H3ABioNet

Pan African Bioinformatics Network for H3Africa

Introduction to: GitHub

Azza Ahmed

moz://a

SageBionetworks

Suggested outline

- Basic Curriculum
  - https://docs.google.com/presentation/d/19UsTRcuNKWm8fCJUBgps8BP5xwI96A34RmGQC4Ir9Ww/edit#slide=id.g25c238d0a2_0_23
  - https://docs.google.com/presentation/d/1LNvTgx7fqOSZEDYKkitNqVpNUqXUlK3o502RSVnHAQY/edit#slide=id.g28bca07808_2_138
  - Jargon
  - Issues, Labels, Milestones
  - Project Management
  - Contributor vs Collaborator
  - Pull Requests
  - Forking
  - New folders / new files
- Quiz
- BREAK*
  - Practical: form small groups and have them work together (collaborators on the same project, etc.)
    - Develop a project, work together
    - README
      - Formatting, Lists, Images, Links, Tables, Checklists
      - Code of Conduct
      - CONTRIBUTING
      - Markdown guide: https://github.com/adam-p/markdown-here/wiki/Markdown-Cheatsheet https://guides.github.com/features/mastering-markdown/
  - Gh-pages
  - Gitbook
  - Continuous Integration (e.g. https://travis-ci.org/)
  - http://swcarpentry.github.io/git-novice/

Contents

- Automated Version Control (with Git)
- Who is “GitHub”
- Getting started
- Wanting more?

The No “Git” universe

- Problem:
  Endless cycles of revisions

- Sol:
  Version Control

- Examples:
  Google docs- Version history
  Word- Track changes
  Authoerea- Version history

Version control

Version control systems

- Version control allows many people to work in parallel.
- Not new- legacy examples from the 1980s: RCS, CVS, or Subversion from the 1980
- It is is like an *unlimited ‘undo’.*

Version control systems

- Tools that:
  - keep track of changes
  - Records of changes – **“commits”**
- Keeps metadata about changes
- The complete history of commits and their metadata makes up a “repository/ repo”

Repos can be kept in sync across different computers

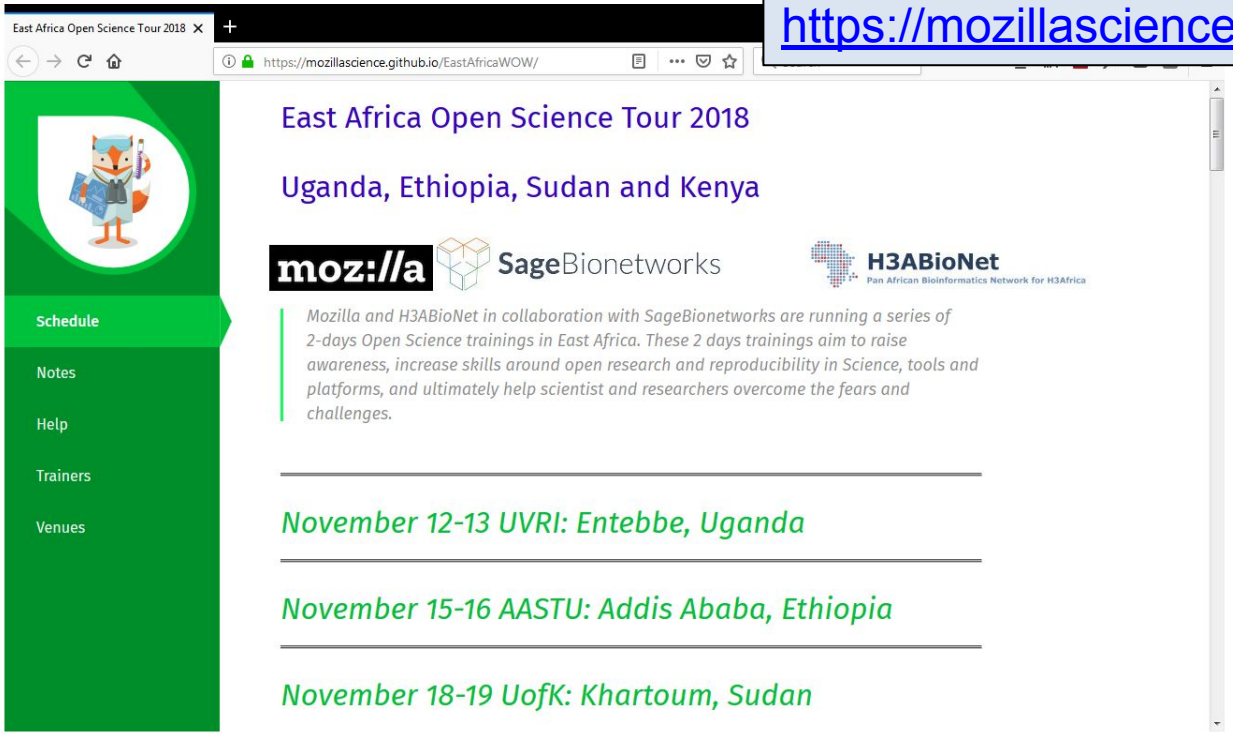
Git and GitHub

- Git is an automated version control system
- Many hosting services:
  - GitHub,
  - BitBucket,
  - GitLab … etc

- Can host everything: code, documentaion, … etc
- Offers paid plans for private projects, and free plans giving this feature for research and educational use

Who uses Github?

https://mozillascience.github.io/EastAfricaWOW/



Who uses Github?

