git and GitHub

What is Git?

A free and open source distributed version control system.

Records changes to a file or set of files over time so that you can recall specific versions later.

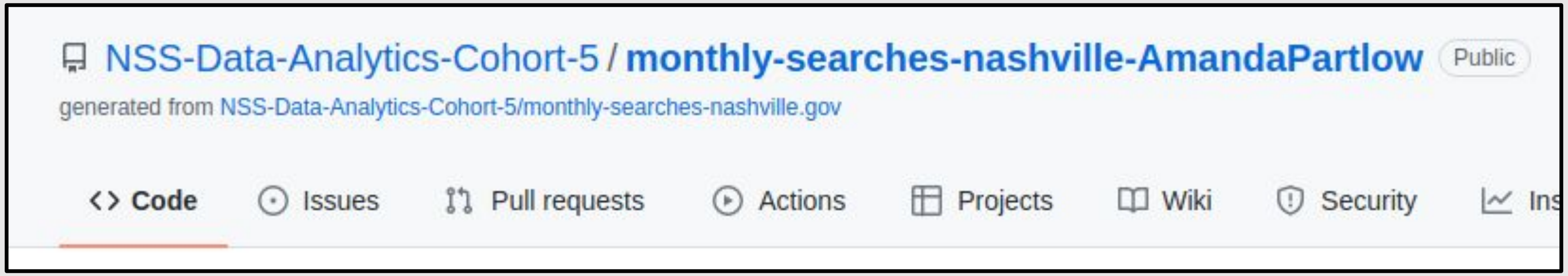
Files and changes to those files are stored in a **repository**.

What is GitHub?

A hosting service for Git repositories.

Also includes features for collaboration and project management.

Repository on GitHub



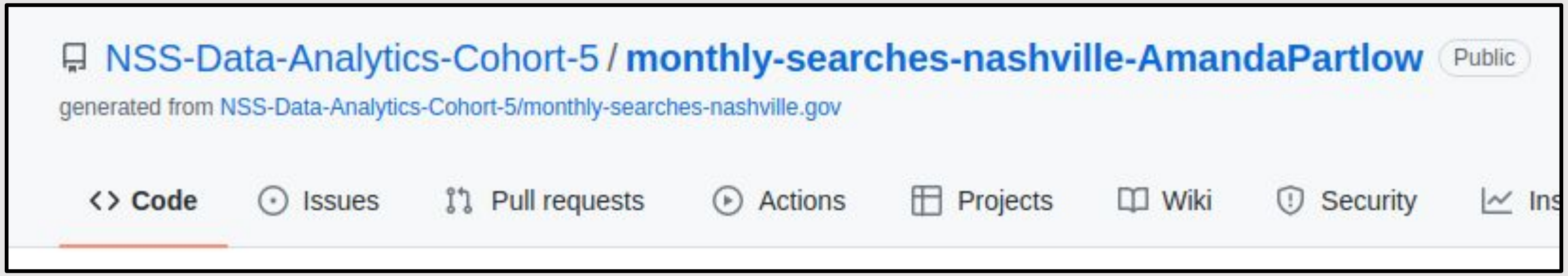
NSS-Data-Analytics-Cohort-5 / **monthly-searches-nashville-AmandaPartlow** Public

generated from NSS-Data-Analytics-Cohort-5/monthly-searches-nashville.gov

Internet

Your Computer

Repository on GitHub



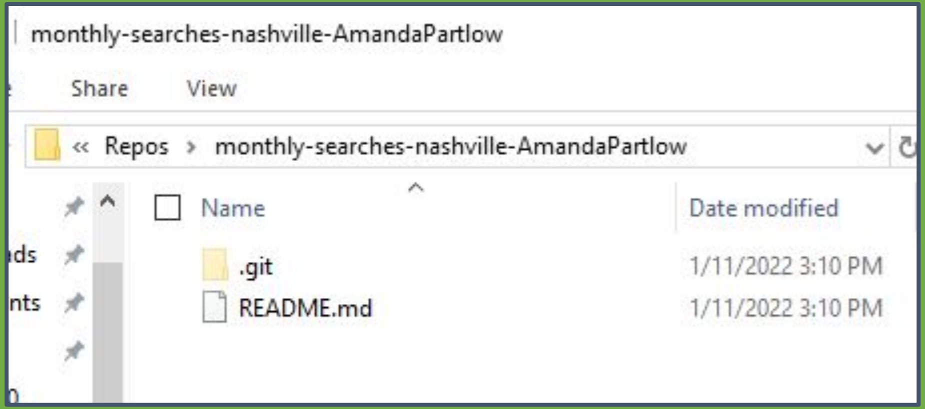
NSS-Data-Analytics-Cohort-5 / **monthly-searches-nashville-AmandaPartlow** Public

