



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/19UsTRcuNkWm8fCJUBgps8BP5xwI96A34RmGQC4lr9Ww/edit#slide=id.g25c238d0a2_0_23
- https://docs.google.com/presentation/d/1LNvTgx7fqOSZEDYKkitNqVpNUqXUIK3o502RSVnHAQY/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
Authorea- 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 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/>

East Africa Open Science Tour 2018



Schedule

Notes

Help


Trainers

Venues


East Africa Open Science Tour 2018

Uganda, Ethiopia, Sudan and Kenya

moz://a



SageBionetworks



H3ABioNet


Pan African Bioinformatics Network for H3Africa


Mozilla and H3ABioNet in collaboration with SageBionetworks are running a series of 2-days Open Science trainings in East Africa. These 2 days trainings aim to raise awareness, increase skills around open research and reproducibility in Science, tools and platforms, and ultimately help scientist and researchers overcome the fears and challenges.

November 12-13 UVRI: Entebbe, Uganda

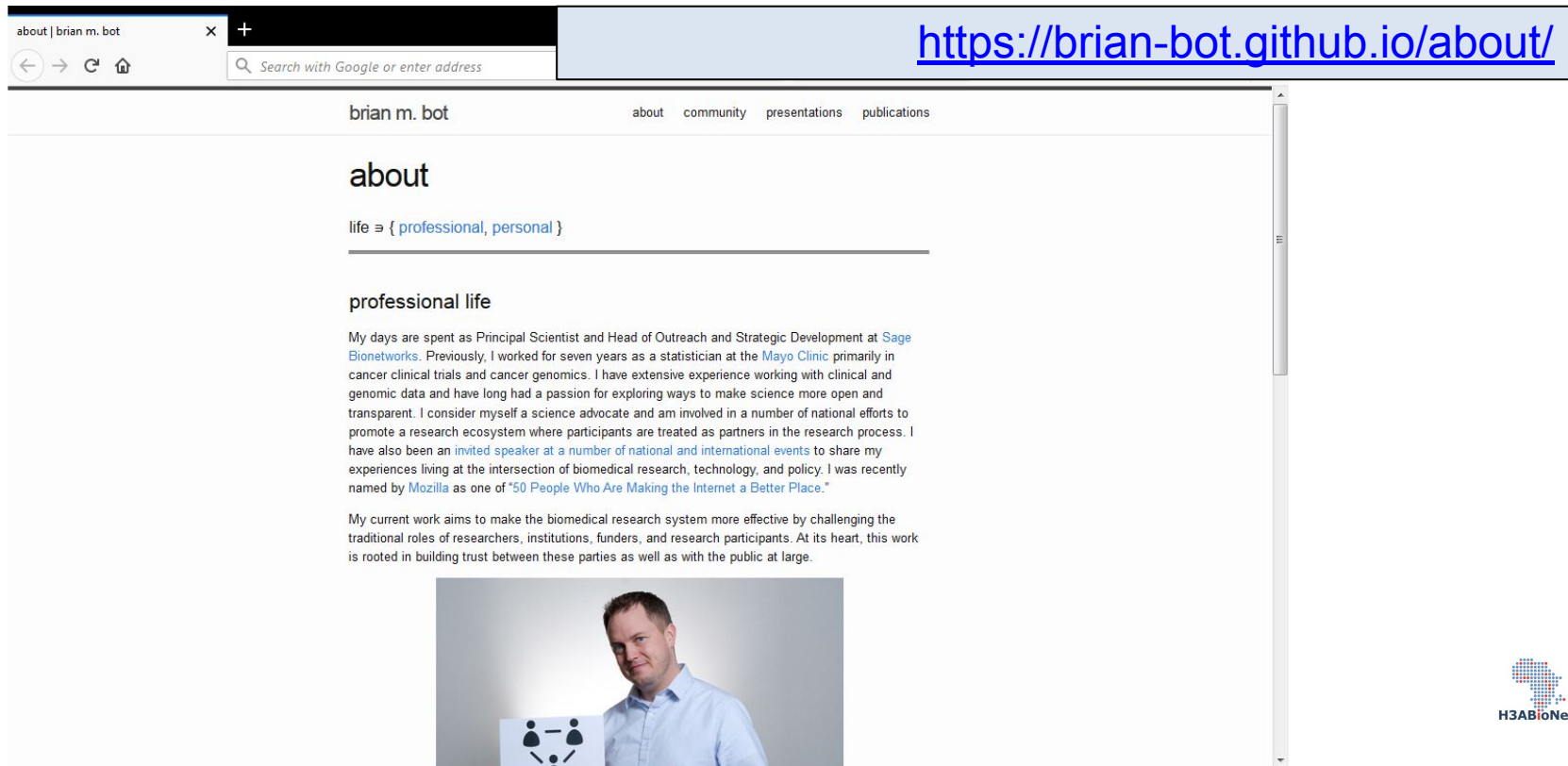
November 15-16 AASTU: Addis Ababa, Ethiopia

