

# Git & Github

Hope Townsend, Aug 2023

# Why should I care?

- Version control
  - Group work
  - Helps keep track of your own code and helps force better documentation
- Many tools are only accessible via Github
- Will be used in Software-Engineering for Scientists



<https://storage.googleapis.com/gweb-uniblog-publish-prod/images/Google-Docs-logo-transparent.max-2800x2800.png>

# Git

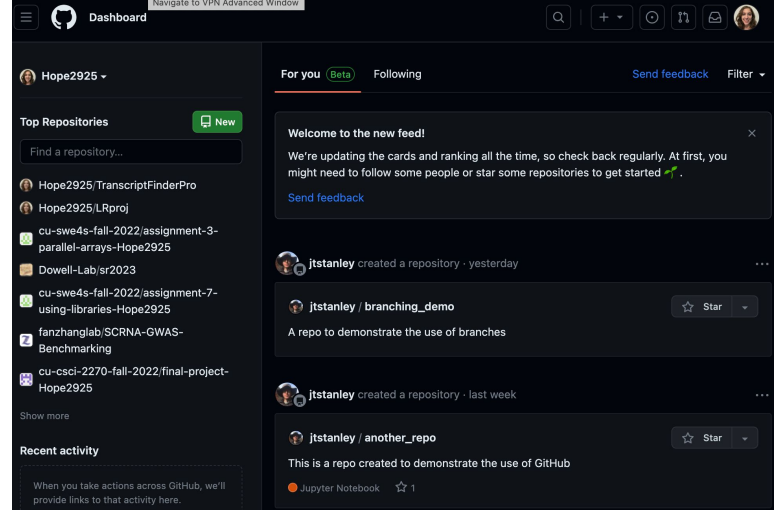
- the version control system/**software**



https://th.bing.com/th/id/R.7894b78e927a1287dc688ab40ff74b9?rik=QIRI%2fpx%2brJ2Zeg&riu=http%3a%2f%2fwww.roelee.co.uk%2fimages%2fsoftware.jpg&ehk=Lpa%2fx9EVSp8ZwDDFRGRYrMo2nKH83%2bsUBLAdzME3xxQ%3d&risl=&pid=ImgRaw&r=0

