

Introduction to Data Science Software and Platforms

8/24/20

Data Science Software and Platforms we will Use

Main Purpose in Class	Software/Platform	What is it?
Coding Lab Assignments in Python	Anacondas	Distribution of the Python and R programming languages. Allows you to download and run popular Python packages and the <u>Jupyter Notebook Application</u> .
	Jupyter Notebooks	Python application that allows you to <u>write data science reports</u> that also need to be integrated with interactive Python code blocks.
	Python	A programming language
Version Control: practice of tracking and managing changes to code.	Git	Version control system .
	Github	Git repository hosting service .
	Github Enterprise STAT207 Organization https://github-dev.cs.illinois.edu/stat207-fa20	A collection of user accounts (users=you, your classmates, Dr. Ellison, TAs, and Cas) that owns Github repositories .
	Command Line Interface	An application that processes commands to a computer program in lines of text.

Downloading a Version of Anaconda (Windows)

Installing Python

You will need Python 3.6 (or later). We will first check if you have Python already (if you have done Data Science) and install it if you don't already have it.

Checking for existing Python

1. Open up your [command prompt](#)
 2. Type `python --version` and press **Enter**.
- If you see Python 3.7.1 (or similar), you are all set – no need to install Python. (*Skip to the git section.*)
 - If you see 'python' is not recognized as an internal or external command, operable program or batch file., install it now:

Installing Python

1. Visit <https://conda.io/miniconda.html> to get Miniconda, a light-weight version of the python programming language
2. Download and install the latest Windows, **64-bit** installer for the latest version of Python (eg: 3.7).
3. After the install finishes, exit your command prompt, re-launch it, and verify it installed by following the steps above (in "checking for existing python").

<http://courses.las.illinois.edu/fall2020/stat207/datascience-setup.html>

Downloading a Version of Anaconda (MAC OS X)

Installing Python

You will need Python 3.6 (or later). We will first check if you have Python already (if you have done Data Science) and install it if you don't already have it.

Checking for existing Python

1. Open up your [command prompt](#)
 2. Type `python --version` and press **Enter**.
- If you see Python 3.7.1 (or similar), you are all set – no need to install Python!
 - If you see an error or Python 2.7, we will install it now!

Installing Python

1. Visit <https://conda.io/miniconda.html> to get Miniconda, a light-weight version of the python programming language
2. Download the latest Mac OS X, **64-bit bash** installer for the latest version of Python (eg: 3.7).
3. Open up your [command prompt](#) and run the script you downloaded by running the following:

```
cd Downloads
```

```
bash Miniconda3-latest-MacOSX-x86_64.sh
```

You will need to press q to exit the license screen and all default options are fine.

1. Restart your terminal

<http://courses.las.illinois.edu/fall2020/stat207/datascience-setup.html>

Anaconda

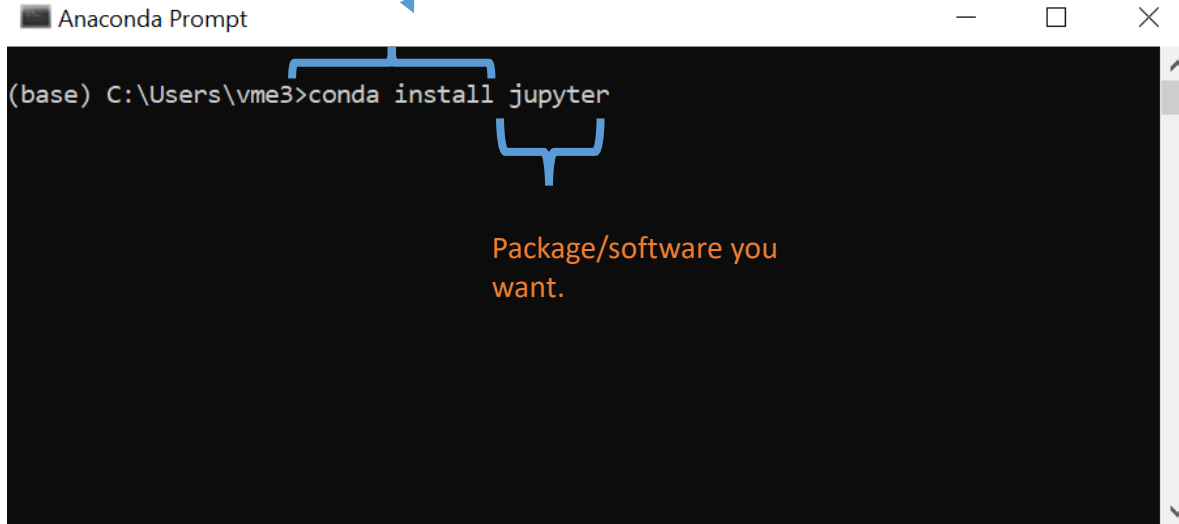
(What applications it comes with)

All versions of Anaconda should come with an **Anaconda Prompt**, a command line interface that allows you to run python commands.

Anaconda Prompt commonly used for:

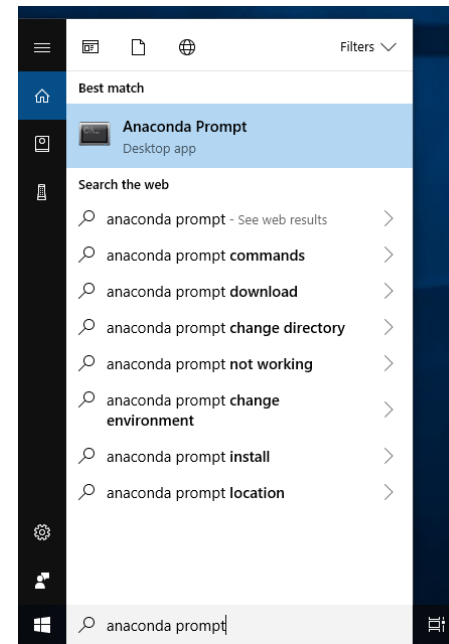
- Installing python packages and applications
- *One way* to launch python applications.

