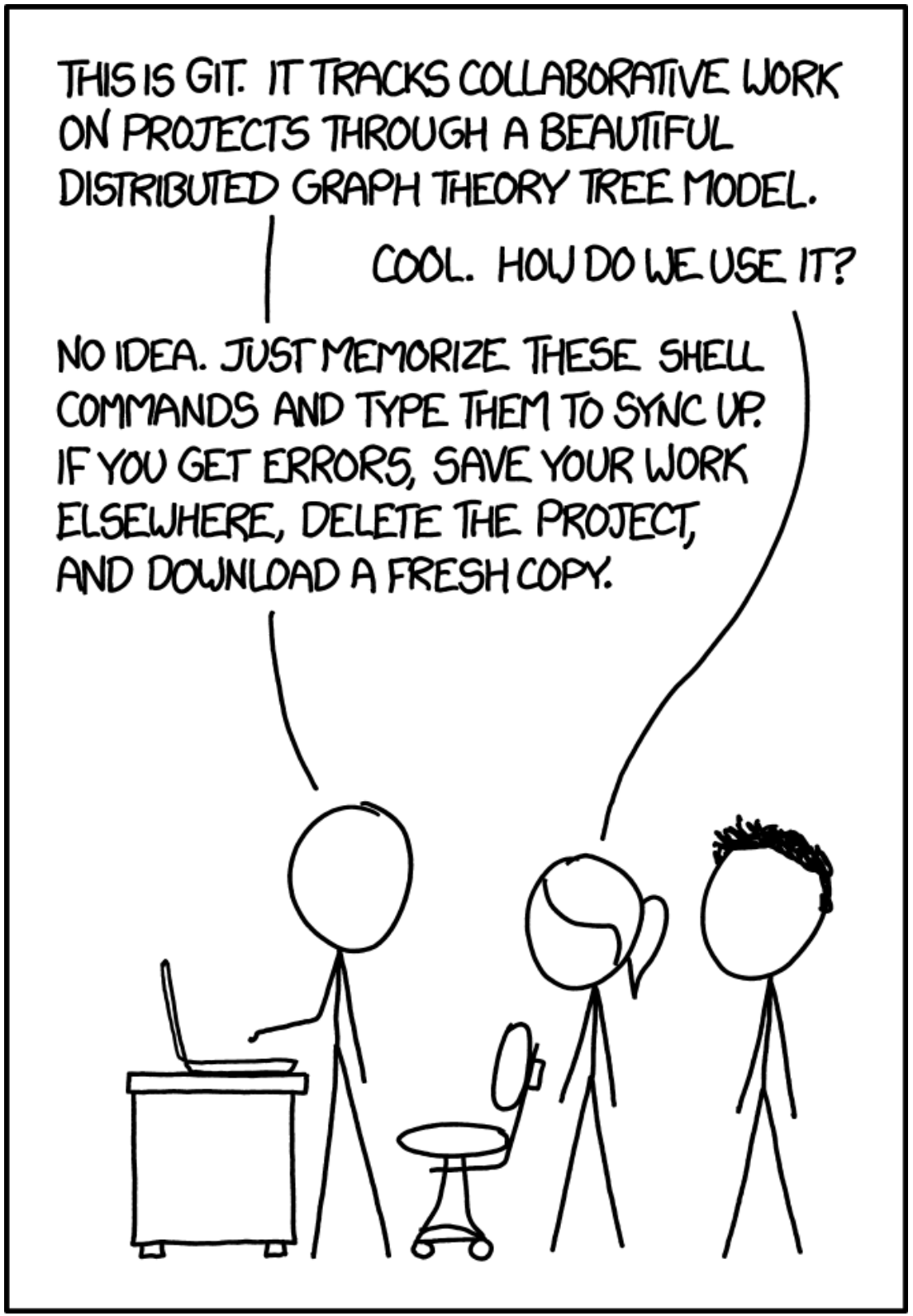
Intro to Git and Github

https://www.atlassian.com/git/tutorials

https://www.youtube.com/playlist?list=PL0lo9MOBetEHhfG9vJzVCTiDYcbhAiEqL


THIS IS GIT. IT TRACKS COLLABORATIVE WORK ON PROJECTS THROUGH A BEAUTIFUL DISTRIBUTED GRAPH THEORY TREE MODEL.
COOL. HOW DO WE USE IT?
NO IDEA. JUST MEMORIZE THESE SHELL COMMANDS AND TYPE THEM TO SYNC UP. IF YOU GET ERRORS, SAVE YOUR WORK ELSEWHERE, DELETE THE PROJECT, AND DOWNLOAD A FRESH COPY.

Install git

- You should already have git installed on your computer
- https://git-scm.com/downloads


```
[03:15:28:~] $ git
usage: git [--version] [--help] [-C <path>] [-c name=value]
           [--exec-path[=<path>]] [--html-path] [--man-path] [--info-path]
           [-p | --paginate | --no-pager] [--no-replace-objects] [--bare]
           [--git-dir=<path>] [--work-tree=<path>] [--namespace=<name>]
           <command> [<args>]

These are common Git commands used in various situations:
```

Create a GitHub account

- You should already have a GitHub account
- Make sure you have the education discount for unlimited free repositories
- Be sure Leslie knows your GitHub name and has invited you to the class repository

What is git?