November 18-19 UofK: Khartoum, Sudan





Who uses Github?



The screenshot shows a web browser window displaying the GitHub profile page for 'brian m. bot'. The browser's address bar shows the URL <https://brian-bot.github.io/about/>. The page header includes the name 'brian m. bot' and navigation links for 'about', 'community', 'presentations', and 'publications'. The main content area is titled 'about' and features a link 'life ⇒ { professional, personal }'. Under the 'professional life' section, the text describes the user's role as Principal Scientist and Head of Outreach and Strategic Development at Sage Bionetworks, and their previous work as a statistician at the Mayo Clinic. It also mentions their involvement in national efforts to promote a research ecosystem and their recognition by Mozilla as one of '50 People Who Are Making the Internet a Better Place.' A photograph of the user, a man in a light blue shirt, is shown at the bottom, holding a sign with a logo of three stylized figures.

about | brian m. bot

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

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>

The screenshot shows the GitHub interface for the 'scikit-learn' topic. At the top, the browser address bar shows the URL 'https://github.com/topics/scikit-learn'. The GitHub navigation bar includes links for Features, Business, Explore, Marketplace, and Pricing, along with a search bar and 'Sign in' or 'Sign up' buttons. The main content area is titled 'scikit-learn' and includes a description: 'scikit-learn is a widely-used Python module for classic machine learning. It is built on top of SciPy.' Below this, there are buttons for 'Sign up for GitHub' and 'sign in to edit this page', and metadata: 'Created by David Cournapeau', 'Released January 05, 2010', and 'Latest release about 1 year ago'. A section for 'REPOSITORIES' shows 1,831 results, with filters for 'Language: All' and 'Sort: Best match'. The first repository listed is 'donnemartin / data-science-ipython-notebooks' with 14.2k stars. The second repository is 'jakevdp / PythonDataScienceHandbook' with 12.4k stars. On the right, there is a 'LEARN ABOUT SCIKIT-LEARN' section with links to 'scikit-learn/scikit-learn', 'scikit-learn.org', and 'Wikipedia'. Below that is a 'RELATED TOPICS' section with tags for 'sklearn', 'python', 'scikit', 'machine-learning', 'dataset', 'notebook', 'ipython-notebook', and 'numpy'.

Topic: scikit-learn · GitHub

← → ↻ 🏠 ⓘ **GitHub, Inc. (US)** <https://github.com/topics/scikit-learn> ... 🔒 ☆ 🔍 Search

🔍 Search Sign in or Sign up

scikit-learn

scikit-learn is a widely-used Python module for classic machine learning. It is built on top of SciPy.

[Sign up for GitHub](#) or [sign in](#) to edit this page Created by David Cournapeau Released January 05, 2010 Latest release [about 1 year ago](#)

REPOSITORIES 1,831 Language: All Sort: Best match

donnemartin / data-science-ipython-notebooks ★ 14.2k

Data science Python notebooks: Deep learning (TensorFlow, Theano, Caffe, Keras), scikit-learn, Kaggle, big data (Spar...

python machine-learning deep-learning data-science big-data aws tensorflow

Python Updated on Sep 15 3 issues need help

jakevdp / PythonDataScienceHandbook ★ 12.4k

Python Data Science Handbook: full text in Jupyter Notebooks

scikit-learn numpy python jupyter-notebook matplotlib pandas

LEARN ABOUT SCIKIT-LEARN

- scikit-learn/scikit-learn
- scikit-learn.org
- Wikipedia

RELATED TOPICS See more topics

- sklearn python scikit
- machine-learning dataset notebook
- ipython-notebook numpy

Who uses Github?