generated from NSS-Data-Analytics-Cohort-5/monthly-searches-nashville.gov

Internet

Clone

Your Computer



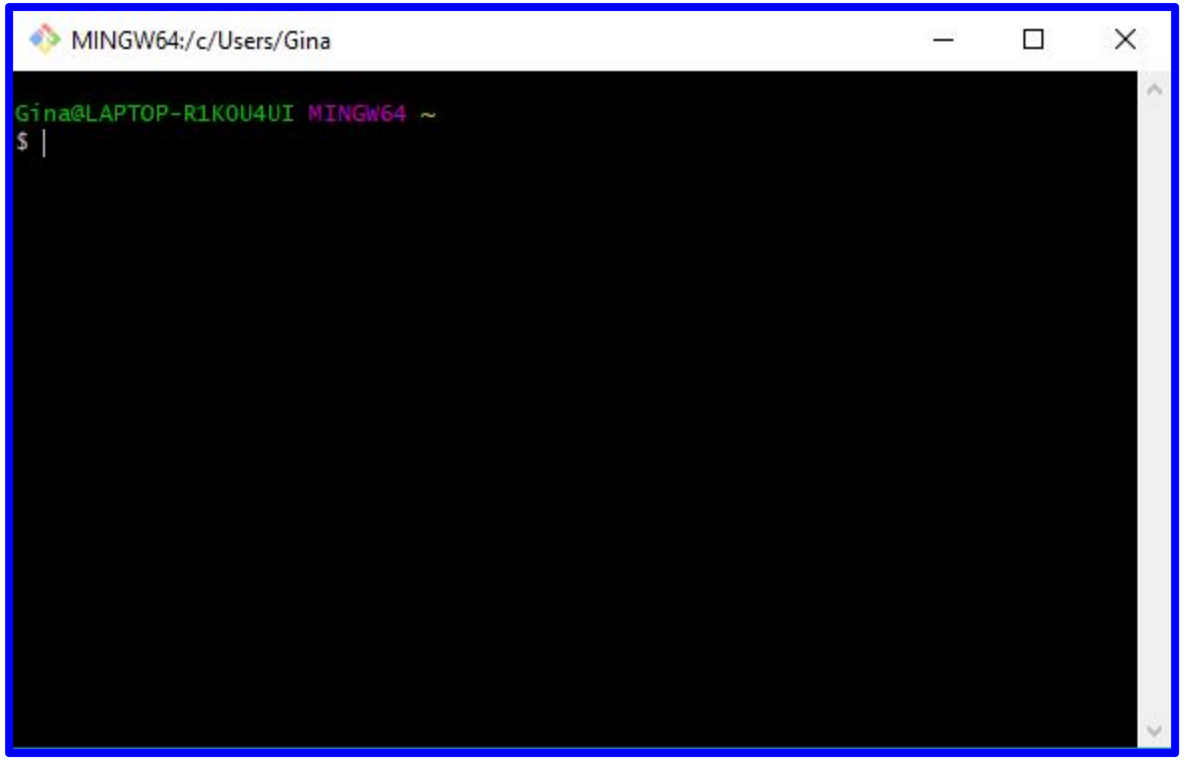
monthly-searches-nashville-AmandaPartlow
Share
View
« Repos › monthly-searches-nashville-AmandaPartlow
ds
nts

Name	Date modified
.git	1/11/2022 3:10 PM
README.md	1/11/2022 3:10 PM

Local Copy of Repository

Setup

- Since we will be working in Windows for our Excel projects, everyone should be working on their Windows machine.
- Install git (if not already done) (https://git-scm.com/downloads)
- After installing, open Git Bash
- If Git Bash didn't install automatically with git, you can get it here: https://gitforwindows.org/



```
Gina@LAPTOP-R1KOU4UI MINGW64 ~
$ |
```

Command Line Introduction

pwd

“print working directory” - tells you where you are located

ls

Lists the contents of the current directory

mkdir <directory name>

Creates a new directory

cd <directory name>

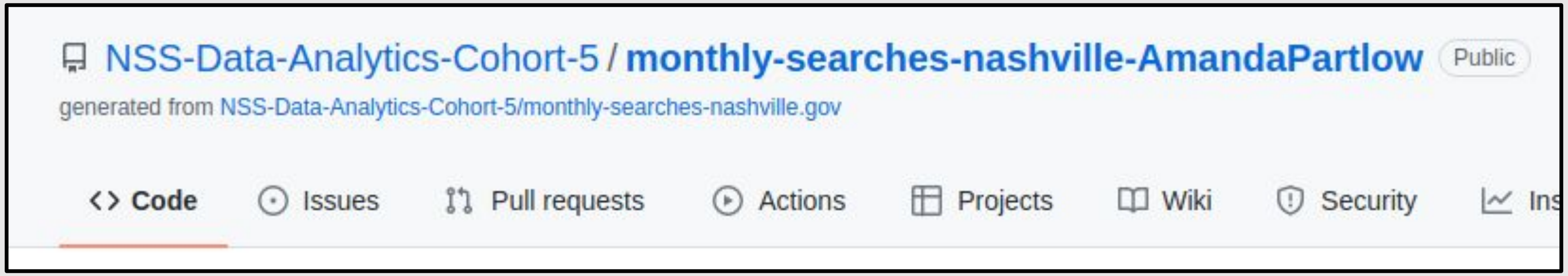
Changes the directory to the one specified

cd ..

Takes you up on level in the directory structure

Use these commands to navigate to where you want to save your class files. Create a new directory here named nss-data-analytics (or similar).

Repository on GitHub



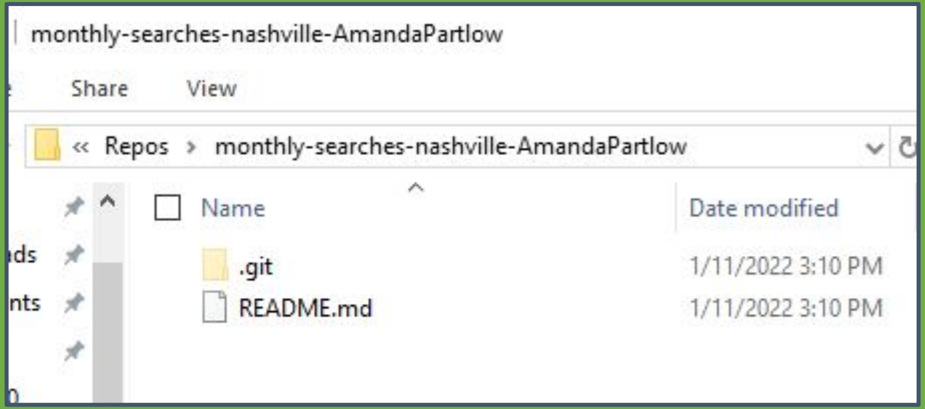
NSS-Data-Analytics-Cohort-5 / **monthly-searches-nashville-AmandaPartlow** Public

generated from NSS-Data-Analytics-Cohort-5/monthly-searches-nashville.gov

Internet

Clone

Your Computer



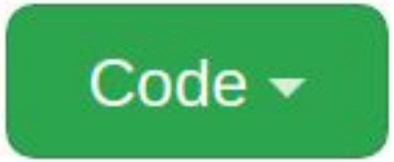
monthly-searches-nashville-AmandaPartlow

Name	Date modified
.git	1/11/2022 3:10 PM
README.md	1/11/2022 3:10 PM

Local Copy of Repository

Repository Setup

Navigate in your browser to the repository that was set up when you accepted the assignment. Locate the “Code” button in the upper right:



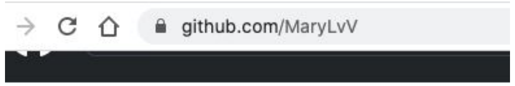
Ensure that HTTPS is selected and copy the URL.

