* Why Github?
  + Version control on steroids
  + Removes the need to rename different version of files
  + Can branch and go back to earlier commits
  + Can directly compare changes made to files
  + Makes collaboration very easy
    - Can clone the entire history
    - Can branch and merge and conflicts will be notified
* <https://git-scm.com/downloads>
  + Change name to “main”
  + For MacOS use Binary installer
  + Use git --version to check for install
* Basic commands
  + Windows
    - pwd – prints working directory
    - cd – changes working directory
    - cd /c/Users/chase/GDrive/GD\_Work/SoDA496/SoDA496\_GitTutorial
    - .. – move one level out
    - ls – list files and folders
    - mkdir – makes working directory
      * mkdir test-folder
    - touch – create new file
      * touch test.txt
    - rm –removes file
      * rm test.txt
* First repo
  + cd /c/GitRepos
  + mkdir soda496
  + cd soda496
  + touch newfile.txt
  + git init
  + git status
  + git add newfile.txt
  + git status
* Config
  + Git config –global user.name ‘Chase Bloch’
  + Git config user.name
  + Git config –global user.email [chasebloch@gmail.com](mailto:chasebloch@gmail.com)
  + Git config user.email
* First commit
  + Git status
  + Add “test” to newfile.txt
  + Git status
  + Git add .
  + Git status
  + Git commit newfile.txt
  + Press esc
  + :wq
  + Git log
  + Make a change to newfile.txt
  + Git add .
  + Git commit –amend newfile.txt
  + Make another change to file
  + git commit -a -m “Added changes”
  + git revert “enter commit id”
  + git reset HEAD~
  + git restore newfile.txt
* Branching and merging
  + Git branch feature
  + Git branch
  + Git checkout feature
  + Git branch
  + Touch feature.r
  + Git add .
  + Git commit -m “Added R feature”
  + Git checkout master
  + Git merge feature
  + Git branch -d feature
* Github Desktop
  + Make a Github account
  + <https://desktop.github.com/>
  + Add soda496 as new repository
  + Show history
  + Push it to Github
  + Show it on github
  + Create a new file directly on github
  + Show that it is not in local repository
  + Fetch it on github desktop
* Merge in github desktop
  + Create feature branch
  + Edit one of the files and commit
  + Go back to master and make a different edit
  + Try to merge in desktop