<https://github.com/freeCodeCamp/freeCodeCamp>



Learn to code for free.



Join a supportive community of millions of coders.



Build projects and earn free certifications.



Get experience by coding for nonprofits.

Start coding (it's free)

As featured in:

Who uses Github?

<https://github.com/broadinstitute/gatk>

The screenshot shows the GitHub repository page for `broadinstitute/gatk`. The browser address bar displays the URL `https://github.com/broadinstitute/gatk`. The repository page includes a navigation bar with links for Features, Business, Explore, Marketplace, and Pricing, along with a search bar and a 'Sign in or Sign up' button. The repository name `broadinstitute / gatk` is shown, along with statistics: 131 Watchers, 487 Stars, and 196 Forks. Below this, there are tabs for Code, Issues (916), Pull requests (76), Projects (3), Wiki, and Insights. A large banner encourages users to 'Join GitHub today' with a 'Sign up' button. The page identifies the repository as the 'Official code repository for GATK versions 4 and up' and provides a link to `https://software.broadinstitute.org/gatk`. A tag cloud lists various terms: genomics, spark, science, dna, ngs, sequencing, genome, bioinformatics, and gatk. At the bottom, it shows repository statistics: 3,612 commits, 448 branches, 33 releases, 76 contributors, and a BSD-3-Clause license. A 'Branch: master' dropdown and a 'New pull request' button are visible. A 'Find file' button and a 'Clone or download' button are also present. The commit history shows a recent commit by `tomwhite` titled 'Make HaplotypeCallerSpark extend AssemblyRegionWalkerSpark (#5386)' with the latest commit made 5 days ago. Other commits include 'Add slightly modified version of GATK3 github issue template (#4796)' from 5 months ago and 'cited CCA in Mutect docs (#5238)' from 2 months ago.

GitHub - broadinstitute/gatk

Features Business Explore Marketplace Pricing Search Sign in or Sign up

broadinstitute / gatk Watch 131 Star 487 Fork 196

Code Issues 916 Pull requests 76 Projects 3 Wiki Insights

Join GitHub today

GitHub is home to over 28 million developers working together to host and review code, manage projects, and build software together.

Sign up

Dismiss

Official code repository for GATK versions 4 and up <https://software.broadinstitute.org/gatk>

