Repository name' field containing 'hello-world' with a green checkmark. Below these is a hint: 'Great repository names are short and memorable. Need inspiration? How about **potential-eureka**.' Underneath is a 'Description (optional)' text area containing 'My first repository on GitHub'. At the bottom, there is a section for repository visibility with two radio buttons. The 'Public' option is selected, accompanied by a monitor icon and the text 'Public: Anyone can see this repository. You choose who can commit.' The 'Private' option is unselected, accompanied by a lock icon and the text 'Private: You choose who can see and commit to this repository.'

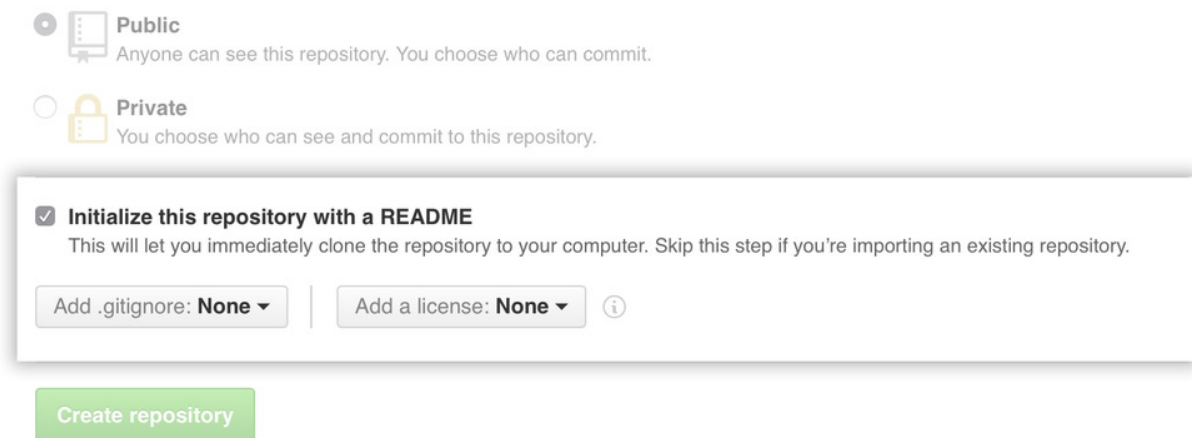
While you might be inclined to create a private repository, unless you are working on a project that you don't want individuals to help contribute on (like software you aren't ready to open source), it doesn't hurt to create a public repository. You can always change the privacy settings of your repository later (either making a private repository public, or vice versa).

For the example of creating a repository for your JS Tetris emulation, creating the repository as *Public* makes sense. It enables other GitHub users (and people not on GitHub for that matter), to find your project and potentially use it. So, even if this Tetris game is just an opportunity to follow a tutorial on the internet, others might find your project and comment on your code through comments on Pull Requests or creating Issues to identify what you could change or improve to make your emulation better.

Initialize with a README.md?

Before you click the **Create repository** button there is one last option I want to discuss: **Initialize this repository with a README**. To sum up why this option is important quickly, having a commit (or snapshot of the state of your project at a given time) created when you first create your repository enables you to begin working on the repository immediately. This is a great option if you are starting from scratch or don't already have a Git project that you want to import.

If you are planning on importing a Git project that is already on your computer, you shouldn't select the checkbox, because you will encounter a conflicting history issue when trying to send your local Git project to your shiny new GitHub repository.



The screenshot shows the GitHub repository creation interface. At the top, there are two radio button options: 'Public' (selected) and 'Private'. Below these is a section titled 'Initialize this repository with a README' which is checked. Underneath this section are two dropdown menus: 'Add .gitignore: None' and 'Add a license: None'. At the bottom of the form is a green 'Create repository' button.

☒ **Public**
Anyone can see this repository. You choose who can commit.

☐ **Private**
You choose who can see and commit to this repository.

☒ **Initialize this repository with a README**
This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: **None** | Add a license: **None** ⓘ

Create repository

The README file is a super special file on GitHub repositories, so, I definitely recommend creating one, even if it is after you create your repository and import your existing Git project. The README file is displayed on the **Code** tab of your repository automatically and kind of acts like a billboard of important information related to your project. Potential contributors to projects typically review a project's README file to get an understanding of what the project is, who to contact, and most importantly, how to contribute.