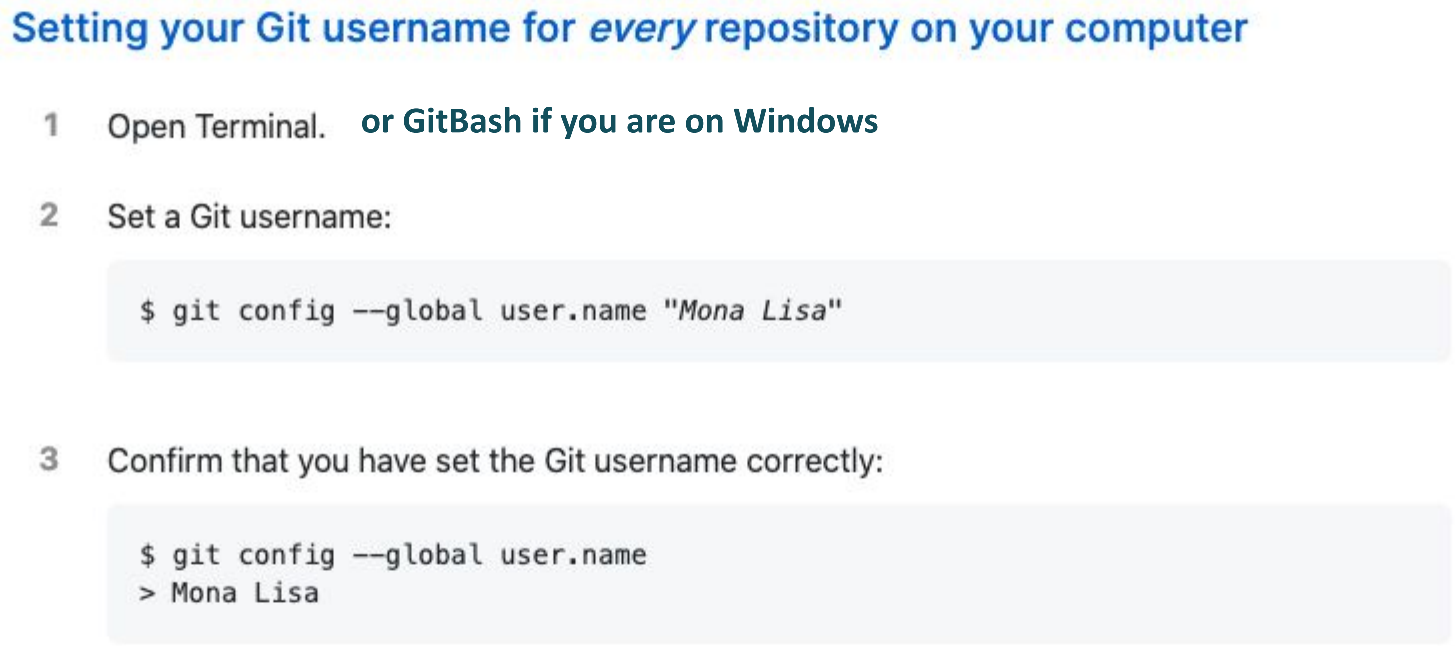
Clone your remote repository to create a local repo by running from Git Bash:

```
git clone <url to repository that you copied>
```

Next, configure your username and email in GitBash. This only needs to be done once. If you're on Mac, you'll need to do this again once we start SQL.

Your username is the account name. In the example below it would be **MaryLvV**. It is below your actual name and in the URL.





Setting your Git username for *every* repository on your computer

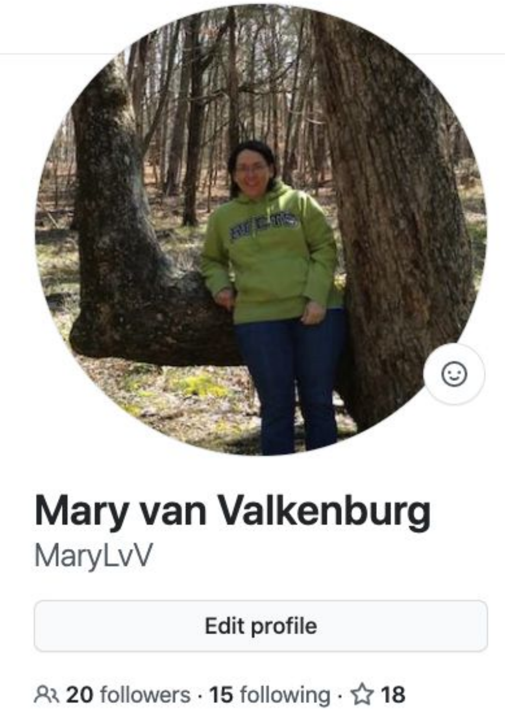
1 Open Terminal. **or GitBash if you are on Windows**