- Version control software
  - System that keeps records of your changes
  - Allows for collaborative development (when paired with a remote server, like GitHub)
  - Tracks who made which changes when
  - Allows you to revert to a previous state
- Started in 2005 by Linus Torvald to help with Linux development
- Open source, free, widely used

How does git work?
(on a superficial level)

- Creates “snapshots” of your work – commit
  - Records what all files in a repository look like at a given point in time
  - User decides when to create a commit
  - Can move backwards (and forwards) through commits
- A commit contains:
  1. Information about how the files changed from the previous commit
  2. A reference to the previous commit
  3. A hash name

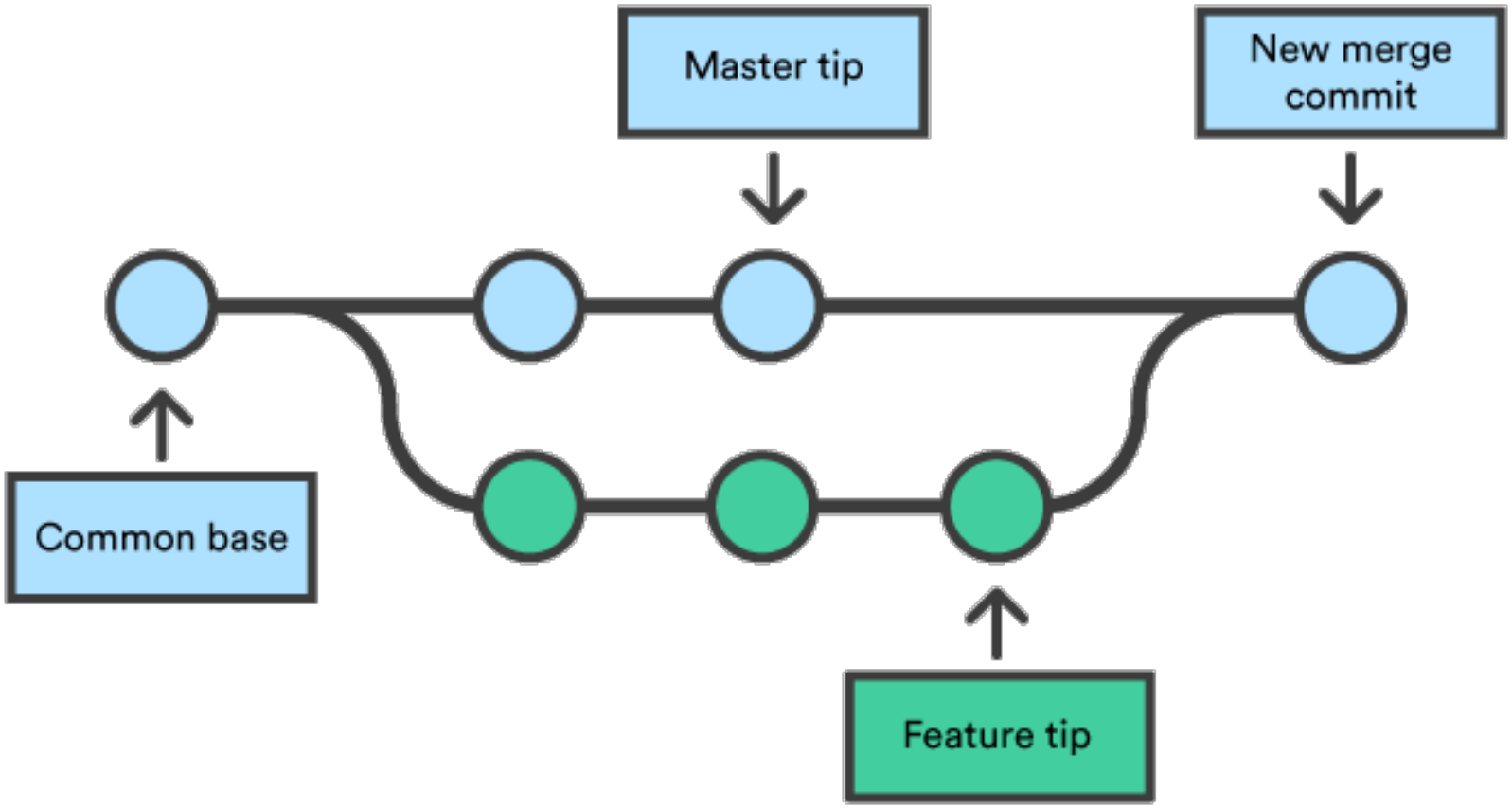
     `7845f1dbb92bcb190d6849e08e05ce49e5071cec`

How does git work?
(on a superficial level)

- A collection of all the files and commits is called a repository (repo)
- Exists on local machine or remote server (GitHub)
- Cloning - copying a repository
  - Allows teams to work together
  - Pulling - downloading commits from a remote repository onto your computer
  - Pushing - adding your local changes to a remote repository

How does git work?

Branching



Let's do it!

```
$ mkdir example    #create a directory for your repo
$ cd example       #move there
$ git init         #create the repo, git can also create the
                   #folder for you with git init example
$ touch readme.md  #create an empty file called readme.md

$ ls -lah          #should see file and .git folder

$ git status       #shows status of get repo, we have
                   #untracked files

$ git add .        #adds all untracked files to staging area
$ git status       #a new file!
$ git commit -m "add file"
$ git status
```

Make a new branch

```
$ git checkout -b working     #make a new branch and change
                              #to it
$ nano readme.md              #make a change to a file
$ git status
$ git add readme.md           #stage changes
$ git commit -m “update readme”
$ git checkout master         #move back to master
$ cat readme.md               #notice it’s empty!
$ git branch                  #see all your branches
$ git merge working           #merge working into master
$ cat readme.md
```

What about GitHub?

- Web-based git repository hosting service
- Allows for collaboration with anyone who also has a GitHub account
- Extra functionality on top of git
  - UI
  - Documentation
  - Bug tracking
  - Feature requests
  - Pull requests
  - …