Code for installing *some* software/packages.

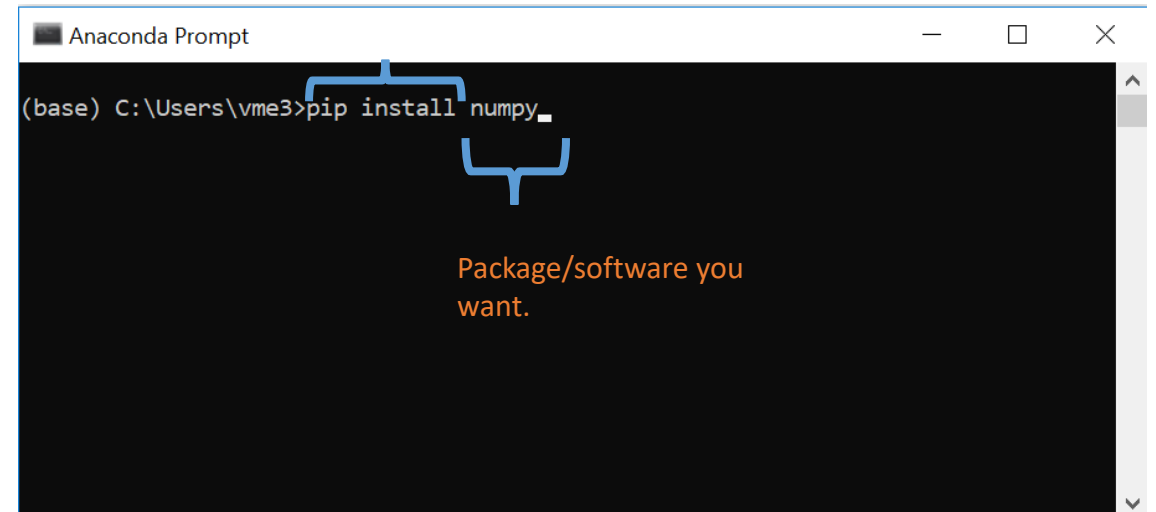


```
(base) C:\Users\hme3>conda install jupyter
```

A blue bracket is drawn under the word 'jupyter' in the command. An orange arrow points from the text 'Code for installing some software/packages.' to the bracket. Below the command, the text 'Package/software you want.' is written in orange.



Code for installing *some* software/packages.



```
(base) C:\Users\hme3>pip install numpy
```

A blue bracket is drawn under the word 'numpy' in the command. An orange arrow points from the text 'Code for installing some software/packages.' to the bracket. Below the command, the text 'Package/software you want.' is written in orange.

Downloading Jupyter Notebooks and Other Python Packages/Applications

Part 1c: Set up your Python notebook

In Data Science, all of our programming will be done in “notebooks”. Your python install will need a few **libraries** in order to run the notebooks. Using your command line, run the following:

```
conda install jupyter
conda install pandas
conda install matplotlib
conda install seaborn
```

Potential Error Workaround: IF you get an error about "conda not found" when trying to do this, you can also install these packages by doing the following.

- Searching for the "miniconda" program you just downloaded, and run what should say "Anaconda Prompt."
- This will open up another command line window that is specifically for running python commands (for instance commands that install packages).
- Run the code in this Anaconda Prompt instead

```
conda install jupyter
conda install pandas
conda install matplotlib
conda install seaborn
```

This might take a couple of minutes. You will need to type [y] to confirm you want to install of of the packages (the option [y]/n shows that y is default when you choose no option).

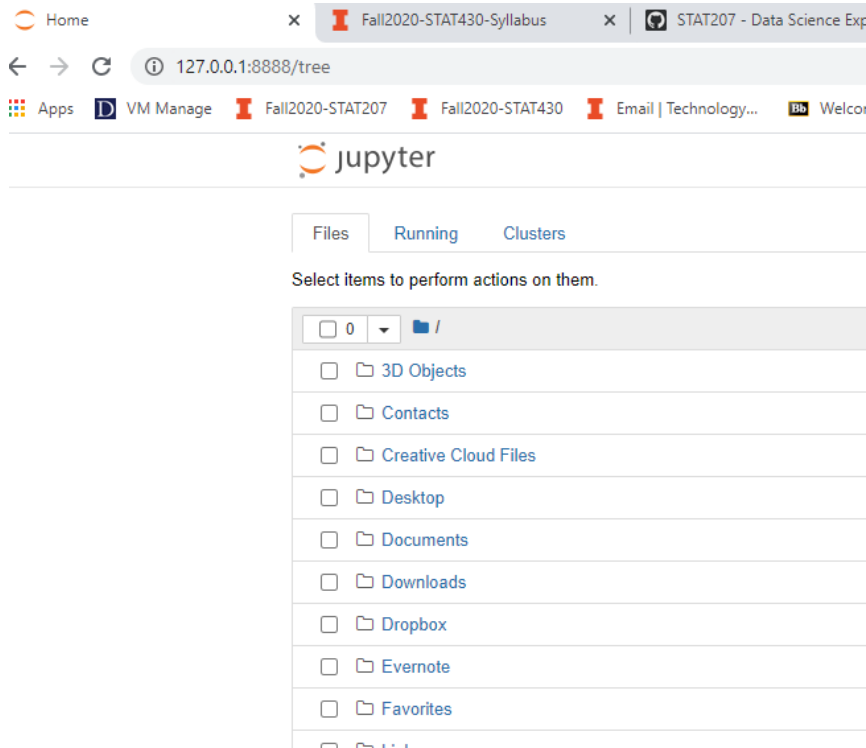
You can check what has been installed already using the command:

```
conda list
```