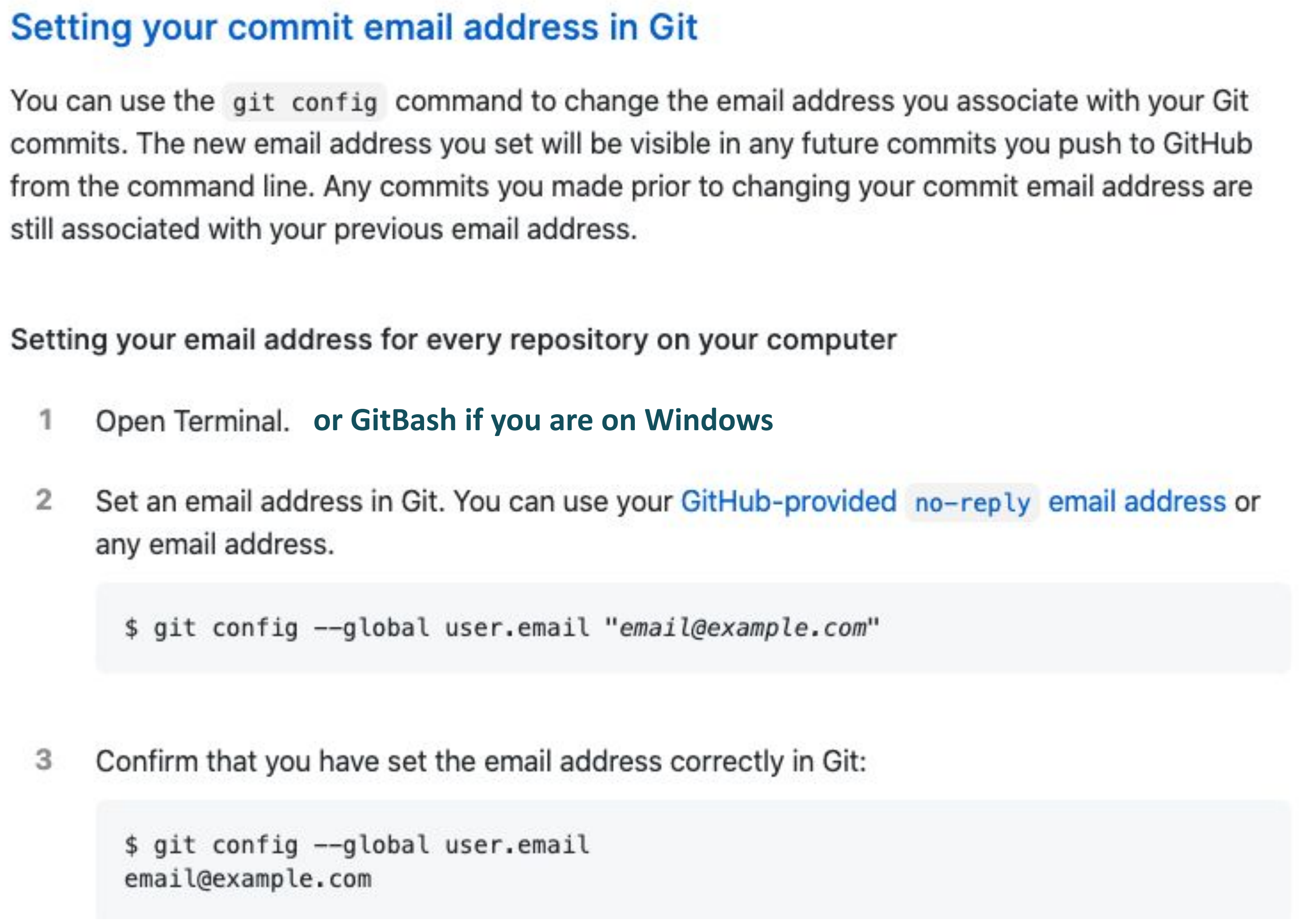
2 Set a Git username:

```
$ git config --global user.name "Mona Lisa"
```

3 Confirm that you have set the Git username correctly:

```
$ git config --global user.name
> Mona Lisa
```





Setting your commit email address in Git

You can use the `git config` command to change the email address you associate with your Git commits. The new email address you set will be visible in any future commits you push to GitHub from the command line. Any commits you made prior to changing your commit email address are still associated with your previous email address.

Setting your email address for every repository on your computer

1. Open Terminal. **or GitBash if you are on Windows**

2. Set an email address in Git. You can use your GitHub-provided `no-reply` email address or any email address.

```
$ git config --global user.email "email@example.com"
```

3. Confirm that you have set the email address correctly in Git:

```
$ git config --global user.email
email@example.com
```

Adding and Committing Changes

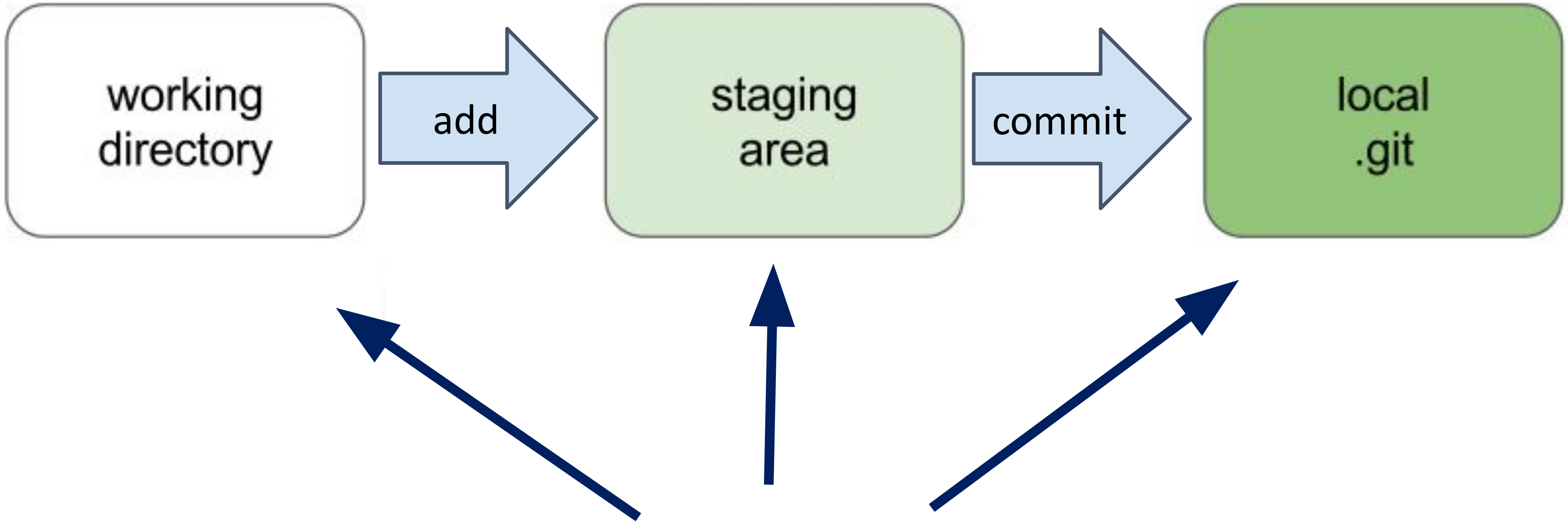
After finishing a unit of work (eg. finishing one of the project questions, end of class meeting), you should save and **commit** your changes.

