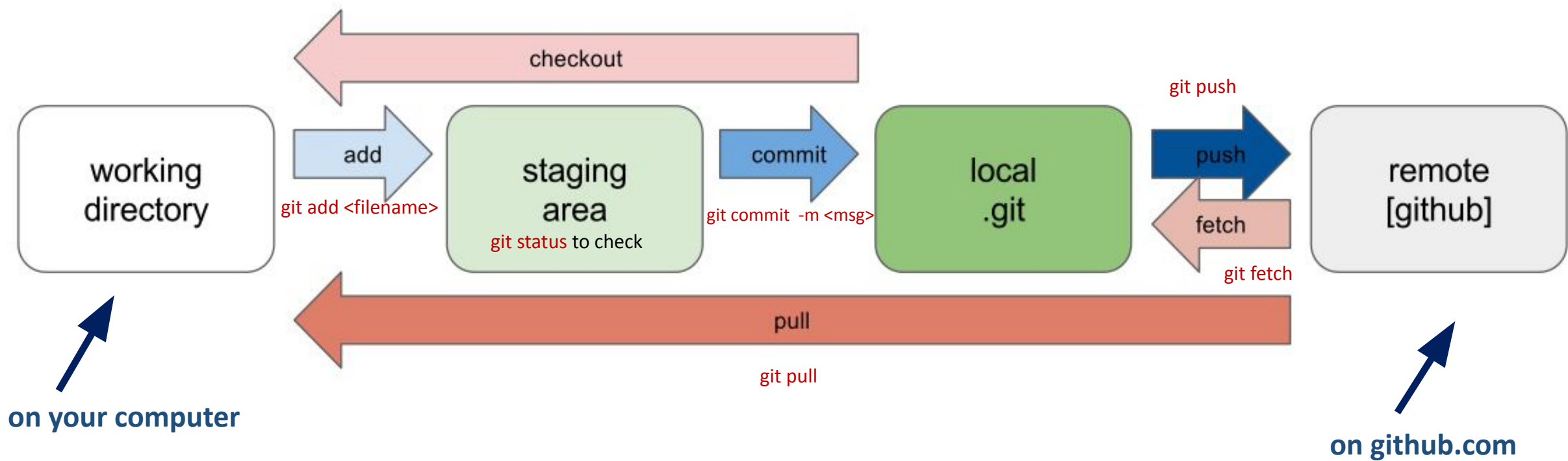
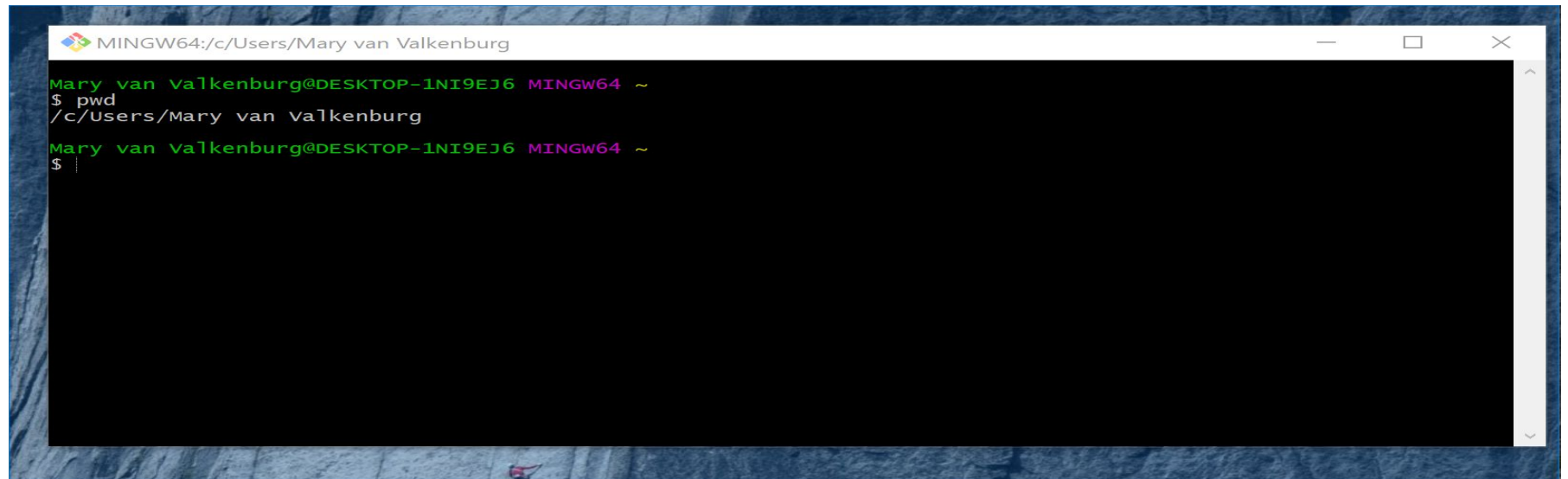