genomics spark science dna ngs sequencing genome bioinformatics gatk

3,612 commits 448 branches 33 releases 76 contributors BSD-3-Clause

Branch: master New pull request Find file Clone or download

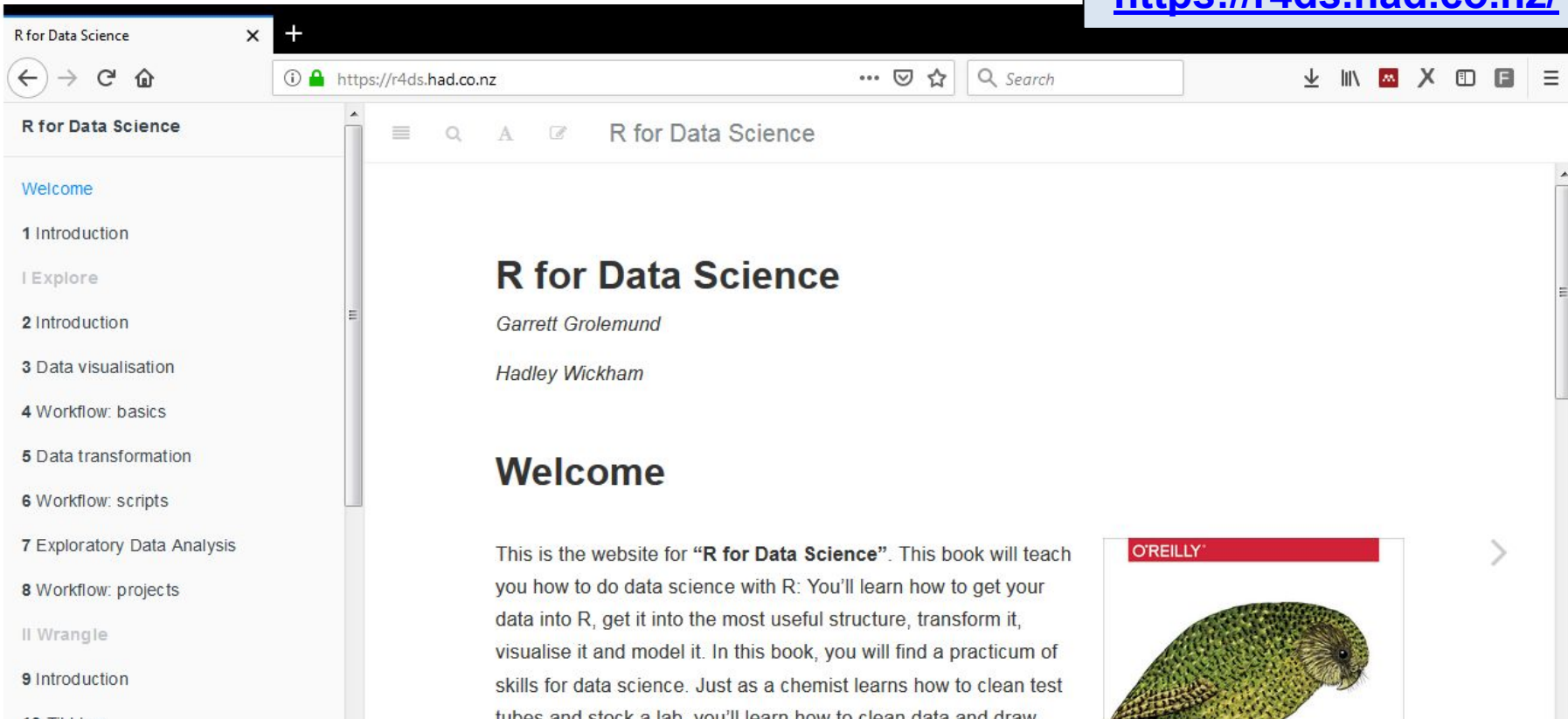
tomwhite Make HaplotypeCallerSpark extend AssemblyRegionWalkerSpark (#5386) Latest commit 626c887 5 days ago

.github Add slightly modified version of GATK3 github issue template (#4796) 5 months ago

docs cited CCA in Mutect docs (#5238) 2 months ago

Who uses Github?

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



The screenshot shows a web browser displaying the 'R for Data Science' website. The browser's address bar shows the URL 'https://r4ds.had.co.nz/'. The website has a dark header with the title 'R for Data Science'. On the left, there is a sidebar with a table of contents. The main content area features the title 'R for Data Science' by Garrett G. Grolemund and Hadley Wickham, followed by a 'Welcome' section. A small O'Reilly book cover is visible in the bottom right corner.

R for Data Science

Welcome

- 1 Introduction
- I Explore
- 2 Introduction
- 3 Data visualisation
- 4 Workflow: basics
- 5 Data transformation
- 6 Workflow: scripts
- 7 Exploratory Data Analysis
- 8 Workflow: projects
- II Wrangle
- 9 Introduction
- 10 Tidy