1. First, add your changes to the **staging area**

```
git add <files>
```

2. Then commit those changes with a *descriptive* message (eg. “Completed question 3”):

```
git commit -m ‘<message>’
```



working directory

add

staging area

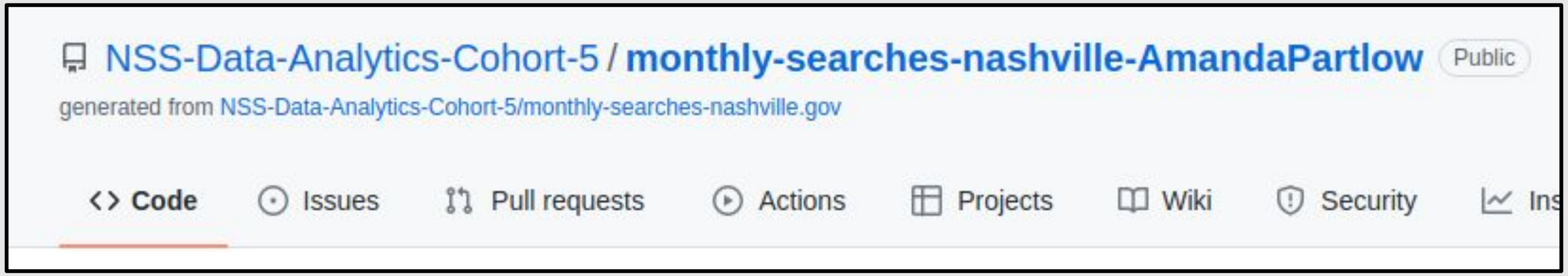
commit

local .git

on your computer

Image: https://edav.info/github.html

Repository on GitHub


NSS-Data-Analytics-Cohort-5 / monthly-searches-nashville-AmandaPartlow Public

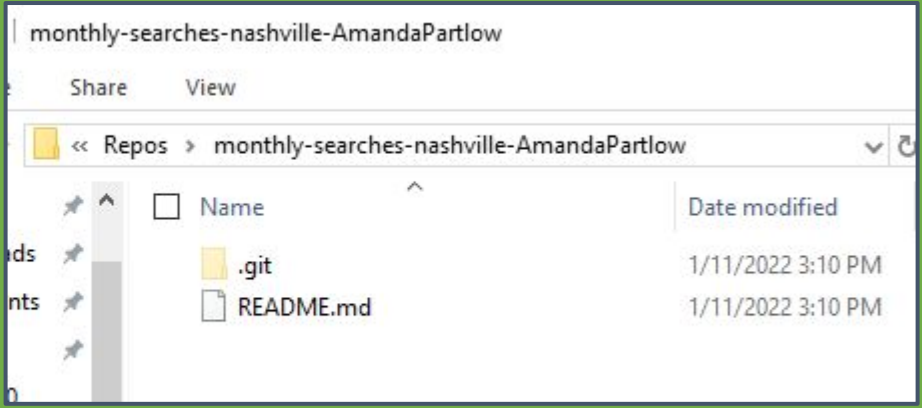
generated from NSS-Data-Analytics-Cohort-5/monthly-searches-nashville.gov

Internet

Clone

Push

Your Computer


monthly-searches-nashville-AmandaPartlow

Name	Date modified
.git	1/11/2022 3:10 PM
README.md	1/11/2022 3:10 PM

Local Copy of Repository

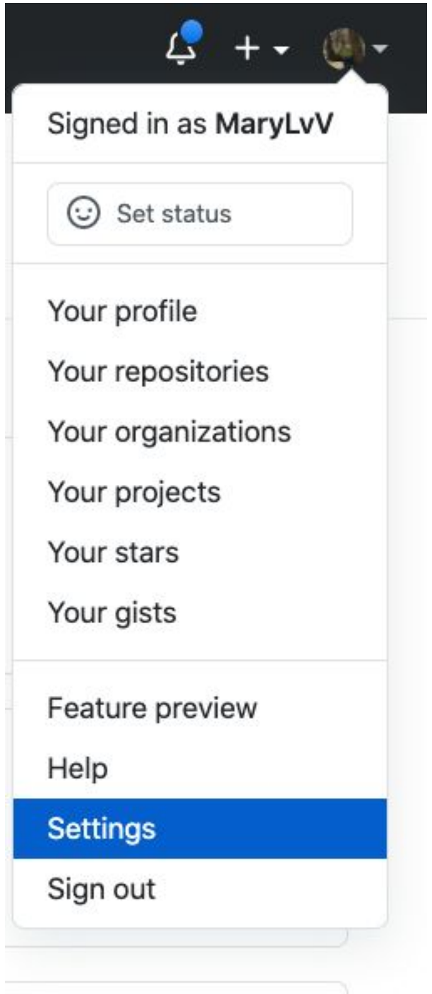
Create a PAT

Before pushing changes, we must create a personal access token on GitHub.