https://brian-bot.github.io/about/

about | brian m. bot

Search with Google or enter address

brian m. bot

about community presentations publications

about

life ∋ { professional, personal }

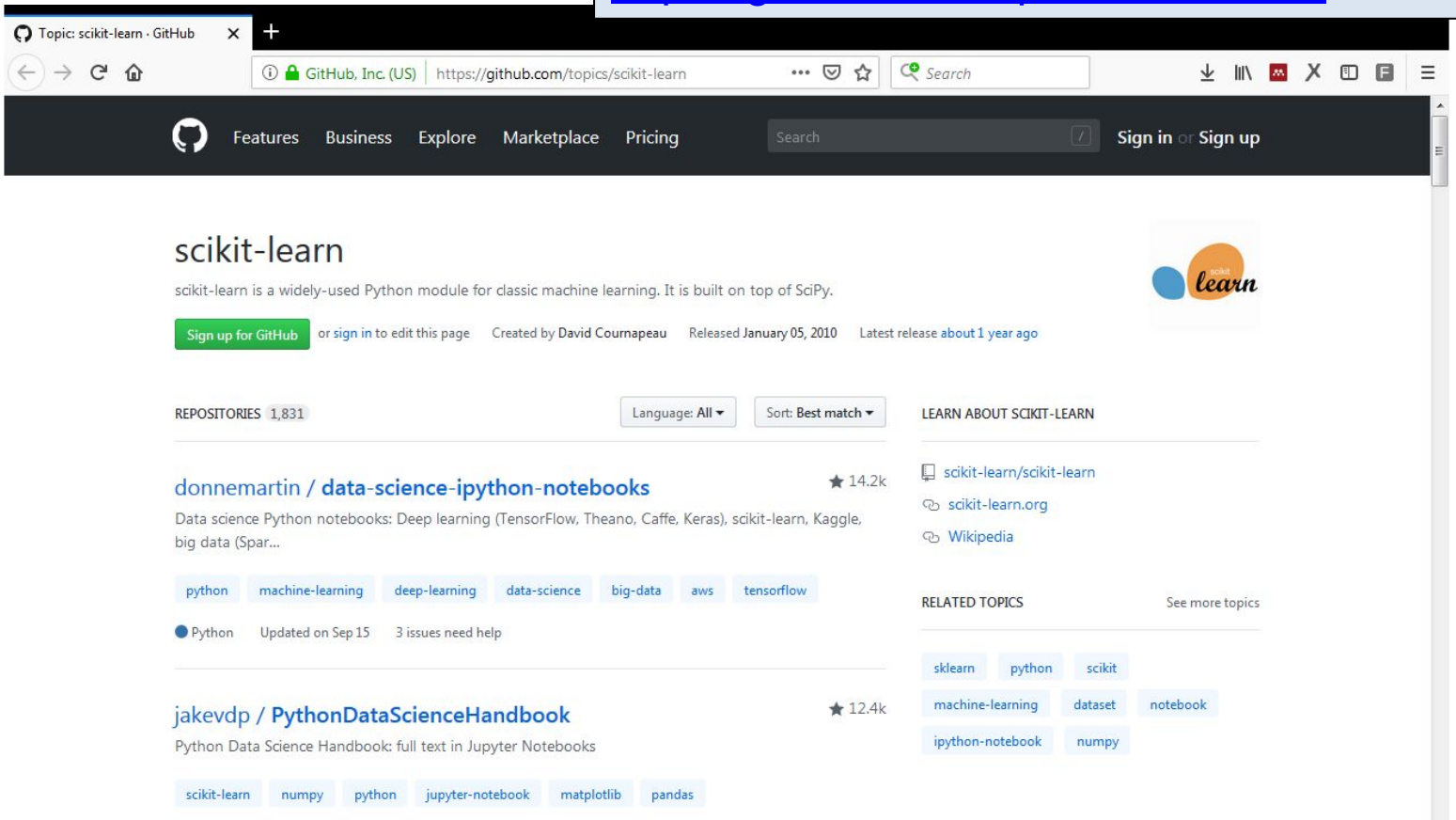
professional life

My days are spent as Principal Scientist and Head of Outreach and Strategic Development at Sage Bionetworks. Previously, I worked for seven years as a statistician at the Mayo Clinic primarily in cancer clinical trials and cancer genomics. I have extensive experience working with clinical and genomic data and have long had a passion for exploring ways to make science more open and transparent. I consider myself a science advocate and am involved in a number of national efforts to promote a research ecosystem where participants are treated as partners in the research process. I have also been an invited speaker at a number of national and international events to share my experiences living at the intersection of biomedical research, technology, and policy. I was recently named by Mozilla as one of "50 People Who Are Making the Internet a Better Place."

My current work aims to make the biomedical research system more effective by challenging the traditional roles of researchers, institutions, funders, and research participants. At its heart, this work is rooted in building trust between these parties as well as with the public at large.

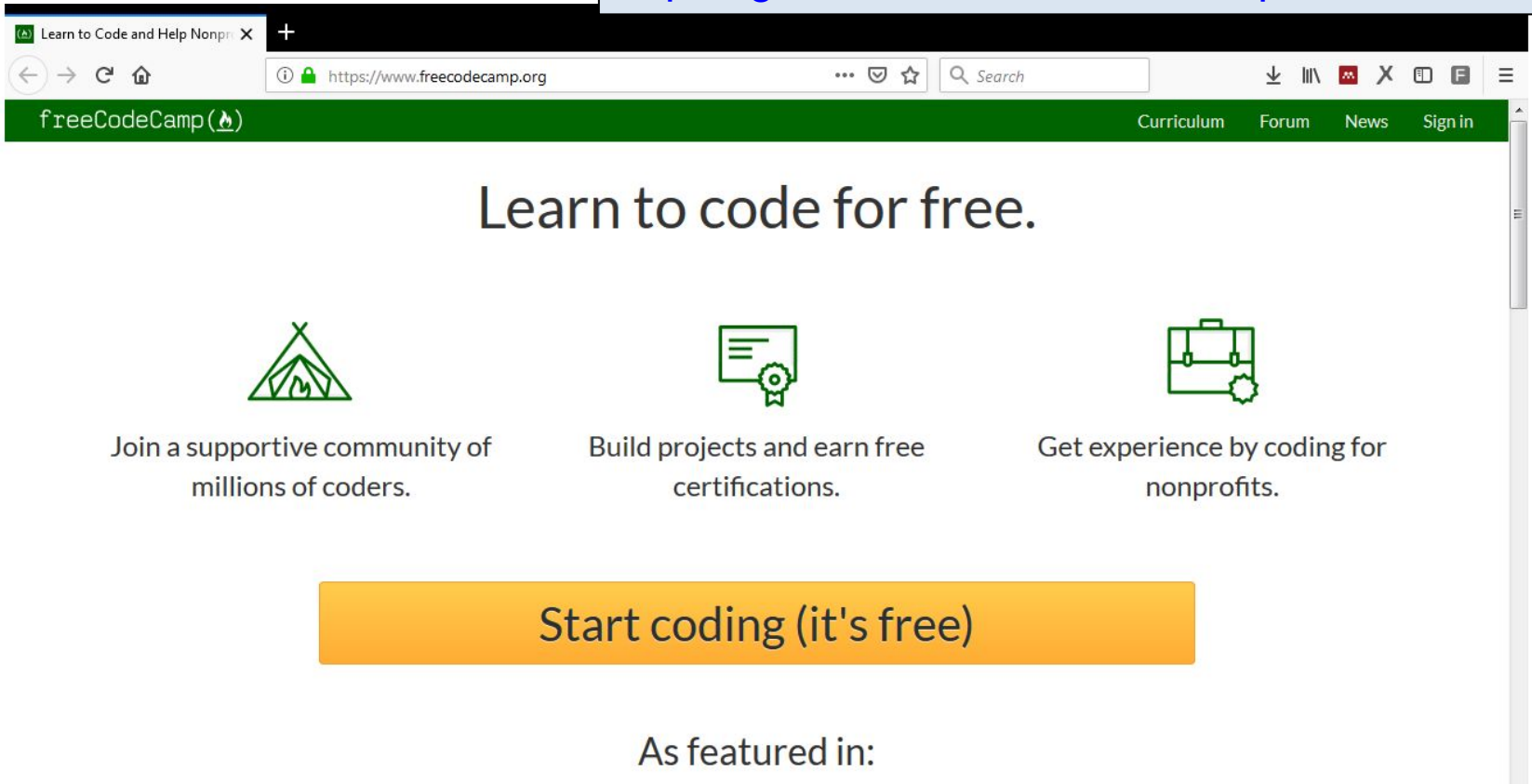
Who uses Github?