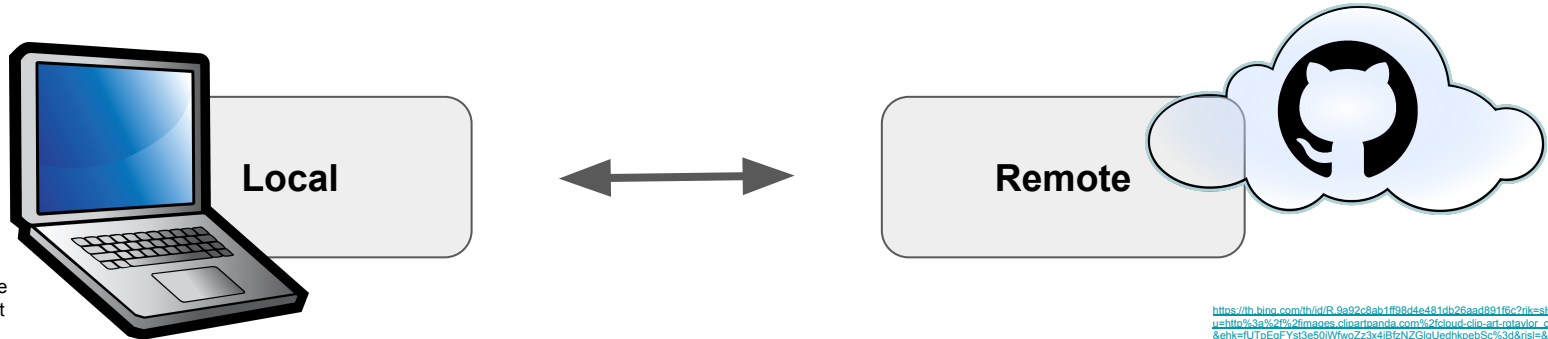
# Github

- **web-based service** to facilitate sharing Git repositories



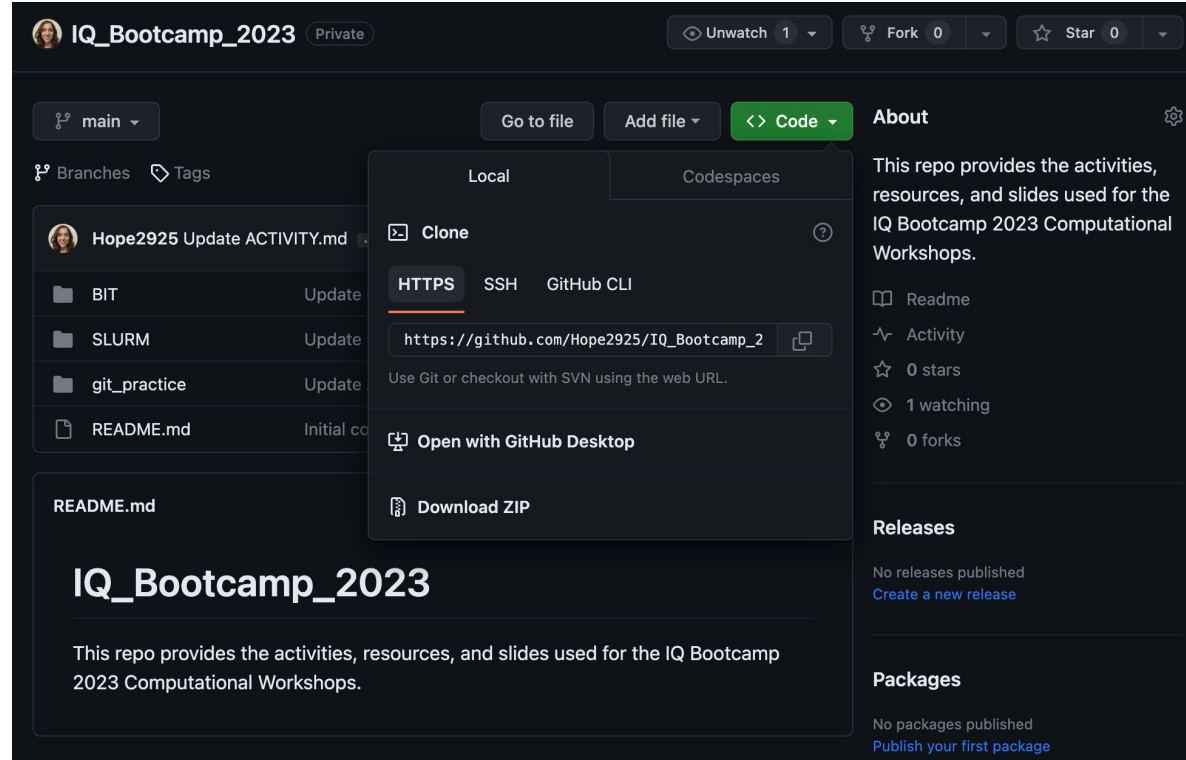
# Git Lingo

- **Repository**= folder with the version controlled content
  - Can contain folders & files within it
  - Usually one per project
- **README**= file with important information about repository (e.g. how to use it if tool, dependencies, other important notes)
- **Remote**= Github's version of a repository
- **Local**= Your personal machine's version of a repository



# How do we stay connected? (steps 1 & 2 of ACTIVITY.md)

1. Find Github repository (can create one or use another's)
2. Clone the repository:
  - a. Copy the https link



# Why is there https:// and ssh://?

- How can the remote system ensure only certain people can read/write the repositories?
  - It must be able to recognize the User securely
- 2/3 ways: Personal Access Tokens (PATs) (https) & SSH keys (ssh)
  - We will be using PATs today as they have some advantages over SSH. Interested in more?  
Click [here](#) for a short layman description and [here](#) for Github's description.
- But for now, let's see what happens if you don't have one.