R for Data Science


Garrett Grolemund

Hadley Wickham

Welcome

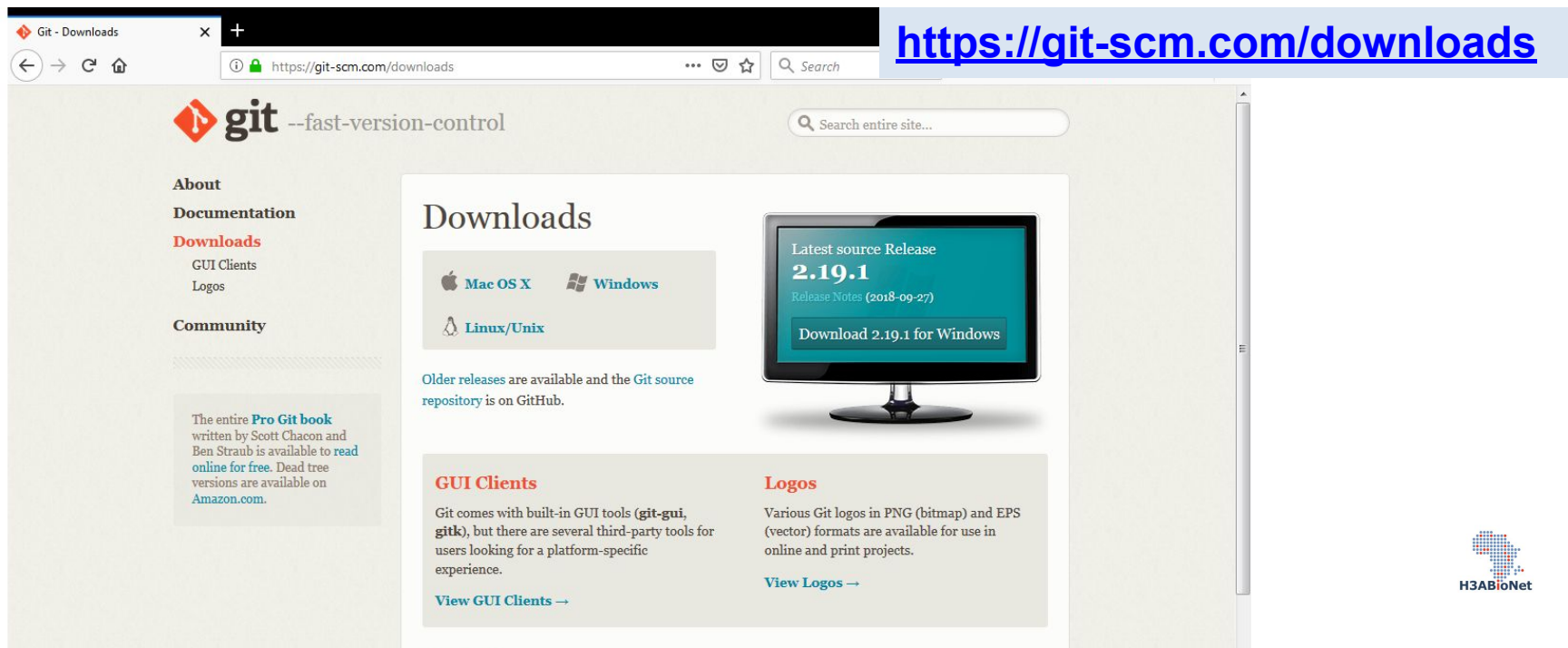
This is the website for “**R for Data Science**”. This book will teach you how to do data science with R: You’ll learn how to get your data into R, get it into the most useful structure, transform it, visualise it and model it. In this book, you will find a practicum of skills for data science. Just as a chemist learns how to clean test tubes and stock a lab, you’ll learn how to clean data and draw

O'REILLY



Getting started

Step 0: Install git and create a GitHub account



The screenshot shows a web browser window with the address bar displaying <https://git-scm.com/downloads>. The page title is "Git - Downloads". The main content area features the Git logo and the tagline "--fast-version-control". On the left, there is a sidebar with links for "About", "Documentation", "Downloads" (highlighted), "GUI Clients", "Logos", and "Community". The "Downloads" section includes links for "Mac OS X", "Windows", and "Linux/Unix". A central monitor graphic displays the "Latest source Release 2.19.1" and a "Download 2.19.1 for Windows" button. Below this, a text box states "Older releases are available and the Git source repository is on GitHub." The bottom section is divided into "GUI Clients" and "Logos", both with "View" links. A footer note mentions a "Pro Git book" available online for free.

<https://git-scm.com/downloads>

git --fast-version-control

Search entire site...

About

Documentation

Downloads

GUI Clients

Logos

Community

The entire [Pro Git book](#) written by Scott Chacon and Ben Straub is available to [read online for free](#). Dead tree versions are available on [Amazon.com](#).

Downloads

Mac OS X Windows

Linux/Unix

Latest source Release
2.19.1
Release Notes (2018-09-27)
Download 2.19.1 for Windows

Older releases are available and the Git source repository is on GitHub.