https://github.com/topics/scikit-learn



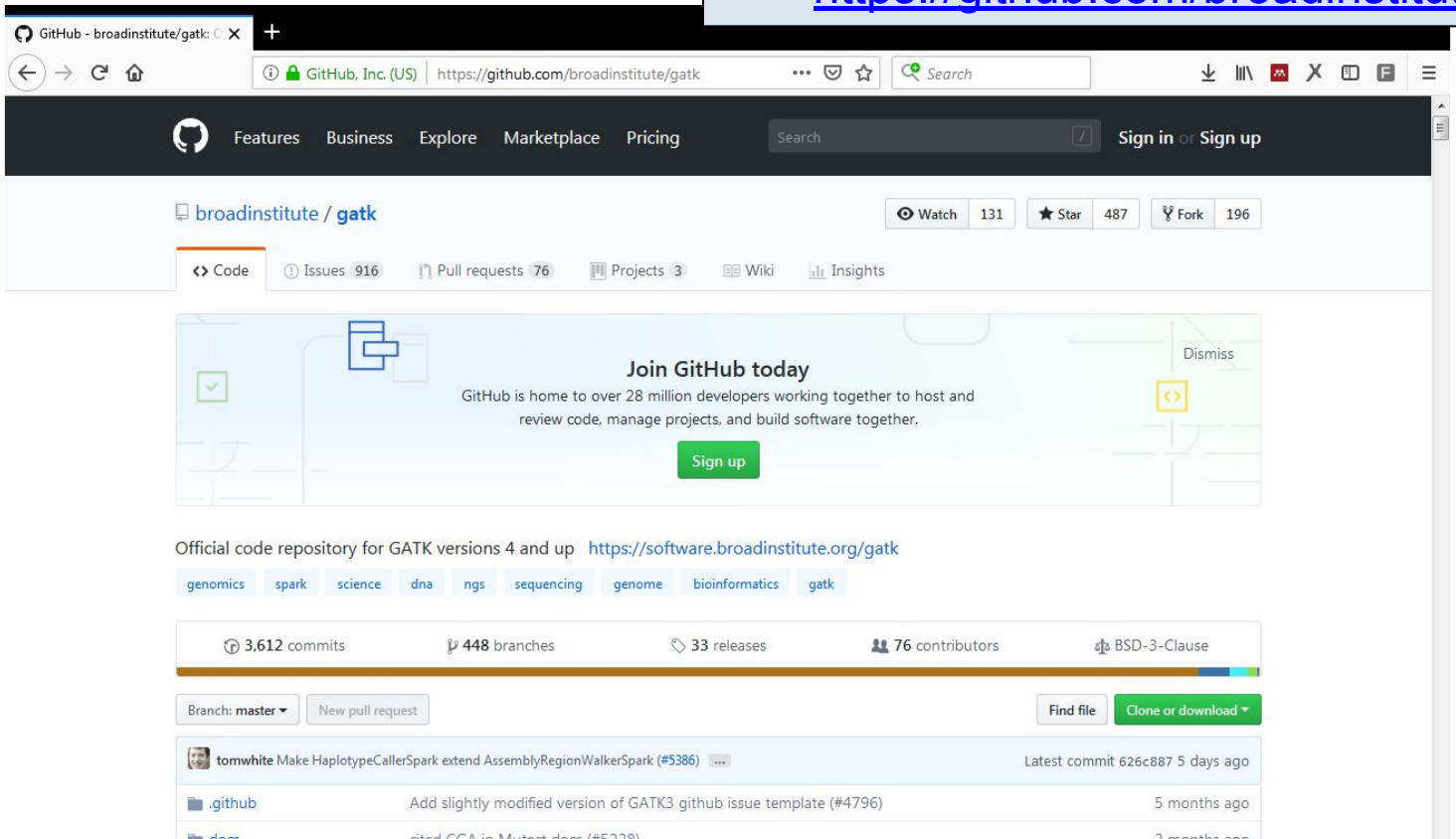
Who uses Github?

https://github.com/freeCodeCamp/freeCodeCamp



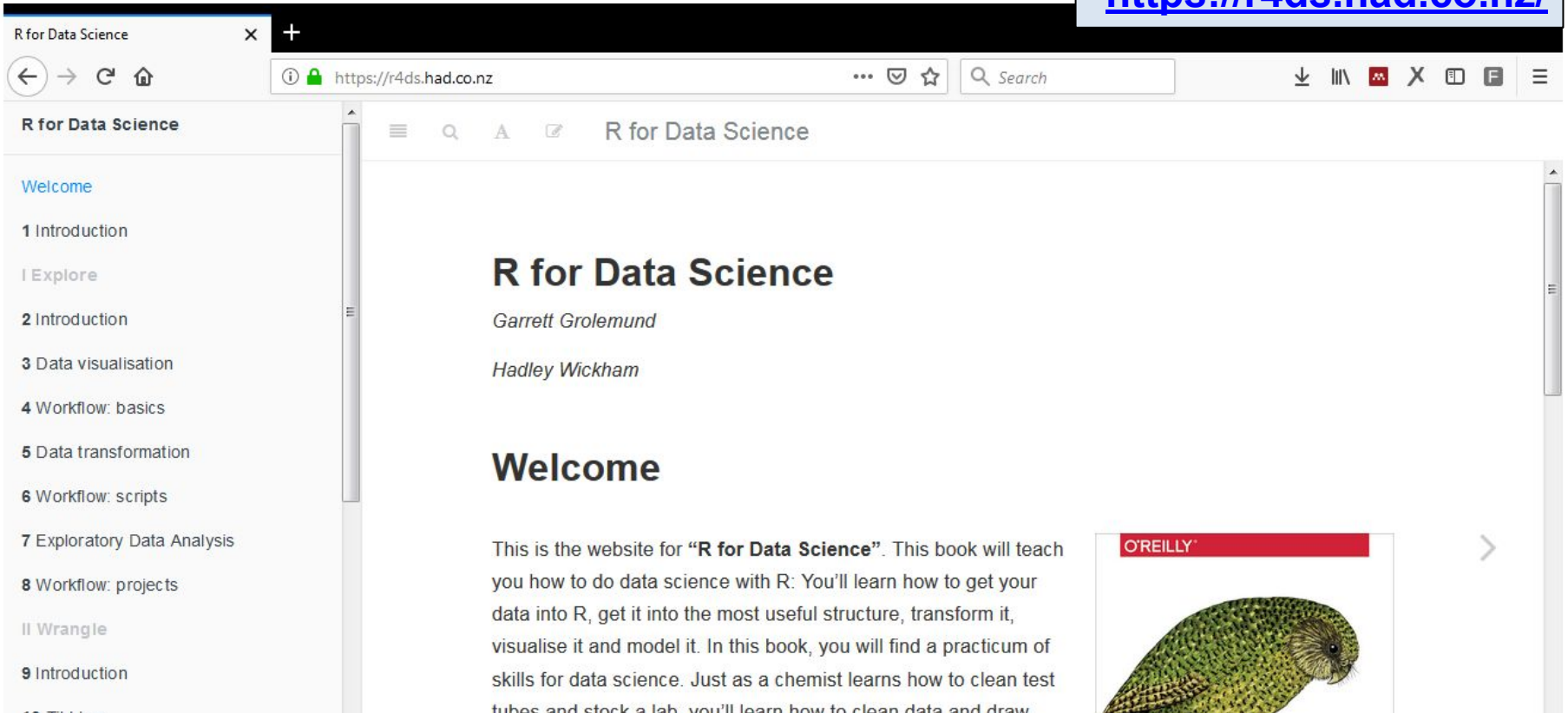
Who uses Github?