Demo: Continue by  
following  
ACTIVITY.md Step 3  
on

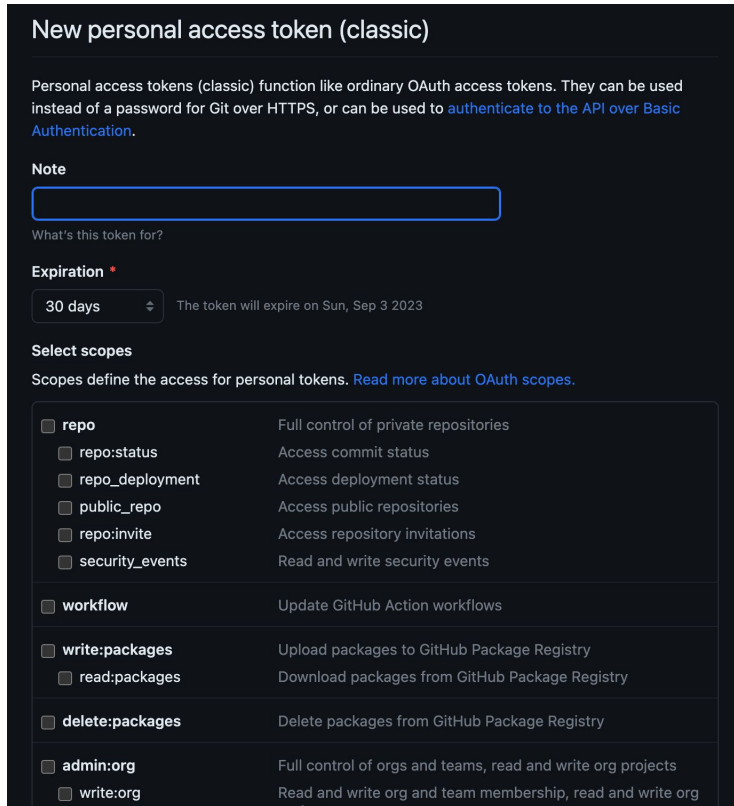
# Create a personal access token

Full instructions on [ACTIVITY.md](#)

1. Create a PAT
2. Type a name/note
3. Select the access you want
4. Copy the token password and save it somewhere on your computer

*This is what you'll type in as your password*

Alternatives?



The screenshot shows the GitHub interface for creating a new personal access token (classic). The title is "New personal access token (classic)". Below the title, there is a paragraph explaining that these tokens function like ordinary OAuth access tokens and can be used instead of a password for Git over HTTPS, or to authenticate to the API over Basic Authentication. A "Note" field is present with a text input box. Below the note field, there is a section for "Expiration" with a dropdown menu set to "30 days" and a note that the token will expire on Sun, Sep 3 2023. The "Select scopes" section follows, with a note that scopes define the access for personal tokens and a link to read more about OAuth scopes. The scopes are listed in a table-like format with checkboxes and descriptions.

Scope	Description
<input type="checkbox"/> repo	Full control of private repositories
<input type="checkbox"/> repo:status	Access commit status
<input type="checkbox"/> repo_deployment	Access deployment status
<input type="checkbox"/> public_repo	Access public repositories
<input type="checkbox"/> repo:invite	Access repository invitations
<input type="checkbox"/> security_events	Read and write security events
<input type="checkbox"/> workflow	Update GitHub Action workflows
<input type="checkbox"/> write:packages	Upload packages to GitHub Package Registry
<input type="checkbox"/> read:packages	Download packages from GitHub Package Registry
<input type="checkbox"/> delete:packages	Delete packages from GitHub Package Registry
<input type="checkbox"/> admin:org	Full control of orgs and teams, read and write org projects
<input type="checkbox"/> write:org	Read and write org and team membership, read and write org



# If time... branches

How does a tool posted on Github make updates without messing up the current version?

Branches!!

Code for using branches can be found on the README