# git and GitHub



*(should already be done)*

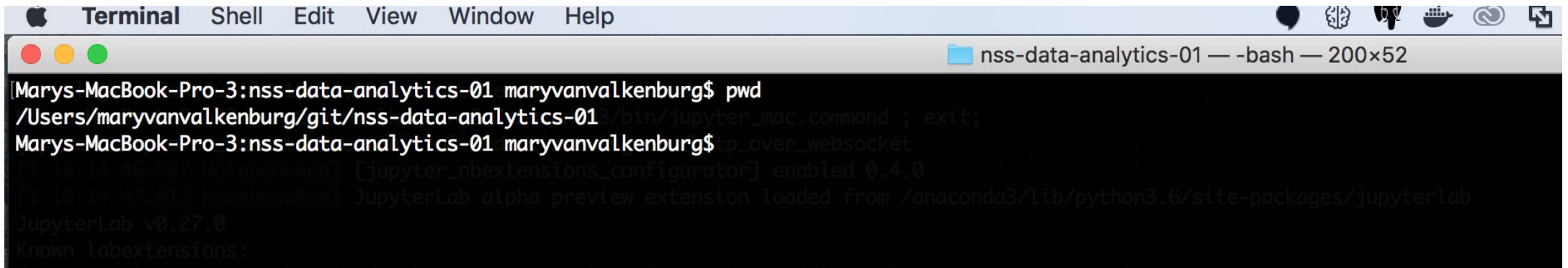
- Install git (<https://git-scm.com/downloads>)
- Since we will be working in Windows for our Excel projects, everyone will need to clone their assignment repositories to their Windows machine
- On Windows:
  - Open Git Bash:

A screenshot of a Git Bash terminal window. The title bar at the top reads "MINGW64:/c/Users/Mary van Valkenburg". The terminal content shows the prompt "Mary van Valkenburg@DESKTOP-1NI9EJ6 MINGW64 ~" followed by the command "\$ pwd" and its output "/c/Users/Mary van Valkenburg". Below that, the prompt "Mary van Valkenburg@DESKTOP-1NI9EJ6 MINGW64 ~" is shown again, followed by "\$" and a cursor. The terminal has a black background with green text for the prompt and pink text for the command and output. The window has standard Windows window controls (minimize, maximize, close) in the top right corner.

If Git Bash didn't install automatically with git, you can get it here: <https://gitforwindows.org/>

## *For Mac/Linux users for later*

- When working on Mac:
  - Open Terminal

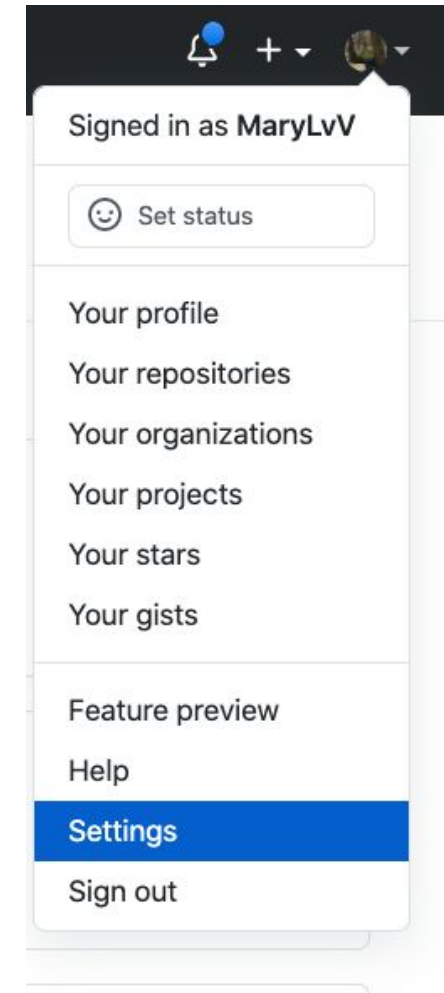


```
Terminal  Shell  Edit  View  Window  Help
nss-data-analytics-01 — -bash — 200x52
[Marys-MacBook-Pro-3:nss-data-analytics-01 maryvanvalkenburg$ pwd
/Users/maryvanvalkenburg/git/nss-data-analytics-01
[Marys-MacBook-Pro-3:nss-data-analytics-01 maryvanvalkenburg$ jupyter lab --ip_over_websocket
[I 16:14:46.847 NotebookApp] [jupyter_nbextensions_configurator] enabled 0.4.0
[I 16:14:47.012 NotebookApp] JupyterLab alpha preview extension loaded from /anaconda3/lib/python3.6/site-packages/jupyterlab
JupyterLab v0.27.0
Known labextensions:
```