https://github.com/broadinstitute/gatk



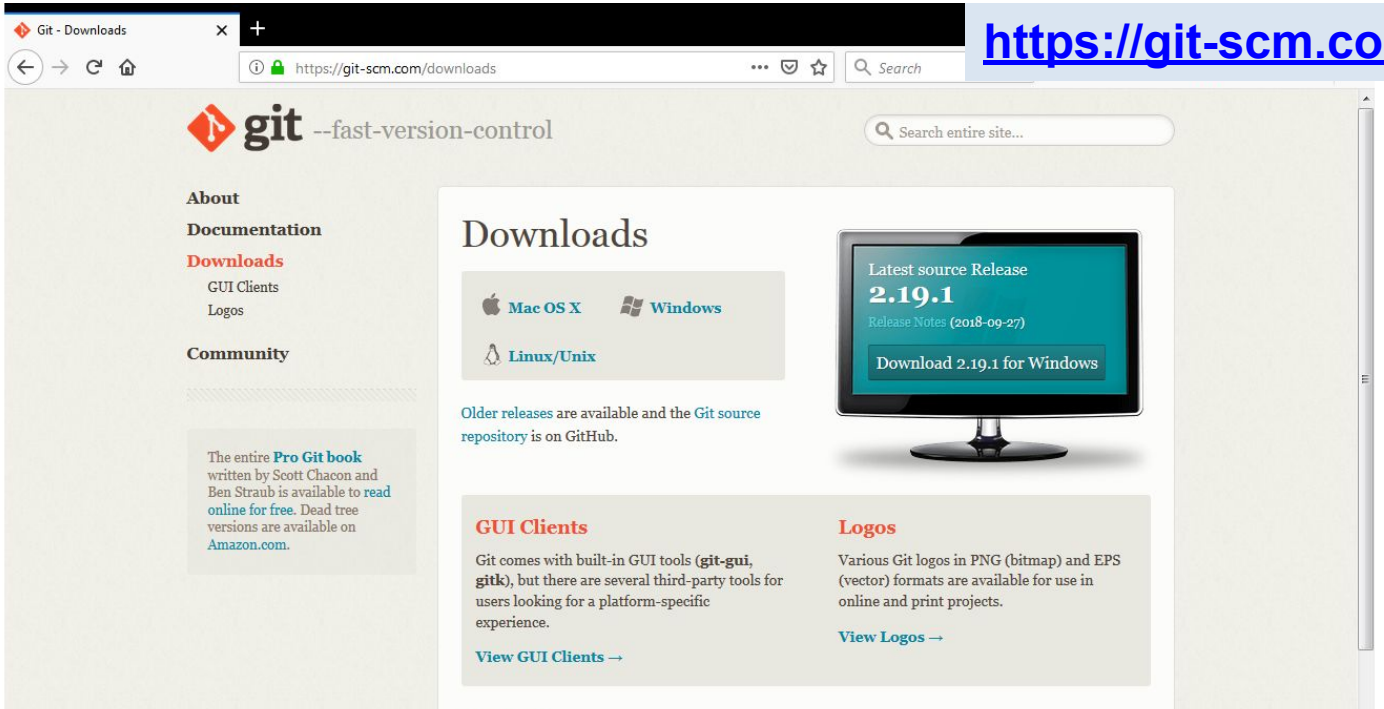
Who uses Github?