This "billboard" is your opportunity to provide visitors with helpful information about your project. If you have installation instructions, including them on the README is definitely recommended. As a markdown file, the README supports displaying images, inline links, and most importantly, GIFs and emoji. Other information that is helpful to include in your project's README file includes points of contact (or who should a user at-mention) in an Issue or Pull Request if they encounter an issue or have a suggestion. If you have a planning document for the fixes and enhancements you are planning on implementing in your project, including a link to that in the README is also a great way to provide information to a potential contributor.

The README file can also include "badges" from integrations like CircleCI, TravisCI, Appveyor, Slack, and many more. Displaying those integrations lets potential contributors know what integrations your project uses and if the current build is passing or has any issues.

Like most aspects of your project, the README file isn't written in stone, so it can grow and change as your project starts as a JS Tetris emulation and evolves into a multiplayer VR Tetris game where people play as the blocks!

Welcome to GitHub!

In closing, welcome to GitHub. Whether you came here from a resolution you made, or just wanted to stop lurking and create an account, welcome! I'm sure your new handle is as awesome as you are and your first repository is looking ready to grow with you commit after commit. May your 2018 be full of passing pull requests!

tags: hello word



4 Kudos

Share

[< Newer Article](#) [Back to Blog](#) [Older Article >](#)

7 Comments



METALMAN01

Copilot Lvl 3

...

Thank you very much but without any thing to try and help when your stuck it makes it very frustrating. I'm knew to Python but, once I get a little help I believe I will do well. I'm a very fast learner but, like I said I HAVE KNOW IDEA OF WHAT I'M DOING SHOULD I QUIT.



0 Kudos



MOD

nadiajoyce

Moderator

...

@METALMAN01 This article is more about how to get started with setting up your GitHub account.

If you have specific questions when you get stuck on something while working in Python or on some project, you can always open a new Topic asking for help in the Project Development Help and Advice board. If you do that, with a specific question, another community member may be able to help you troubleshoot and solve your problem.

Welcome to the GitHub Community Forum. We're happy you're here!

Cheers!



0 Kudos

**METALMAN01**

Copilot Lvl 3

...

Thank you for taking the time to help I really, really want to better myself I was almost killed in a auto mobile accident 4 years ago. Now I have a hard time getting around so I got a computer and I've been stuck to it 6 to 7 hours a day. So my next venture is to learn Python because I also have looked into other languages and Python seems to be the one.

**1 Kudo****METALMAN01**

Copilot Lvl 3

...

I've already written a quote.

**0 Kudos****MOD nadiajoyce**

Moderator

...

@METALMAN01 It's great that you're trying to learn Python! This Community Forum may be a good resource for learning more from your peers. If, while working on your Python studies, you run into something you don't understand or that you need help with, you can just open a new Topic in the [Project Development Help and Advice board](#). Because this article is about setting up a github.com account, not about Python, you're going to find better help that board than by commenting on this article.

Thanks again for being here. Cheers!

**0 Kudos**



Tradge20

Copilot Lvl 2

...

Hi, I'm new to coding in general and I've be lost all day trying to get this to run thru Sublime Text (or is it subl running thru this?)... anyway, I now see I have 2 problems. I thought I'd be able to import still. but how do I fix it so I can import. my original problem is that I haven't ben able to run sublime so I can take a course. I'm using win 7 and Sublime 3 thanks and nice to meet you.



0 Kudos



MOD

nadiajoyce

Moderator

...

[@Tradge20](#) This article is specifically about setting up a new GitHub account. Your question seems to be a bit off-topic for this post, so I'd recommend opening a new Topic in the [Project Development Help and Advice](#) board with your question. It's much more likely other community members will see your question there and maybe be able to help you.



0 Kudos

You must be a registered user to add a comment. If you've already registered, sign in. Otherwise, register and sign in.

[Comment](#)



Other ways to get help



GitHub Help



@GitHubHelp



Contact GitHub



GitHub Enterprise
Support



More places to learn

Information

GitHub Learning Lab

GitHub Professional
Services

GitHub Guides

Open Source Guides

Blogs

GitHub

Atom

Electron

GitHub API



Community areas

Atom Community

GitHub Education Community

Open Source Maintainers on GitHub