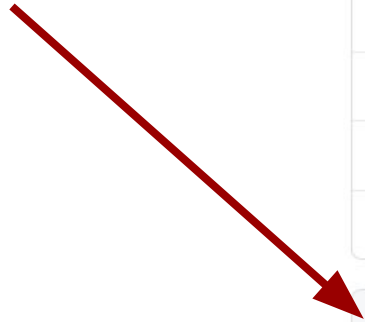
- Find your working directory with this command: **pwd**
  - These are other helpful shell commands for navigating through your directories:
    - **ls** lists the contents of the current directory
    - **cd <directory name>** changes the directory to the one specified
    - **cd ..** Takes you up one level in the directory structure
- Navigate to where you want to create our local git repositories and make your directory and create a directory called git: with **mkdir git**
- Change to the new git directory with **cd git**
- Make a directory called nss-data-analytics (or similar) with **mkdir <new folder name>**
- Change to the directory with **cd <new folder name>**

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**



Personal settings

- Profile
- Account
- Appearance New
- Account security
- Billing & plans
- Security log
- Security & analysis
- Emails
- Notifications
- Scheduled reminders
- SSH and GPG keys
- Repositories
- Organizations
- Saved replies
- Applications
- Developer settings**
- Moderation settings
- Blocked users

## 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

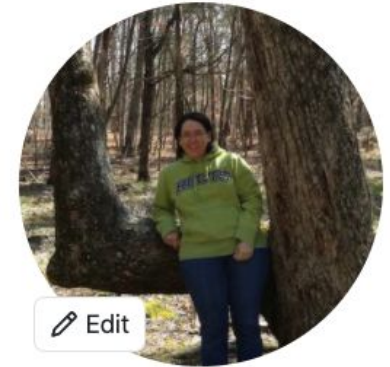
## Company

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

## Profile picture



Edit

c. Select Personal access tokens

[Settings](#) / Developer settings

GitHub Apps

OAuth Apps

Personal access tokens

## GitHub Apps

Want to build somethin  
developing on the GitH

d. And Generate new token

## Personal access tokens

Generate new token

Revoke all

Tokens you have generated that can be used to access the [GitHub API](#).



- GitHub Apps
- OAuth Apps
- Personal access tokens

## 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](#).

e. Add a note and set the scope to repo

- ☒ 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

f. Scroll down and click

Generate token

# Very important

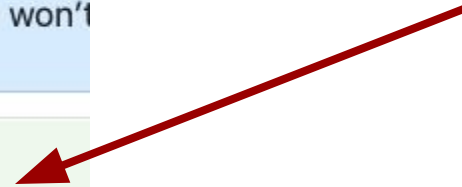
Personal access tokens

you have generated that can be used to access the [GitHub API](#)

Be sure to copy your new personal access token now. You won't

gh10801febb0d95ea99c7efc9da23d5c0de44ce 

Click the little clipboard to copy your token



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

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

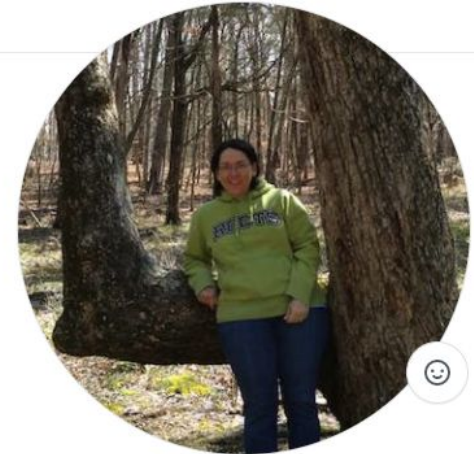
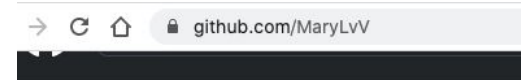
2 Set a Git username:

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

3 Confirm that you have set the Git username correctly:

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

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



**Mary van Valkenburg**

MaryLvV

Edit profile

👤 20 followers · 15 following · ☆ 18

## 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
```

**Click the link provided by your instructor to accept the first assignment. This will create an assignment repository for you.**

**Go to your assignment repository using cd in GitBash**

**Clone your remote repository to create a local repo**

copy clone url from github.com



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

**Add a file (any file) to your repo. Don't worry, you'll delete it later.**

**cd to your repo**

**git add .**

**git commit -m "test file"**

**git push origin main**

**\* When prompted for a password, paste your PAT into GitBash**