https://r4ds.had.co.nz/



Getting started

Step 0: Install git and create a GitHub account

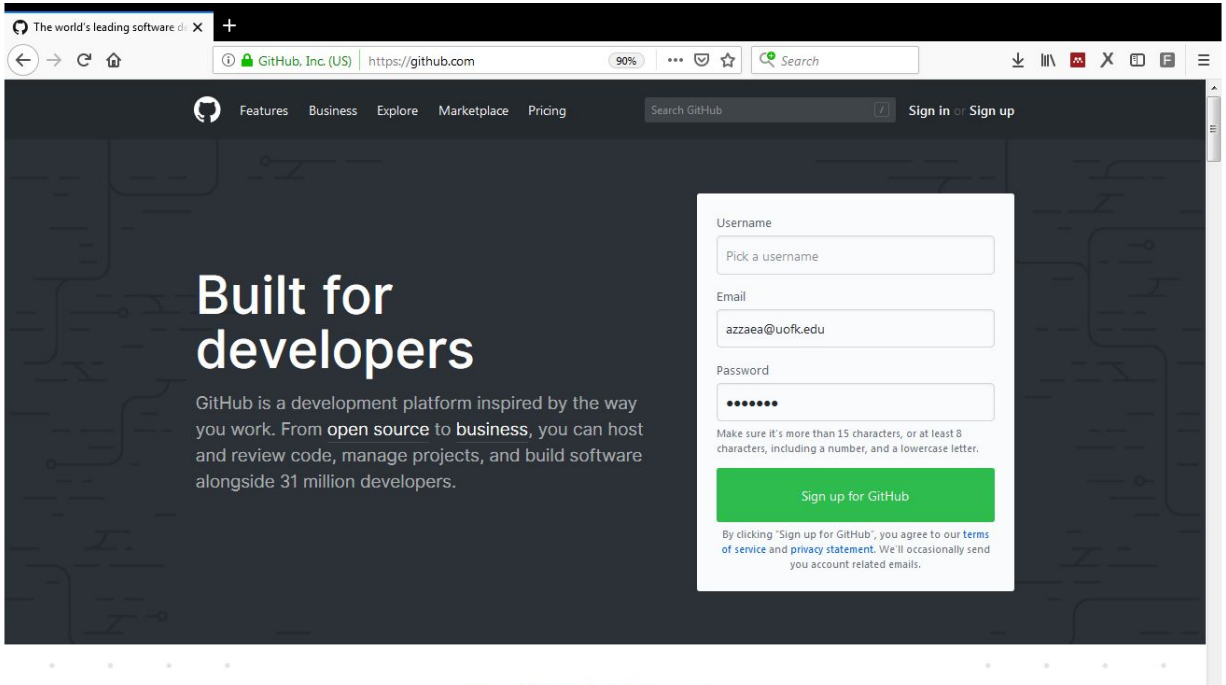


https://git-scm.com/downloads

Getting started

Step 0: Install git and create a GitHub account

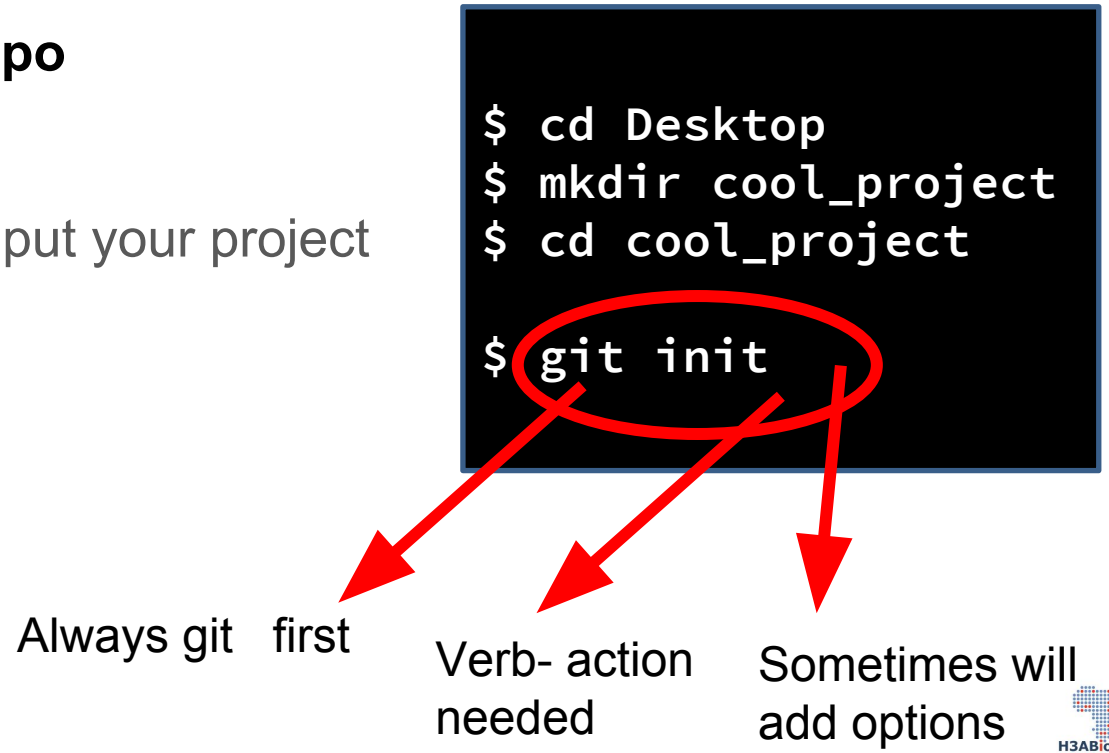
https://github.com/



Getting started

Step 1: Creating a local repo

- Open a terminal
- Go to where you like to put your project
- Create a new folder
- Initialize your repo

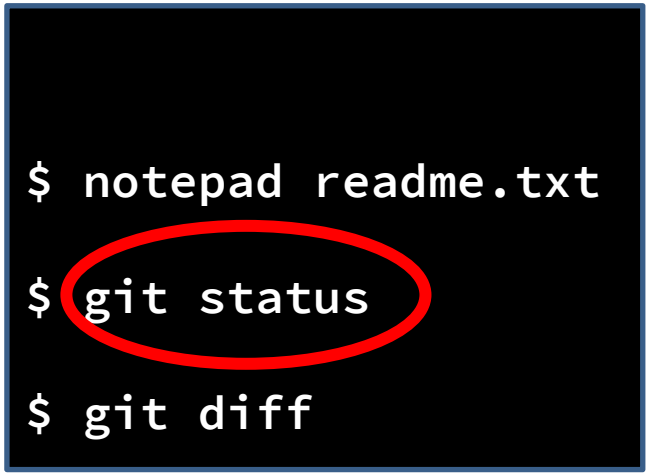


Getting started

Step 2: Add some content

- In your terminal

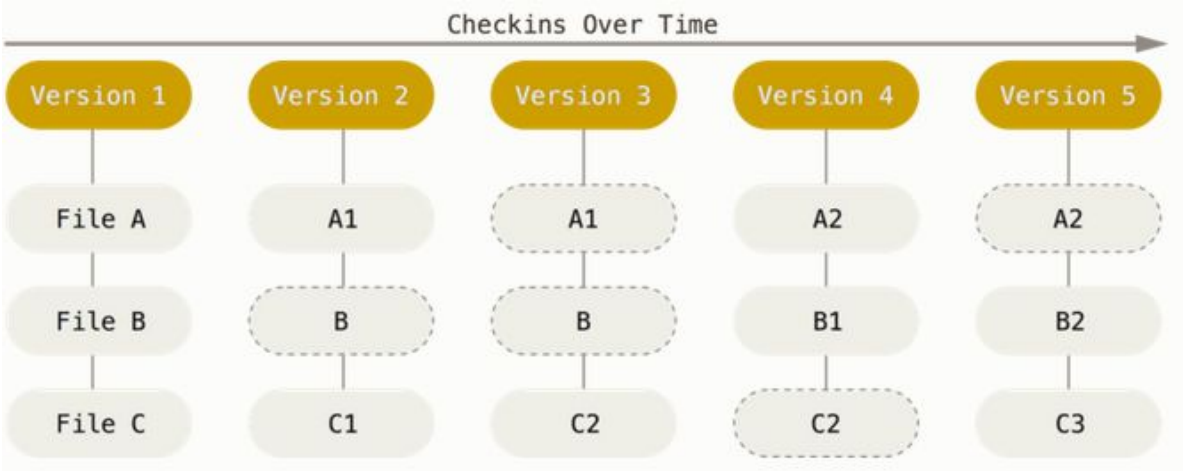
- Create some file
- Check what you did



A few key words

- Commit ≈ *save*
  A record of what you have changed (since last time ….)