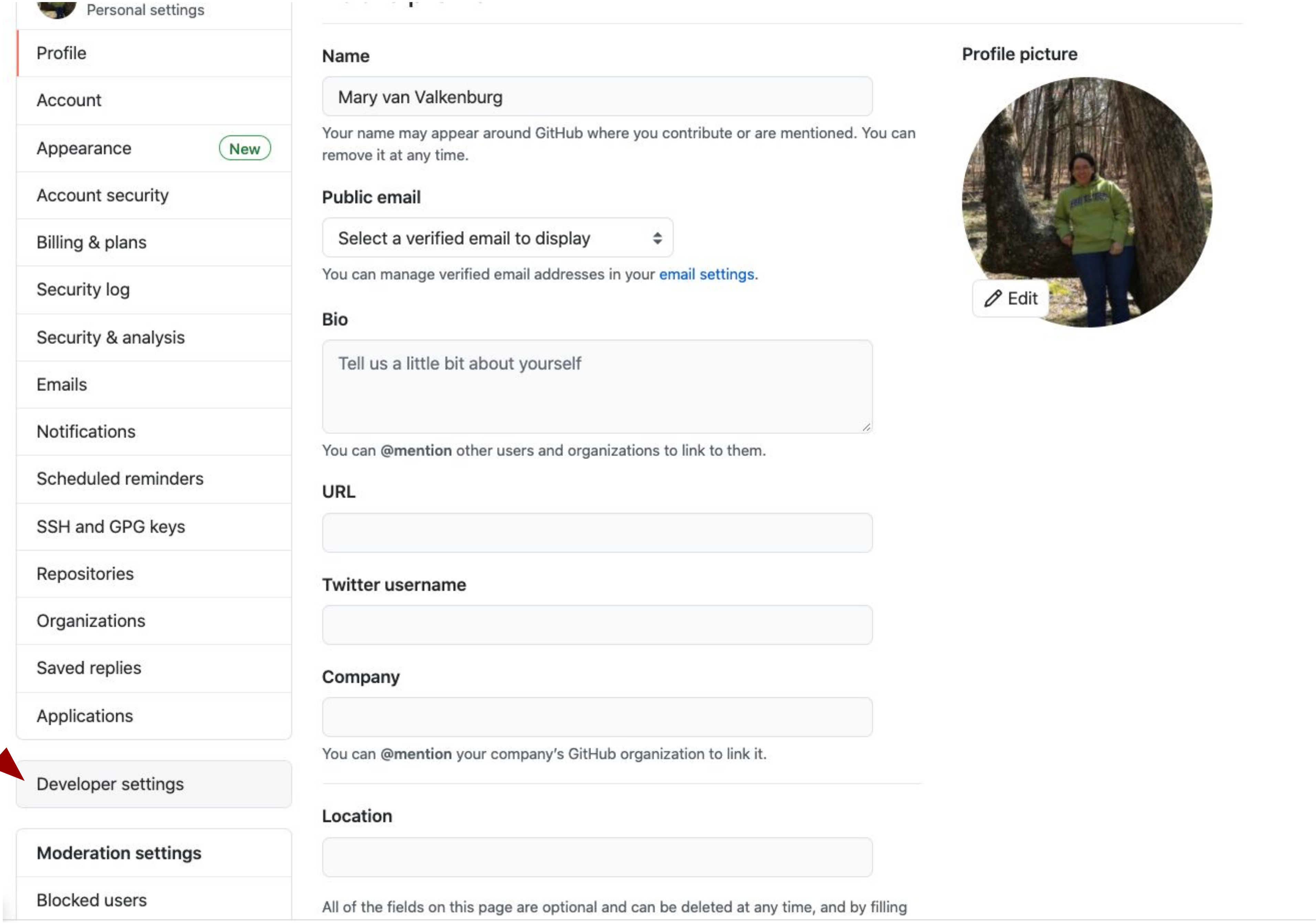
1. Open *settings* by clicking on your profile picture in the upper right corner.
2. In the menu on the left, choose *Developer Settings*, then choose *Personal access tokens*
3. Choose *Generate new token*, give it a name, and ensure that you have selected the **repo** scope
4. Take the resulting token and save it somewhere (or email to yourself).

Go to your account at **https://github.com/**

1. Create a personal access token (PAT)
   GitHub now requires PATs for command line interactions
   a. **Click on your avatar in the upper right corner and select Settings**



b. Click on Developer Settings



Name

Mary van Valkenburg

Your name may appear around GitHub where you contribute or are mentioned. You can remove it at any time.

Public email

Select a verified email to display

You can manage verified email addresses in your email settings.

Bio

Tell us a little bit about yourself

You can **@mention** other users and organizations to link to them.

URL

Twitter username

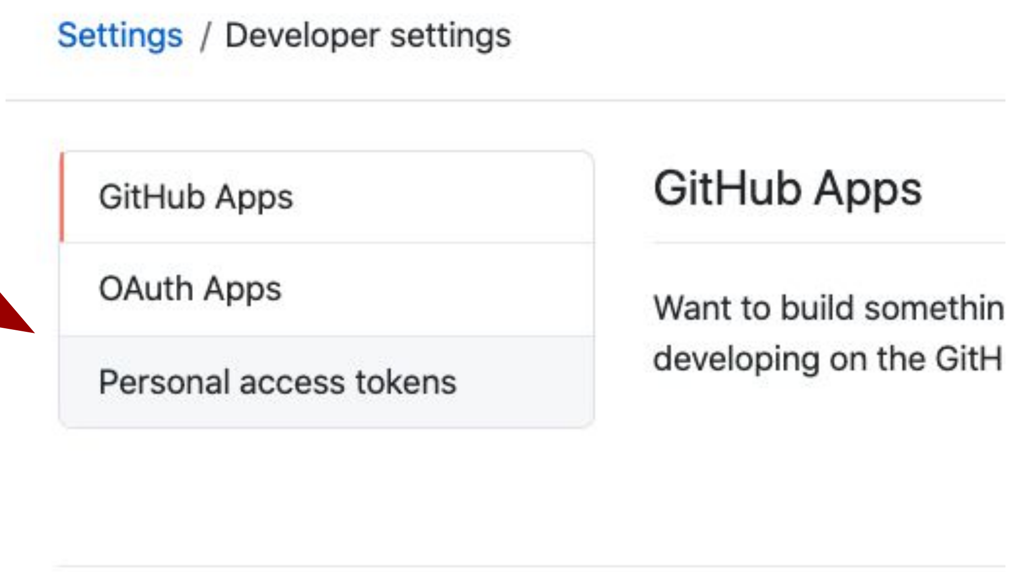
Company

You can **@mention** your company's GitHub organization to link it.