<http://courses.las.illinois.edu/fall2020/stat207/labs/01-intro.html>

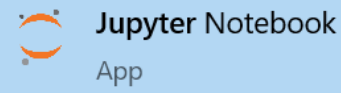
Running Jupyter Notebooks

Displays a File System of your Computer in a Browser



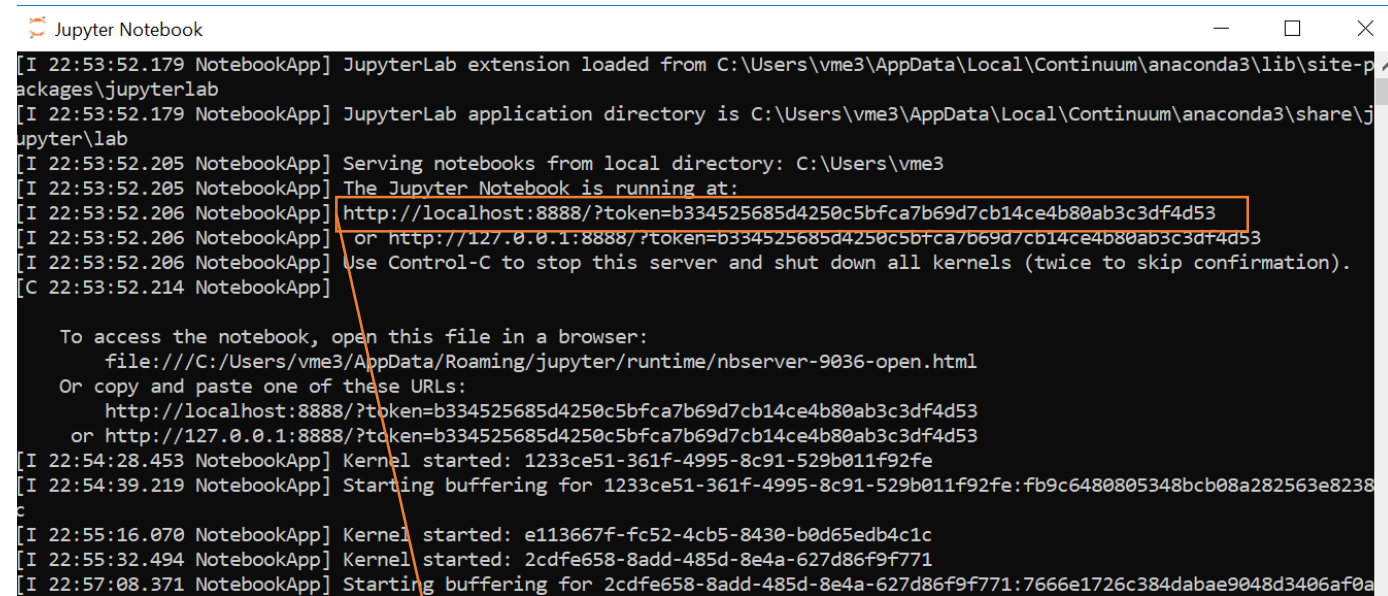
...navigate to where your notebooks are saved!

Best match



Search the web

Launches the Notebook Application with Information about your Jupyter Notebook Session



Url to relaunch the Jupyter Notebook your browser.

Running Jupyter Notebooks

Best match



Jupyter Notebook

App

Search the web



jupyter - See web results



Open your notebook

 jupyter

Quit

Logout

Files

Running

Clusters

Select items to perform actions on them.

Upload

New

☐

0

/ Documents / Teaching / Fall2020 / stat207 / github / instructor / _release / lab_01

Name

Last Modified

File size

..

seconds ago

☐

 lab_01.ipynb

4 hours ago

4.67 kB

☐

 spam_sample.csv

2 months ago

4.75 kB

Running Jupyter Notebooks

The screenshot shows a Jupyter Notebook titled "lab_01 - Jupyter Notebook" in a web browser. The browser's address bar shows the URL "127.0.0.1:8888/notebooks/Documents/Teaching/Fall2020/stat207/github/instructor/_release/lab_01/lab_01.ipynb". The notebook's menu bar includes File, Edit, View, Insert, Cell, Kernel, Widgets, and Help. The "File" menu is open, showing options like "New Notebook", "Open...", "Make a Copy...", "Save as...", "Rename...", "Save and Checkpoint...", "Revert to Checkpoint...", "Print Preview", "Download as", "Trusted Notebook", and "Close and Halt". A blue arrow points from the "Save and Checkpoint..." option to the text "Don't forget to save your work before exiting!". Another blue arrow points from the "Run" button in the top toolbar to the text "[ctrl][s]". The notebook content includes a title "STAT 207 Lab 1: Python with Jupyter Notebooks", a timestamp "Wednesday, September 1, 23:59:59 CST", and a hint "THIS TO PUT YOUR NAME AND NET ID HERE (Hint: this is a Markdown cell - click on it to edit)". The "Instructions:" section states: "Make sure you have the following packages installed for Python on your computer: pandas, matplotlib, pyplot, seaborn. To install them, launch a command prompt window and issue commands of the form: conda install *package_name_here*. To check if they are installed issue the command: conda list and look through the list to make sure the packages you need are there. When done, return to this notebook." The "Part 2." section says: "After ensuring that the packages listed above are installed on your computer import them here in the notebook by running the following commands. You should be able to do this by clicking anywhere inside the cell to select it and then hitting the 'Run' button in the menu at the top of this page (or press [ctrl][Enter] on your keyboard)." A code cell is shown with the following code:

```
In [2]: import pandas as pd          # imports pandas and calls the imported version 'pd'
import matplotlib.pyplot as plt    # imports the package and calls it 'plt'
import seaborn as sns              # imports the seaborn package with the imported name 'sns'
sns.set()
```

 The "Part 3." section starts with: "a) You should have a file 'spam_sample.csv' in the same folder as this notebook. Read the file into a pandas data frame. You may call it 'df' or whatever you like. As illustrated in the class notes, use the .head() function to display the first several lines of the data frame." A code cell is shown with the prompt "In []: " and a cursor. The text "b) Use .shape to determine how many observations (rows) there are." is partially visible at the bottom.