- Staging environment/ index:
  to add files to a commit, they need to go to the staging area



https://git-scm.com/book/en/v2/Getting-Started-Git-Basics

A few key words

- Git has 3 main stages for files:

1. “committed”
   files are stored in the local database

2. “modified”
   files changed but not committed

3. “staged”
   files currently marked as modified and will be committed next time

A few key words

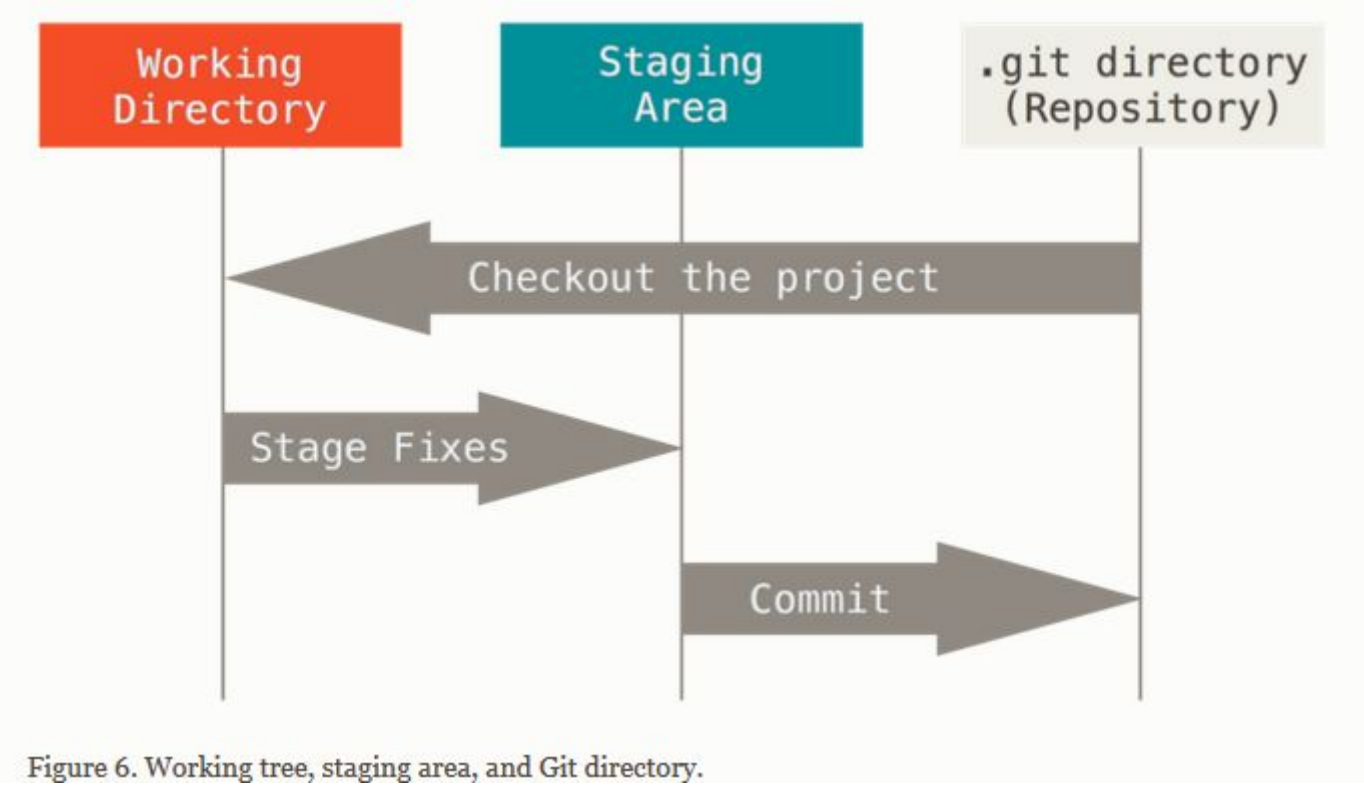
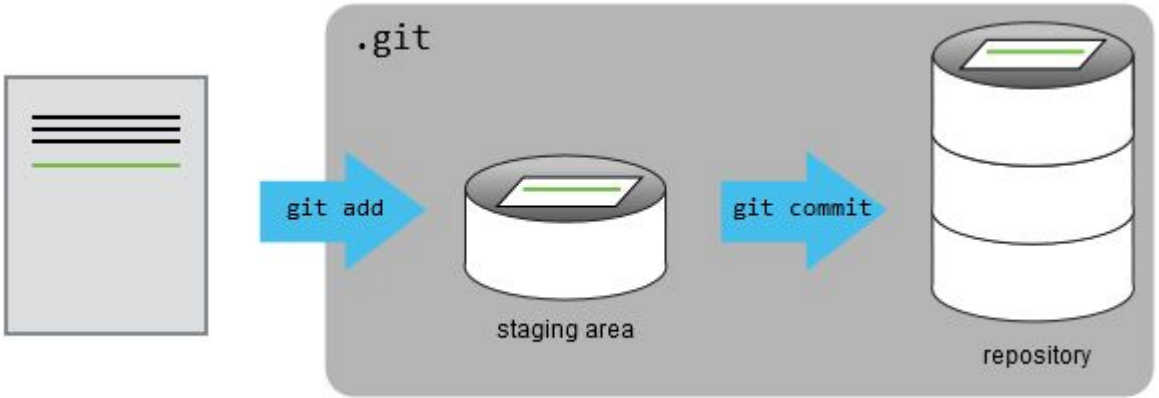


Figure 6. Working tree, staging area, and Git directory.

https://git-scm.com/book/en/v2/Getting-Started-Git-Basics

A few key words



http://swcarpentry.github.io/git-novice/04-changes/index.html

Getting started

Step 3: Add files to the staging area

-
-

```
$ git add readme.txt

$ git commit -m “add my first file”

$ git status
```

Getting started

Step 4: Create a new “branch”

Very handy if you are mainly testing a new feature will live. Roughly speaking, a branch is kind of a “state” of a given project.

Once a branch is “mature”, just “merge” it!

- Create a new branch
- Check what you did

```
$ git checkout -b tst_branch
$ git branch
```

Getting started

Step 5: Add files to your new branch

```
$ notepad readme_branch.txt

$ git add readme_branch.txt

$ git commit -m “add my first file”

$ git status
```

Getting started

Step 6: A few configurations

```
$ git config --global user.name “Azza”
$ git config --global user.email “azzaea@uofk.edu”
```

Getting started

Step 6: A few configurations - line ending

On macOS and Linux:

-
- ```
  $ git config --global core.autocrlf input
  ```

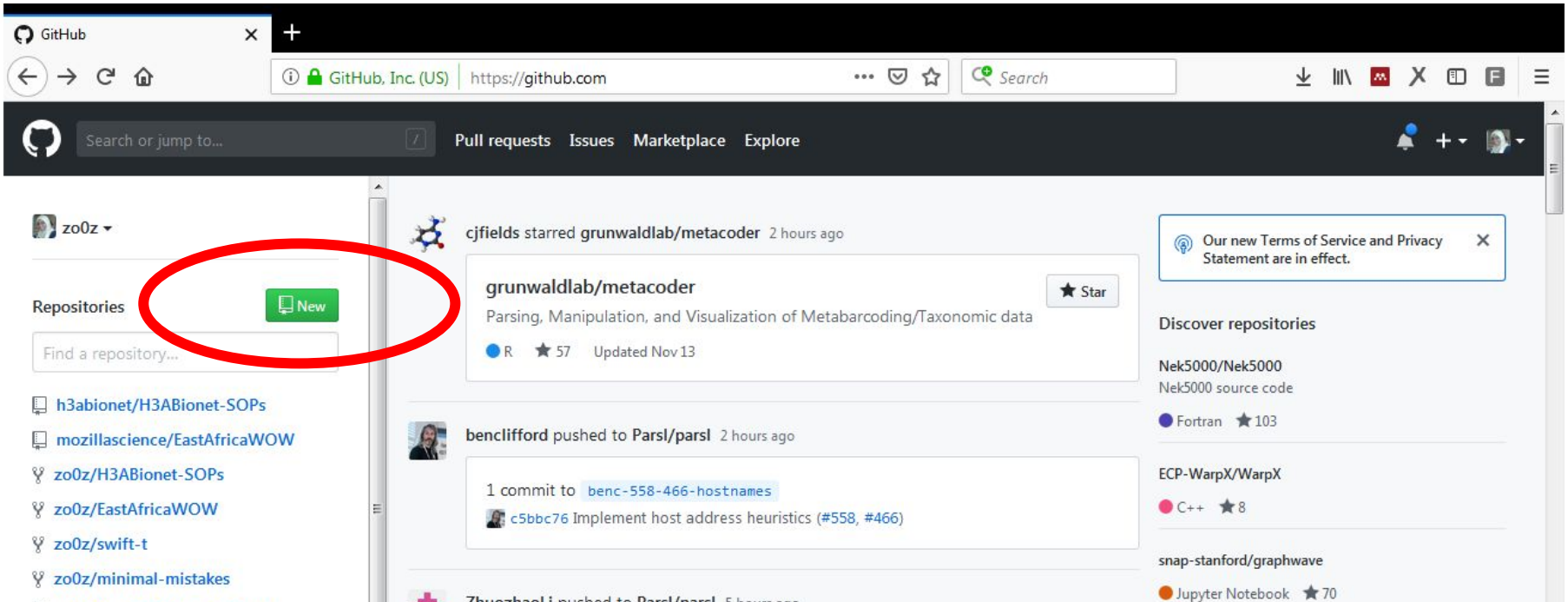
And on Windows:

```
$ git config --global core.autocrlf true
```
```

H3ABioNet

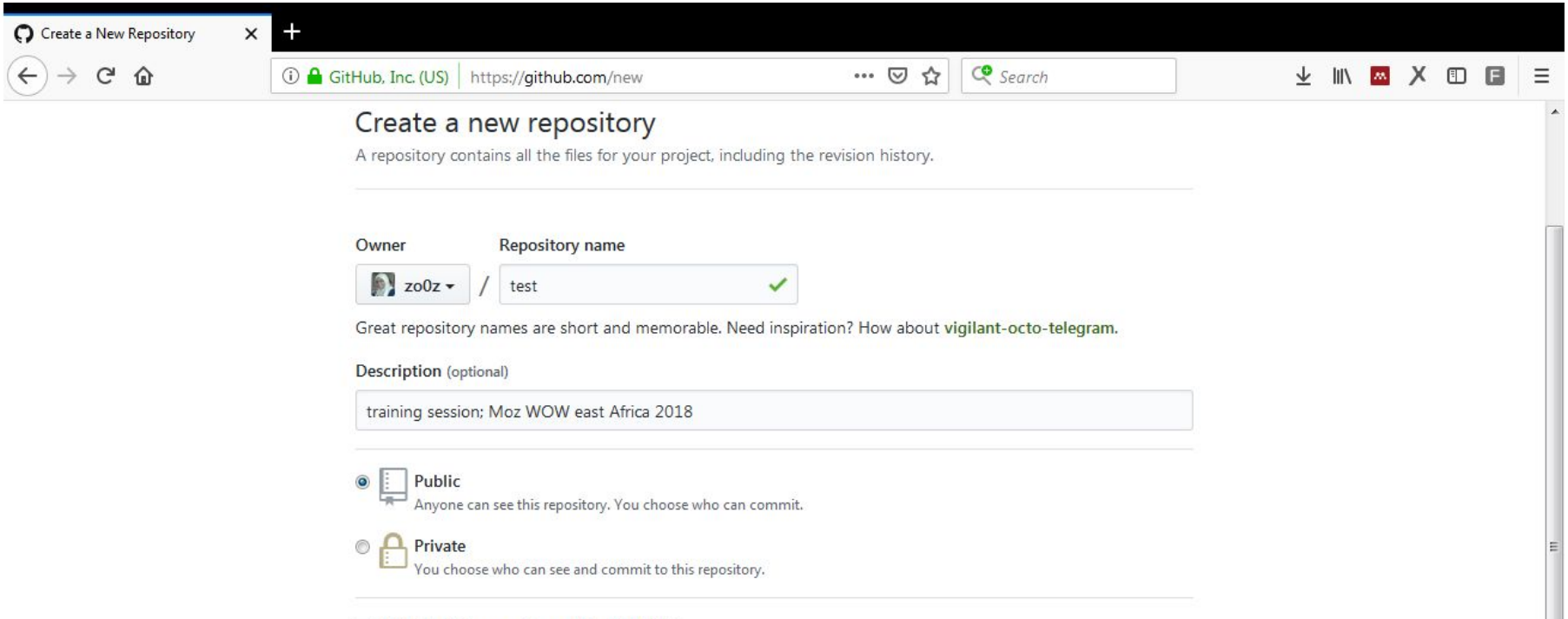
Getting started

Step 7: Create a new GitHub repo- “going global”



Getting started

Step 7: Create a new GitHub repo- “going global”



Getting started

Step 7: Create a new GitHub repo- “going global”

```
$ git remote add origin
http://github.com/zo0z/test.git

$ git push –u origin master
```

Getting started

Step 8: Push a branch to github

This allows you to “publish” your changes.

As you would expect:

```
$ git push origin tst_branch
```

Getting started

Step 9: Create a Pull Request (PR)

A PR is a way to announce (to the repo's owner) you have made changes to code. It helps review code before it is good before going to the master branch

H3ABioNet

Getting started

Step 10: Merge the Pull Request (PR)

H3ABioNet

H3ABioNet

Getting started

Step 11: Sync your local copy

```
$ git pull origin master
```


H3ABioNet

H3ABioNet

Getting started

Step 12: Checking sanity