Location

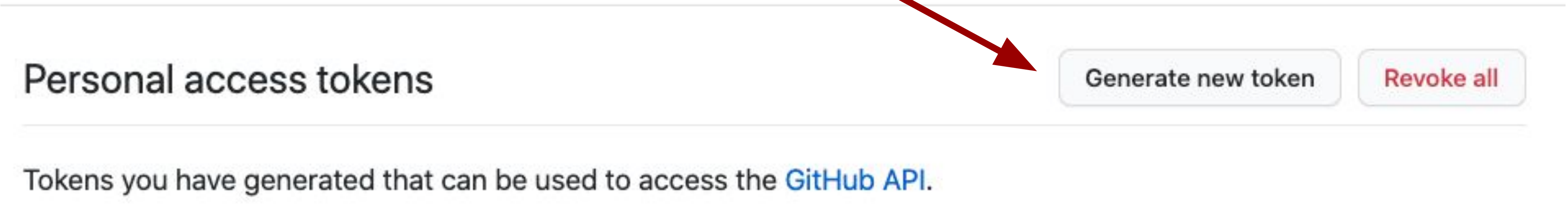
All of the fields on this page are optional and can be deleted at any time, and by filling

c. Select Personal access tokens

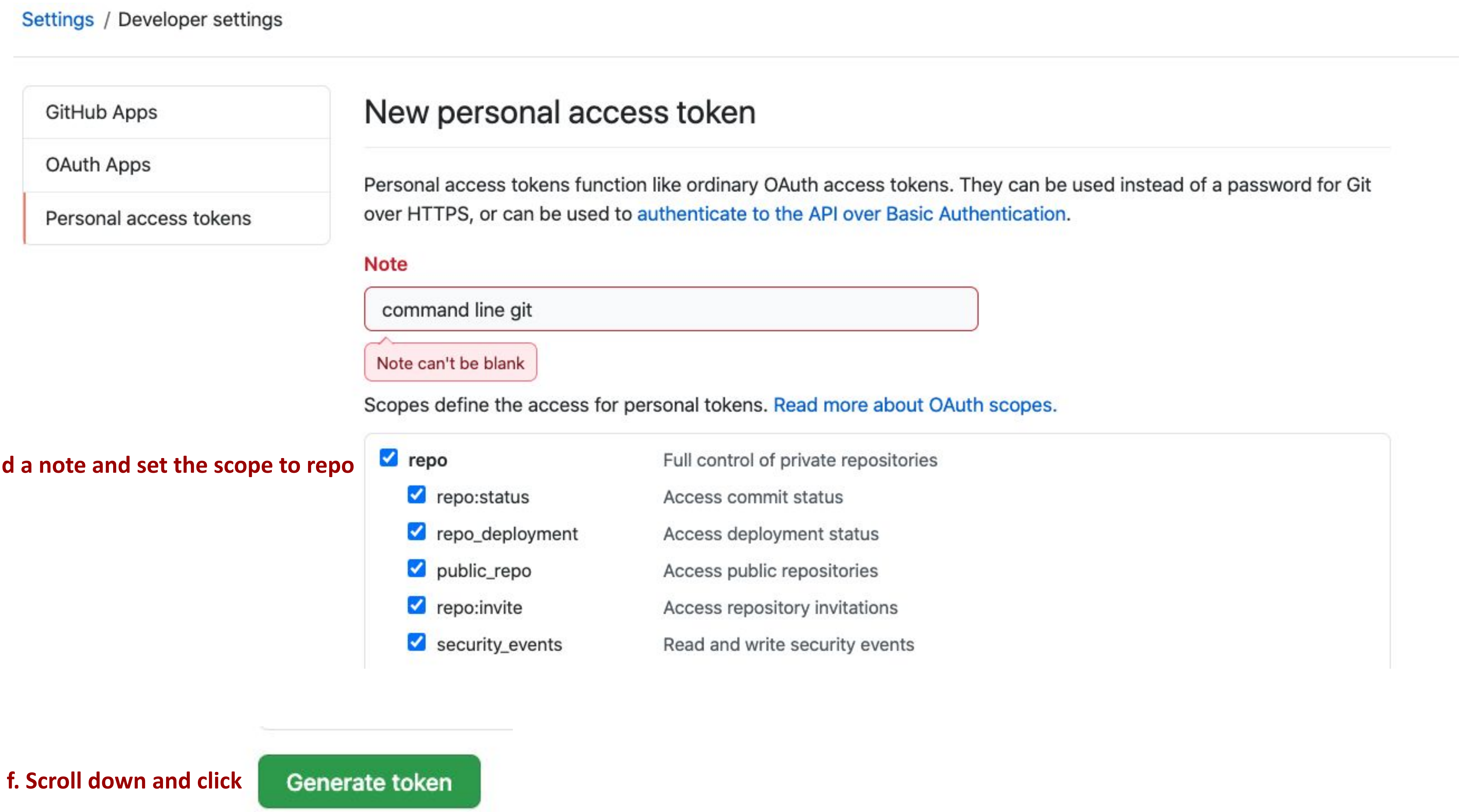


Want to build somethin
developing on the GitH

d. And Generate new token



Tokens you have generated that can be used to access the GitHub API.