GUI Clients

Git comes with built-in GUI tools ([git-gui](#), [gitk](#)), but there are several third-party tools for users looking for a platform-specific experience.
[View GUI Clients →](#)

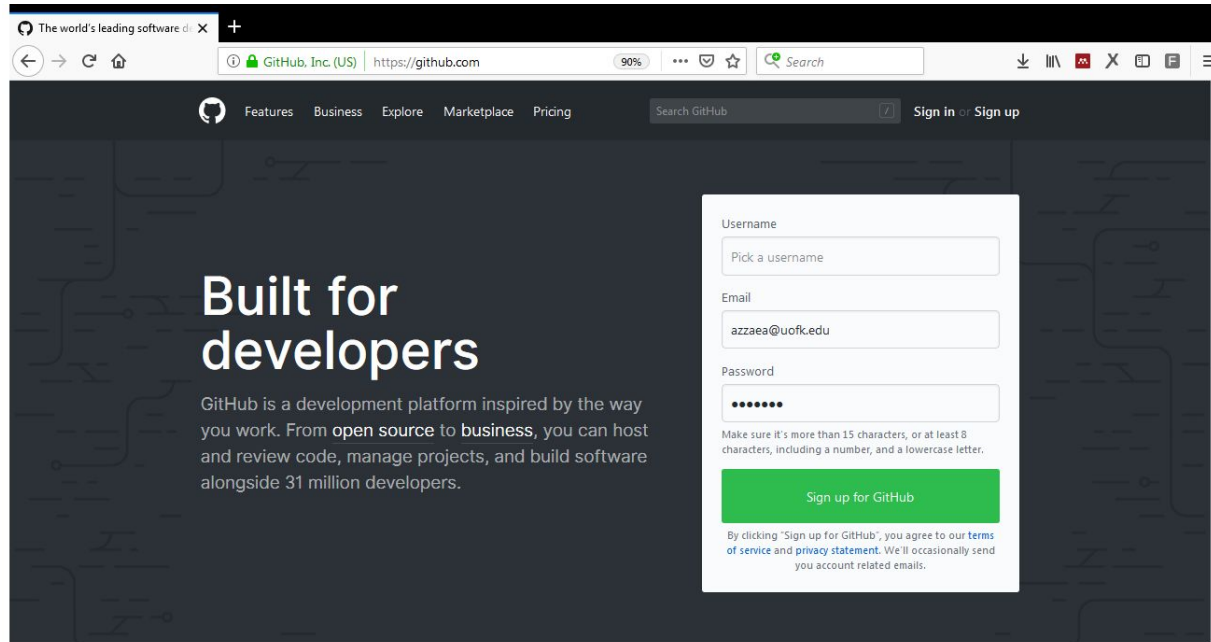
Logos

Various Git logos in PNG (bitmap) and EPS (vector) formats are available for use in online and print projects.
[View Logos →](#)

Getting started

Step 0: Install git and create a GitHub account

<https://github.com/>



The screenshot shows the GitHub homepage with a sign-up modal open. The modal contains the following fields and text:

- Username:** A text input field with the placeholder "Pick a username".
- Email:** A text input field containing the email address "azzaea@uofk.edu".
- Password:** A password input field with masked characters "••••••".
- Sign up for GitHub:** A green button to complete the registration.
- Terms and Conditions:** A note stating, "By clicking 'Sign up for GitHub', you agree to our [terms of service](#) and [privacy statement](#). We'll occasionally send you account related emails."

The background of the page features the text "Built for developers" and a description of GitHub as a development platform.

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

```
$ cd Desktop  
$ mkdir cool_project  
$ cd cool_project  
$ git init
```