Don't forget to **save** your work before exiting!

[ctrl][s]

To **run** code, click the code block and then either do:

- [ctrl][Enter] or
- Click run.

Installing Git (Windows)

Installing Git

Any modern version of git works. We will first check if you have git and install it if you don't already have it.

Checking for git

1. Open up your [command prompt](#)
2. Type `git --version` and press **Enter**.
 - If you see `git version ...` (or similar), you are all set – no need to install git! (*You're done!*)
 - If you see 'git' is not recognized as an internal or external command, operable program or batch file., install it now:

Installing git

1. Visit <https://git-scm.com/downloads> to get git, a distributed version control system/repository tool
2. Download and install the latest Windows installer. (You should not need to select/unselect any of the options that are already preselected in the installation proces... aka just keep hitting next.)
3. After the install finishes, exit your command prompt, re-launch it, and verify it installed by following the steps above.

<http://courses.las.illinois.edu/fall2020/stat207/datascience-setup.html>

Installing Git (Mac OS X)

Installing Git

Any modern version of git works. We will first check if you have git and install it if you don't already have it.

Checking for git

1. Open up your [command prompt](#)
 2. Type `git --version` and press **Enter**.
- If you see git version ... (or similar), you are all set – no need to install git!
 - If you see an error, we will install it now!

Installing git

1. Visit <https://git-scm.com/downloads> to get git, a distributed version control system/repository tool
2. Download and install the latest Mac OS X installer.
3. After the install finishes, verify it installed by following the steps above.

<http://courses.las.illinois.edu/fall2020/stat207/datascience-setup.html>

Setting up a Git Repository in our STAT207 Github Enterprise Organization <https://github-dev.cs.illinois.edu/stat207-fa20>

Part 0: General Class Folder

First, you should create a folder named 'stat207' (we recommend on your Desktop) to hold all of your Python notebooks.

Guide: Setting Up git for Data Science Exploration

To set up git, there are certain commands you will run:

1. Once for the entire semester ("Course Setup"),
2. Once for each computer you use ("Computer Setup"),
3. Once each time you work on Data Science Discovery ("Assignment Setup")

Course Setup

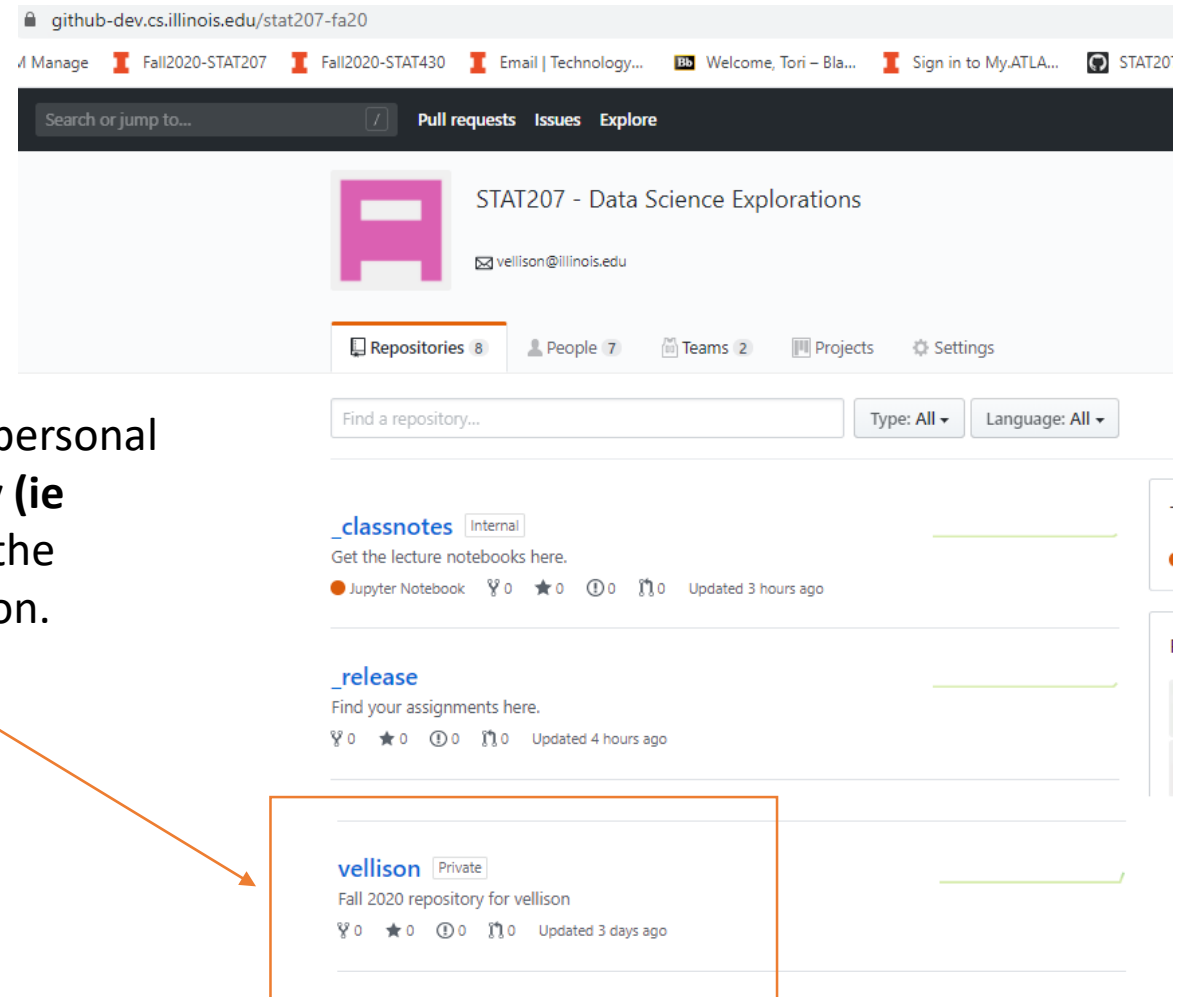
To begin to work on assignments and turn in work, you will need to create a git repository for the course.