New personal access token

Personal access tokens function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to authenticate to the API over Basic Authentication.

Note

command line git

Note can't be blank

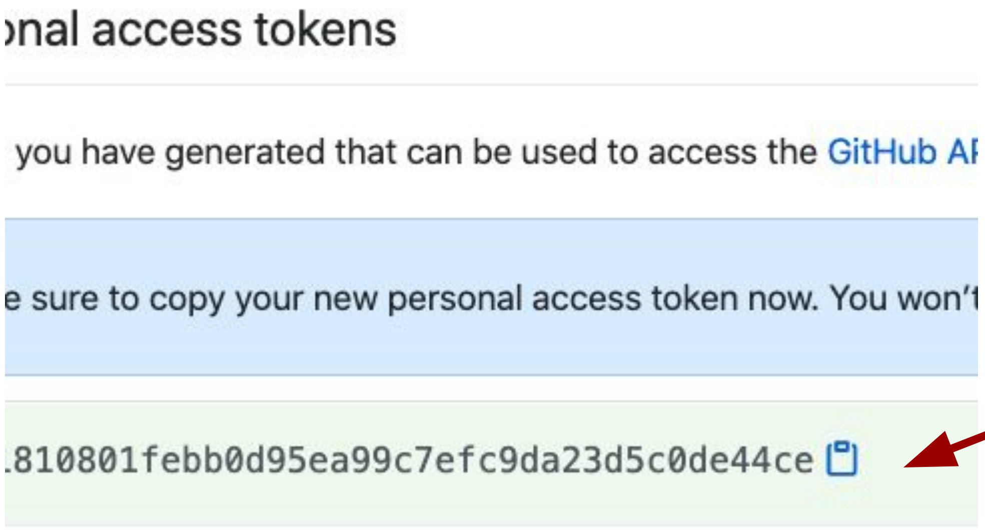
Scopes define the access for personal tokens. Read more about OAuth scopes.

☑ repo	Full control of private repositories
☑ repo:status	Access commit status
☑ repo_deployment	Access deployment status
☑ public_repo	Access public repositories
☑ repo:invite	Access repository invitations
☑ security_events	Read and write security events

e. Add a note and set the scope to repo

f. Scroll down and click

Very important



nal access tokens

you have generated that can be used to access the GitHub AP

e sure to copy your new personal access token now. You won’t

.810801febb0d95ea99c7efc9da23d5c0de44ce

Click the little clipboard to copy your token

Recommended Workflow

1. Work on files locally. When done, run

   ```
   git add <files>
   ```

2. Commit changes:

   ```
   git commit -m ‘<message>’
   ```

The first two steps have already been done.

3. Push to GitHub:

   ```
   git push origin main
   ```

   When prompted for a password, paste your PAT.

Recommended Workflow

1. Work on files locally. When done, run

```
git add <files>
```

2. Commit changes:

```
git commit -m ‘<message>’
```

origin indicates that we are pushing to GitHub

3. Push to GitHub:

```
git push origin main
```

When prompted for a password, paste your PAT.

Recommended Workflow

1. Work on files locally. When done, run

```
git add <files>
```

2. Commit changes:

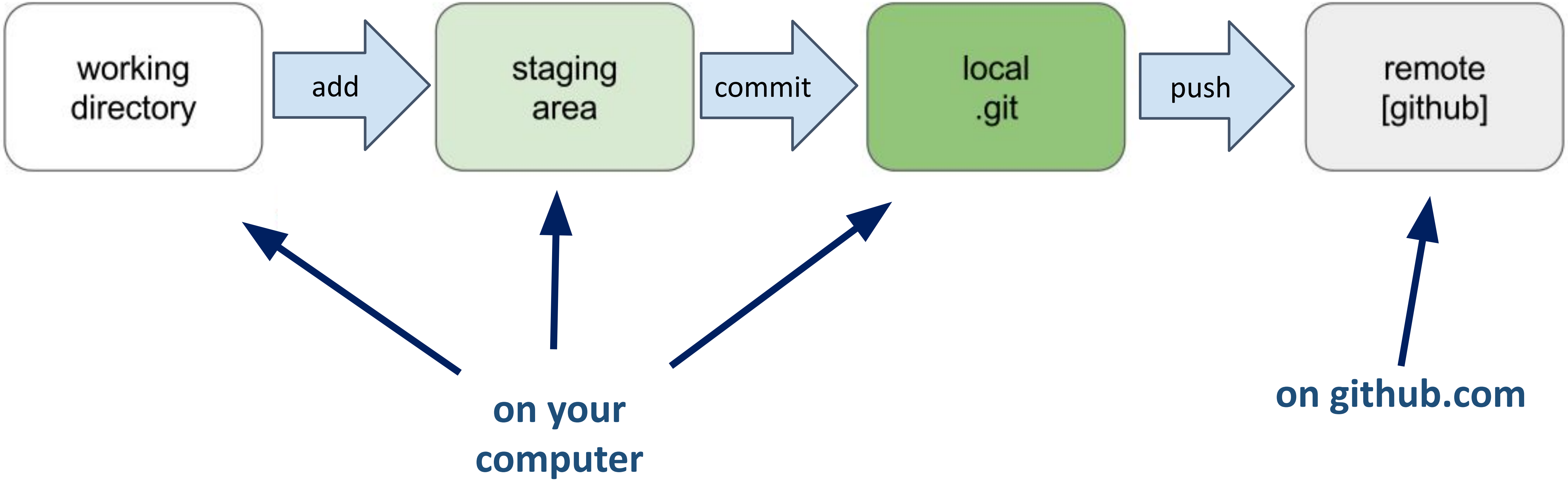
```
git commit -m ‘<message>’
```

We’ll learn about what main means later.

3. Push to GitHub:

```
git push origin main
```

When prompted for a password, paste your PAT.



on your computer

on github.com

Image: https://edav.info/github.html