Always git first

Verb- action
needed

Sometimes will
add options

Getting started

Step 2: Add some content

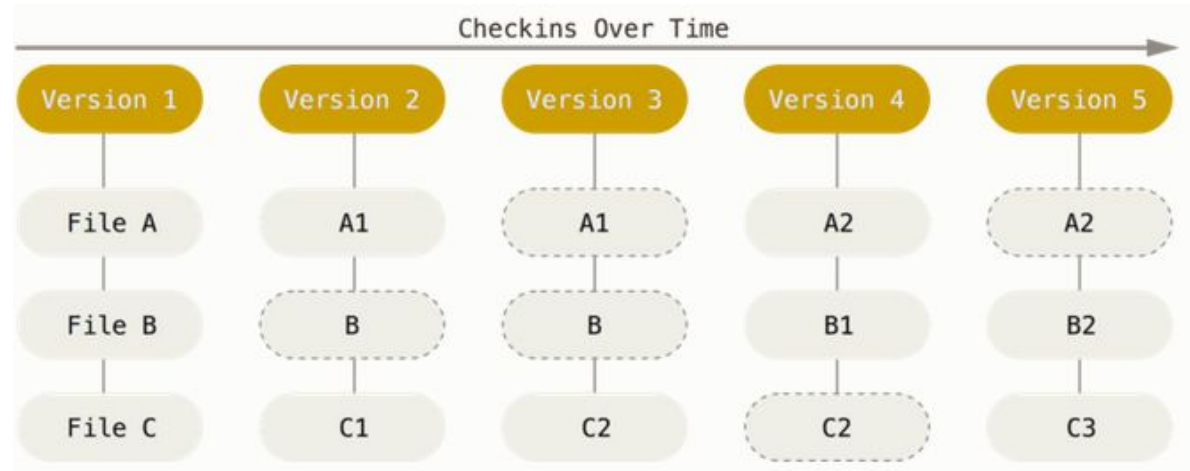
- In your terminal
- Create some file
- Check what you did

```
$ notepad readme.txt  
$ git status  
$ git diff
```

A few key words

- Commit \approx *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

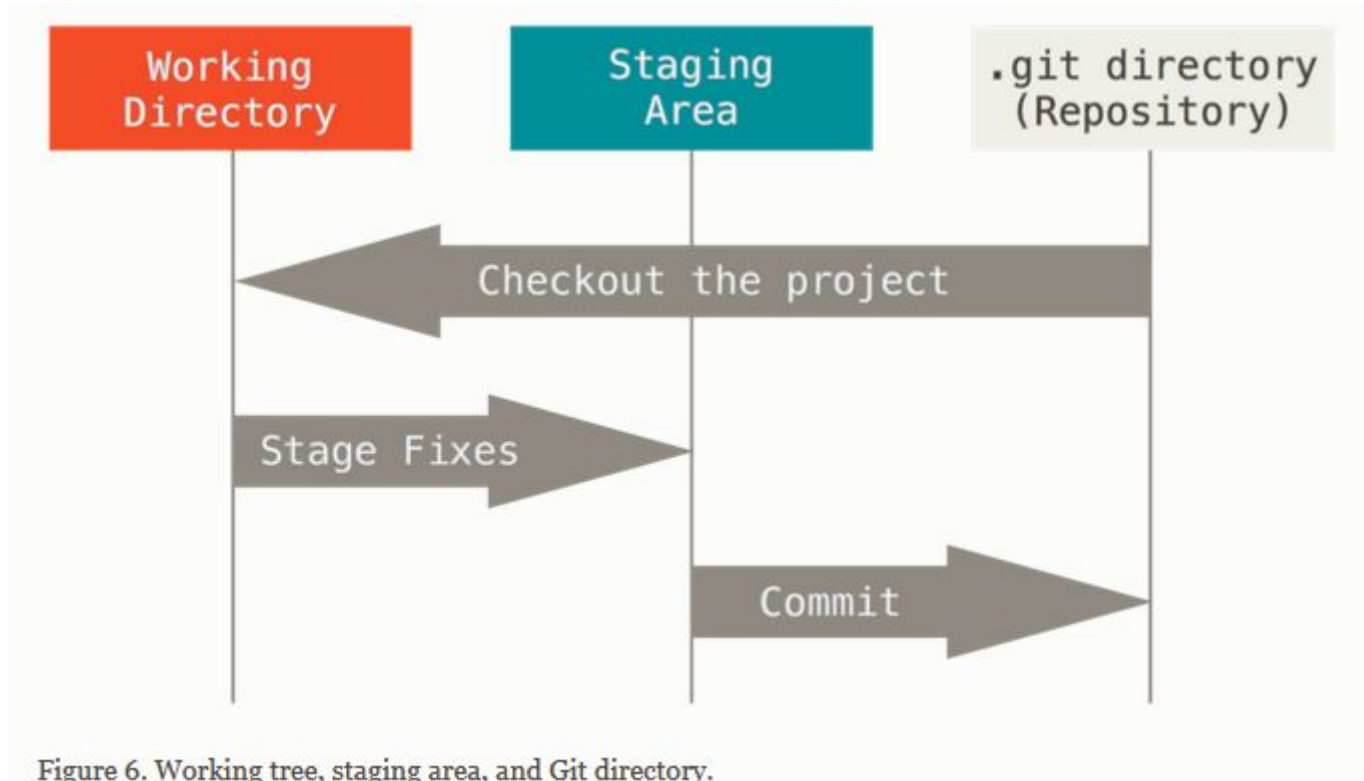
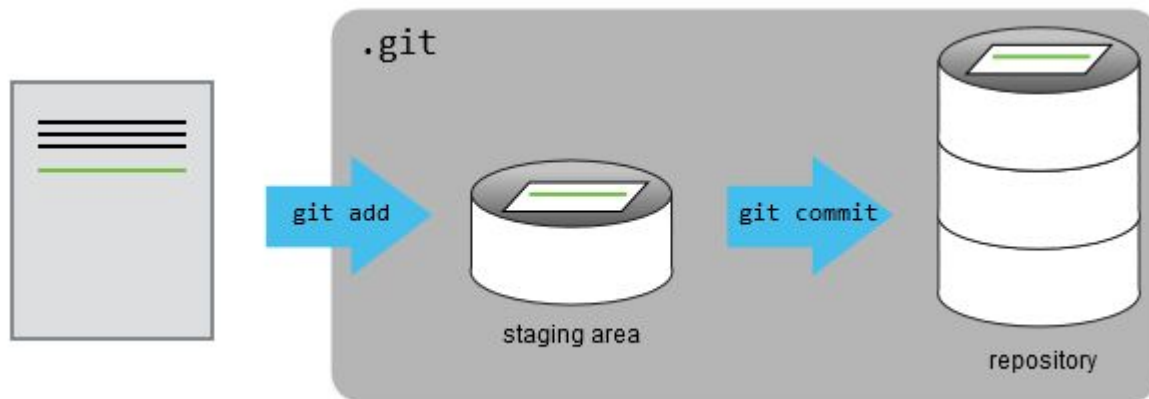