- Visit: <https://edu.cs.illinois.edu/create-ghe-repo/stat207-fa20/>
- Follow the instructions to create your repository, coming back here once you have the URL

Computer Setup

1. Create a stat207 folder on your Desktop (if you haven't already)
2. Using your command line, run the following commands to navigate into your stat207 folder:
 - `cd Desktop`
 - `cd stat207`
3. **Clone** a local copy of your git repository with the following command (making sure to replace YOUR-NETID with your own):
 - `git clone https://github-dev.cs.illinois.edu/stat207-fa20/YOUR_NETID`
 - You may have to enter your netid/password
4. Navigate into your git directory by going into your NetID-named folder (replace NETID with yours):
 - `cd NETID`
5. Set up a connection to the _release repository where code will be released for you:
 - `git remote add release https://github-dev.cs.illinois.edu/stat207-fa20/_release.git`

Creates a personal repository (ie folder) in the organization.



Setting up a Git Repository in our STAT207 Github Enterprise Organization <https://github-dev.cs.illinois.edu/stat207-fa20>

Part 0: General Class Folder

First, you should create a folder named 'stat207' (we recommend on your Desktop) to hold all of your Python notebooks.

Guide: Setting Up git for Data Science Exploration

To set up git, there are certain commands you will run:

1. Once for the entire semester ("Course Setup"),
2. Once for each computer you use ("Computer Setup"),
3. Once each time you work on Data Science Discovery ("Assignment Setup")

Course Setup

To begin to work on assignments and turn in work, you will need to create a git repository for the course.

- Visit: <https://edu.cs.illinois.edu/create-ghe-repo/stat207-fa20/>
- Follow the instructions to create your repository, coming back here once you have the URL

Computer Setup

1. Create a stat207 folder on your Desktop (if you haven't already)
2. Using your command line, run the following commands to navigate into your stat207 folder:

- `cd Desktop`
- `cd stat207`

3. **Clone** a local copy of your git repository with the following command (making sure to replace YOUR-NETID with your own):

- `git clone https://github-dev.cs.illinois.edu/stat207-fa20/YOUR_NETID`
- You may have to enter your netid/password

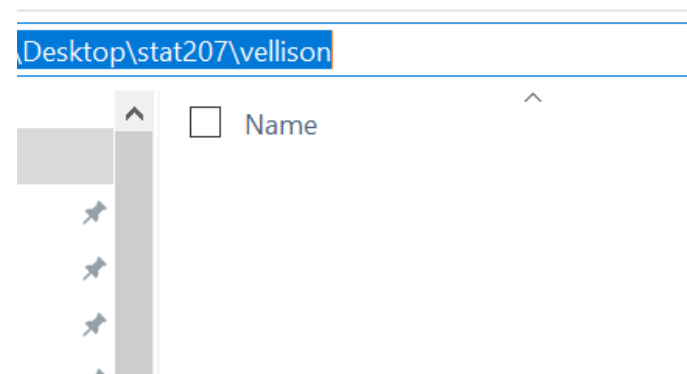
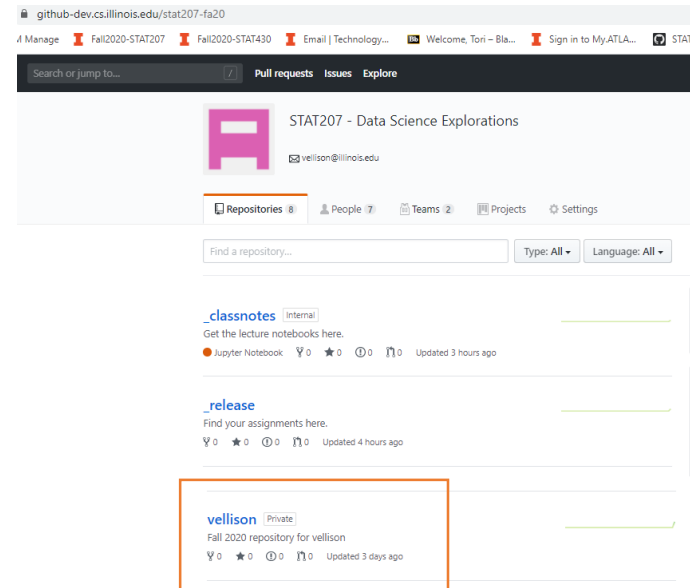
4. Navigate into your git directory by going into your NetID-named folder (replace NETID with yours):

- `cd NETID`

5. Set up a connection to the _release repository where code will be released for you:

- `git remote add release https://github-dev.cs.illinois.edu/stat207-fa20/_release.git`

Only done once: Clones your personal repository to your local computer.