```
$ git branch master

$ git log
```


H3ABioNet

PLOS | COMPUTATIONAL BIOLOGY

EDITORIAL

Ten Simple Rules for Taking Advantage of Git and GitHub

Yasset Perez-Riverol[1]*, Laurent Gatto[2], Rui Wang[1], Timo Sachsenberg[3], Julian Uszkoreit[4], Felipe da Veiga Leprevost[5], Christian Fufezan[6], Tobias Ternent[1], Stephen J. Eglen[7], Daniel S. Katz[8], Tom J. Pollard[9], Alexander Konovalov[10], Robert M. Flight[11], Kai Blin[12], Juan Antonio Vizcaíno[1]*

Rule 1: Use GitHub to Track Your Projects

PLOS COMPUTATIONAL BIOLOGY

EDITORIAL

Ten Simple Rules for Taking Advantage of Git and GitHub

Yasset Perez-Riverol[1]*, Laurent Gatto[2], Rui Wang[1], Timo Sachsenberg[3], Julian Uszkoreit[4], Felipe da Veiga Leprevost[5], Christian Fufezan[6], Tobias Ternent[1], Stephen J. Eglen[7], Daniel S. Katz[8], Tom J. Pollard[9], Alexander Konovalov[10], Robert M. Flight[11], Kai Blin[12], Juan Antonio Vizcaíno[1]*

Rule 2: GitHub for Single Users, Teams, and Organizations

H3ABioNet

H3ABioNet

PLOS COMPUTATIONAL BIOLOGY

H3ABioNet

EDITORIAL

Ten Simple Rules for Taking Advantage of Git and GitHub

Yasset Perez-Riverol[1]*, Laurent Gatto[2], Rui Wang[1], Timo Sachsenberg[3], Julian Uszkoreit[4], Felipe da Veiga Leprevost[5], Christian Fufezan[6], Tobias Ternent[1], Stephen J. Eglen[7], Daniel S. Katz[8], Tom J. Pollard[9], Alexander Konovalov[10], Robert M. Flight[11], Kai Blin[12], Juan Antonio Vizcaíno[1]*

Rule 3: Developing and Collaborating on New Features: Branching and Forking

H3ABioNet

H3ABioNet

PLOS COMPUTATIONAL BIOLOGY

H3ABioNet

EDITORIAL

Ten Simple Rules for Taking Advantage of Git and GitHub

Yasset Perez-Riverol[1]*, Laurent Gatto[2], Rui Wang[1], Timo Sachsenberg[3], Julian Uszkoreit[4], Felipe da Veiga Leprevost[5], Christian Fufezan[6], Tobias Ternent[1], Stephen J. Eglen[7], Daniel S. Katz[8], Tom J. Pollard[9], Alexander Konovalov[10], Robert M. Flight[11], Kai Blin[12], Juan Antonio Vizcaíno[1]*

Rule 4: Naming Branches and Commits: Tags and Semantic Versions

H3ABioNet

H3ABioNet

PLOS COMPUTATIONAL BIOLOGY

EDITORIAL

Ten Simple Rules for Taking Advantage of Git and GitHub

Yasset Perez-Riverol[1]*, Laurent Gatto[2], Rui Wang[1], Timo Sachsenberg[3], Julian Uszkoreit[4], Felipe da Veiga Leprevost[5], Christian Fufezan[6], Tobias Ternent[1], Stephen J. Eglen[7], Daniel S. Katz[8], Tom J. Pollard[9], Alexander Konovalov[10], Robert M. Flight[11], Kai Blin[12], Juan Antonio Vizcaíno[1]*

Rule 5: Let GitHub Do Some Tasks for You: Integrate

H3ABioNet

H3ABioNet

PLOS COMPUTATIONAL BIOLOGY

EDITORIAL

Ten Simple Rules for Taking Advantage of Git and GitHub

Yasset Perez-Riverol[1]*, Laurent Gatto[2], Rui Wang[1], Timo Sachsenberg[3], Julian Uszkoreit[4], Felipe da Veiga Leprevost[5], Christian Fufezan[6], Tobias Ternent[1], Stephen J. Eglen[7], Daniel S. Katz[8], Tom J. Pollard[9], Alexander Konovalov[10], Robert M. Flight[11], Kai Blin[12], Juan Antonio Vizcaíno[1]*

Rule 6: Let GitHub Do More Tasks for You: Automate

H3ABioNet

H3ABioNet

PLOS COMPUTATIONAL BIOLOGY

EDITORIAL

Ten Simple Rules for Taking Advantage of Git and GitHub

Yasset Perez-Riverol[1]*, Laurent Gatto[2], Rui Wang[1], Timo Sachsenberg[3], Julian Uszkoreit[4], Felipe da Veiga Leprevost[5], Christian Fufezan[6], Tobias Ternent[1], Stephen J. Eglen[7], Daniel S. Katz[8], Tom J. Pollard[9], Alexander Konovalov[10], Robert M. Flight[11], Kai Blin[12], Juan Antonio Vizcaíno[1]*

Rule 7: Use GitHub to Openly and Collaboratively Discuss, Address, and Close Issues

H3ABioNet

H3ABioNet

PLOS COMPUTATIONAL BIOLOGY

H3ABioNet

EDITORIAL

Ten Simple Rules for Taking Advantage of Git and GitHub

Yasset Perez-Riverol[1]*, Laurent Gatto[2], Rui Wang[1], Timo Sachsenberg[3], Julian Uszkoreit[4], Felipe da Veiga Leprevost[5], Christian Fufezan[6], Tobias Ternent[1], Stephen J. Eglen[7], Daniel S. Katz[8], Tom J. Pollard[9], Alexander Konovalov[10], Robert M. Flight[11], Kai Blin[12], Juan Antonio Vizcaíno[1]*

Rule 8: Make Your Code Easily Citable, and Cite Source Code!

H3ABioNet

H3ABioNet

PLOS COMPUTATIONAL BIOLOGY

EDITORIAL

Ten Simple Rules for Taking Advantage of Git and GitHub

Yasset Perez-Riverol[1]*, Laurent Gatto[2], Rui Wang[1], Timo Sachsenberg[3], Julian Uszkoreit[4], Felipe da Veiga Leprevost[5], Christian Fufezan[6], Tobias Ternent[1], Stephen J. Eglen[7], Daniel S. Katz[8], Tom J. Pollard[9], Alexander Konovalov[10], Robert M. Flight[11], Kai Blin[12], Juan Antonio Vizcaíno[1]*

Rule 9: Promote and Discuss Your Projects: Web Page and More

H3ABioNet

H3ABioNet

PLOS COMPUTATIONAL BIOLOGY

EDITORIAL

Ten Simple Rules for Taking Advantage of Git and GitHub

Yasset Perez-Riverol[1]*, Laurent Gatto[2], Rui Wang[1], Timo Sachsenberg[3], Julian Uszkoreit[4], Felipe da Veiga Leprevost[5], Christian Fufezan[6], Tobias Ternent[1], Stephen J. Eglen[7], Daniel S. Katz[8], Tom J. Pollard[9], Alexander Konovalov[10], Robert M. Flight[11], Kai Blin[12], Juan Antonio Vizcaíno[1]*

Rule 10: Use GitHub to Be Social: Follow and Watch

H3ABioNet

H3ABioNet

Credits and resources

- https://product.hubspot.com/blog/git-and-github-tutorial-for-beginners
- http://swcarpentry.github.io/git-novice/

Images:

- https://studyguide.itu.dk/cs/campus-facilities/it-facilities/github
- http://phdcomics.com/comics/archive_print.php?comicid=1531