Figure 6. Working tree, staging area, and Git directory.

<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
```

Getting started

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

The screenshot shows the GitHub homepage in a web browser. The browser's address bar displays 'https://github.com'. The GitHub navigation bar includes a search bar, 'Pull requests', 'Issues', 'Marketplace', and 'Explore'. On the left sidebar, under the 'Repositories' section, a green 'New' button is circled in red. Below this button is a search bar labeled 'Find a repository...'. The main content area features a feed of repository activity, including 'cjfields starred grunwaldlab/metacoder' and 'benclifford pushed to Parsl/parsl'. A right sidebar contains a notification about the Terms of Service and Privacy Statement, and a 'Discover repositories' section listing various projects like 'Nek5000/Nek5000', 'Fortran', 'ECP-WarpX/WarpX', and 'snap-stanford/graphwave'.

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



# Getting started

## Step 10: Merge the Pull Request (**PR**)

# Getting started

## Step 11: Sync your local copy

```
$ git pull origin master
```

# Getting started

## Step 12: Checking sanity

```
$ git branch master
```

```
$ git log
```

EDITORIAL

# Ten Simple Rules for Taking Advantage of Git and GitHub

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

 **Rule 1: Use GitHub to Track Your Projects**

EDITORIAL

## Ten Simple Rules for Taking Advantage of Git and GitHub

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

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

EDITORIAL

## Ten Simple Rules for Taking Advantage of Git and GitHub

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

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

EDITORIAL

## Ten Simple Rules for Taking Advantage of Git and GitHub

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

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

EDITORIAL

## Ten Simple Rules for Taking Advantage of Git and GitHub

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

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



EDITORIAL

## Ten Simple Rules for Taking Advantage of Git and GitHub

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

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

EDITORIAL

## Ten Simple Rules for Taking Advantage of Git and GitHub

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

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

EDITORIAL

## Ten Simple Rules for Taking Advantage of Git and GitHub

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

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

EDITORIAL

## Ten Simple Rules for Taking Advantage of Git and GitHub

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

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

EDITORIAL

## Ten Simple Rules for Taking Advantage of Git and GitHub

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

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

# 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?comcid=1531](http://phdcomics.com/comics/archive_print.php?comcid=1531)