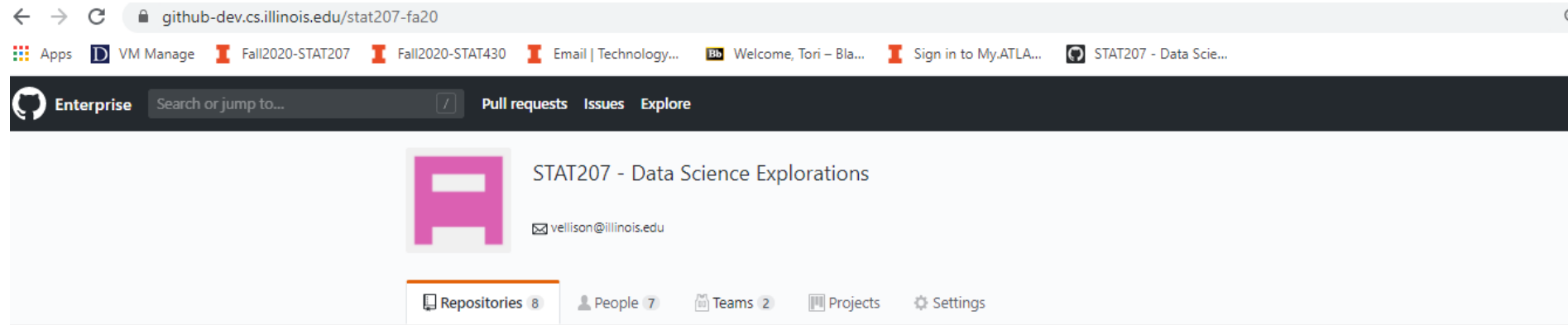


Our STAT207 Github Enterprise Organization

Our organization also contains other repositories *that will be updated* as the semester progresses.

The screenshot shows the Github Enterprise interface for the organization 'STAT207 - Data Science Explorations'. The browser address bar shows 'github-dev.cs.illinois.edu/stat207-fa20'. The organization's profile includes a pink 'H' logo, the name 'STAT207 - Data Science Explorations', and the email 'vellison@illinois.edu'. Below the profile, there are tabs for 'Repositories' (8), 'People' (7), 'Teams' (2), 'Projects', and 'Settings'. A search bar 'Find a repository...' is present, along with filters for 'Type: All' and 'Language: All', and a 'New' button. The repository list shows three items: 1) '_classnotes' (Internal) with a description 'Get the lecture notebooks here.', 'Jupyter Notebook' language, 0 forks, 0 stars, 0 issues, 0 pull requests, and 'Updated 3 hours ago'. 2) '_release' with a description 'Find your assignments here.', 0 forks, 0 stars, 0 issues, 0 pull requests, and 'Updated 4 hours ago'. 3) 'vellison' (Private) with a description 'Fall 2020 repository for vellison', 0 forks, 0 stars, 0 issues, 0 pull requests, and 'Updated 3 days ago'. On the right side, there are sections for 'Top languages' showing 'Jupyter Notebook' and 'People' showing 7 members with their avatars.

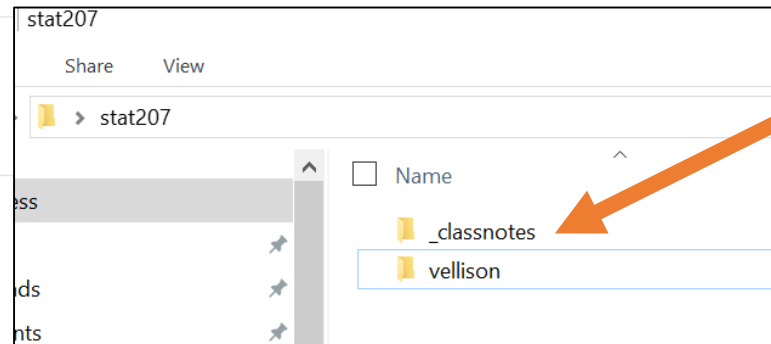
Our STAT207 Github Enterprise Organization



<http://courses.las.illinois.edu/fall2020/stat207/classnotes.html>

pull

You will **pull** the class **lecture notes** from this repository to your **local computer**.



Our STAT207 Github Enterprise Organization



github-dev.cs.illinois.edu/stat207-fa20

Apps VM Manage Fall2020-STAT207 Fall2020-STAT430 Email | Technology... Welcome, Tori - Bla... Sign in to My.ATLA... STAT207 - Data Scie...

Enterprise Search or jump to... Pull requests Issues Explore

STAT207 - Data Science Explorations
vellison@illinois.edu

Repositories 8 People 7 Teams 2 Projects Settings

Find a repository... Type: All Language: All Customize pins New

_classnotes Internal
Get the lecture notebooks here.
Jupyter Notebook 0 stars 0 forks Updated 3 hours ago

_release
Find your assignments here.
0 stars 0 forks Updated 4 hours ago

vellison Private
Fall 2020 repository for vellison
0 stars 0 forks Updated 3 days ago

stat207

Share View

stat207

Name

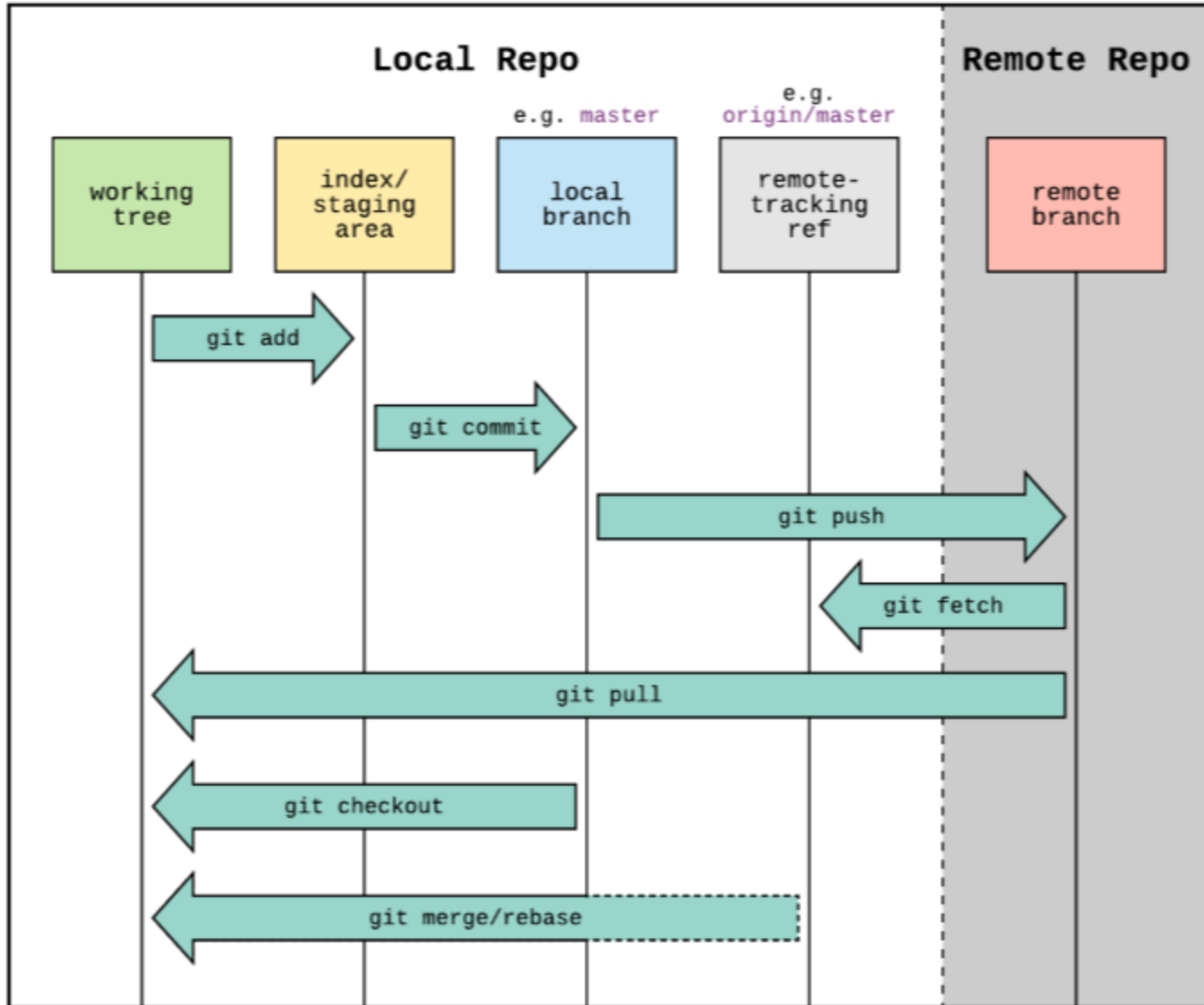
_classnotes

vellison

You will **fetch** and **merge** the class **lab assignments** from this repository to your **LOCAL netid** repository.

<http://courses.las.illinois.edu/fall2020/stat207/labs/01-intro.html>

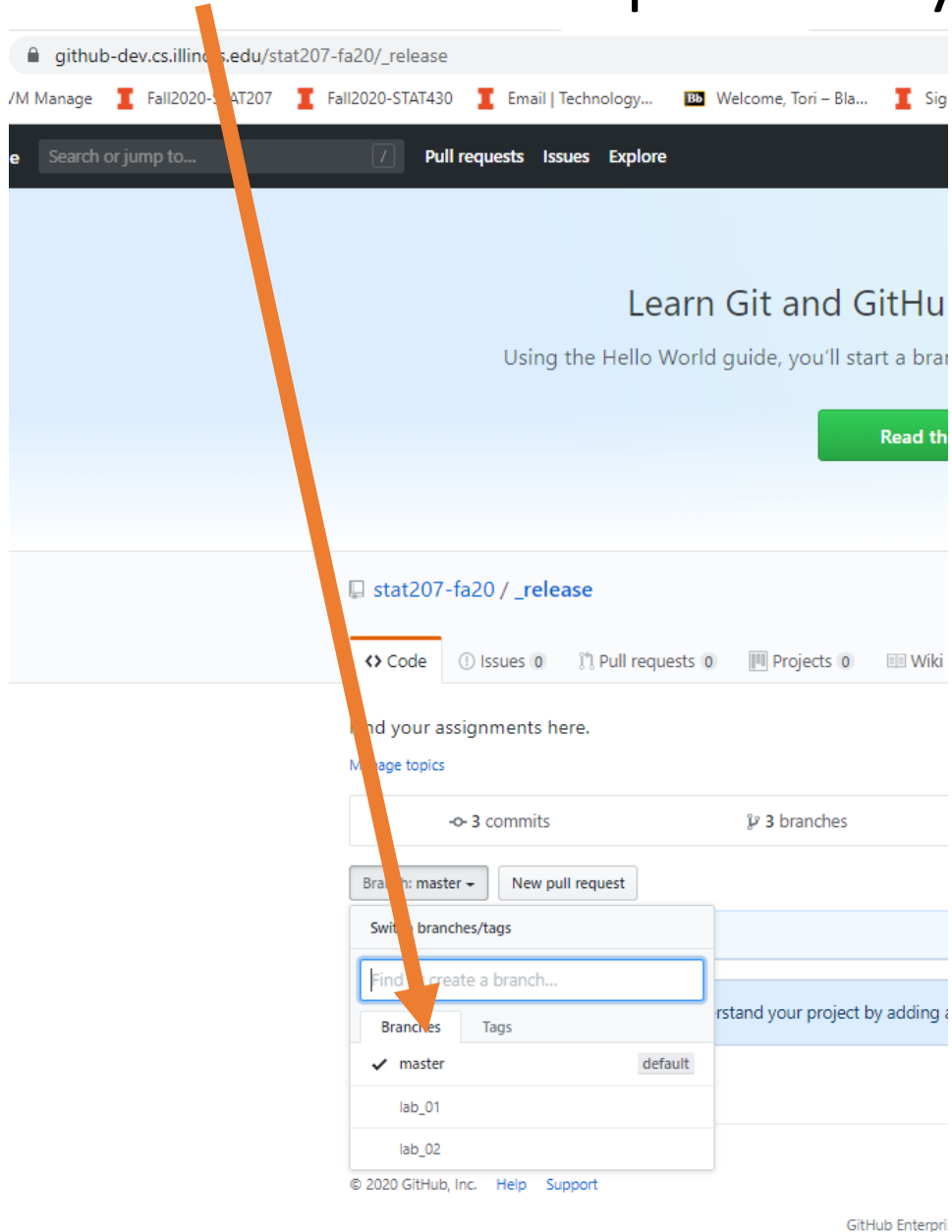
Git Workflow, what do these git commands mean?



Commands:

- **git fetch release**
 - Transmits the “whole version” of a REMOTE repository to your LOCAL computer.
- **git merge release/lab_01 -m “merging initial files”**
 - Merges just the changes made to the lab_01 **branch** of the remote release repository.

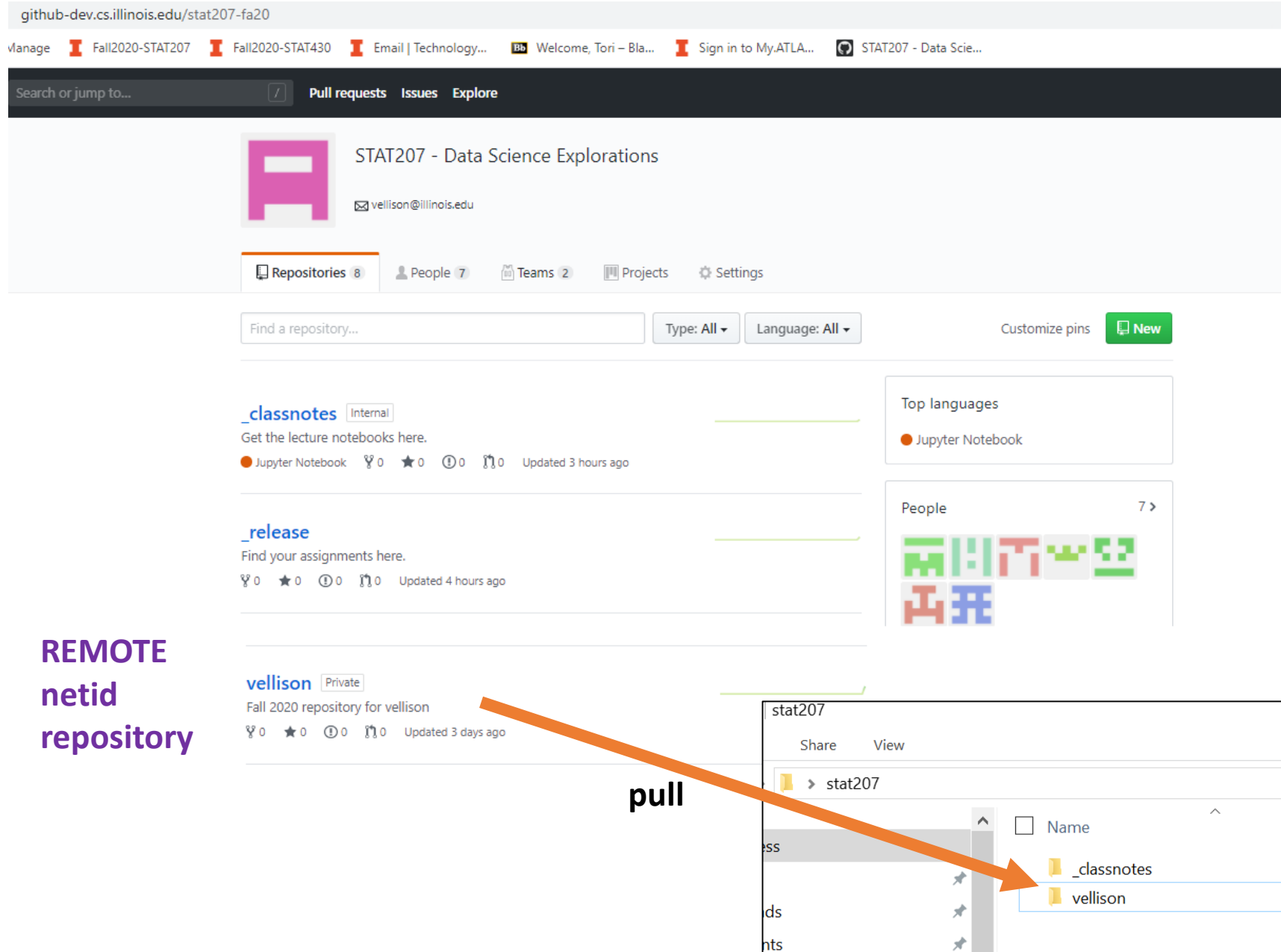
Branches of a Repository



- Repositories can have multiple **branches** (or versions).

Our STAT207 Github Enterprise Organization

<http://courses.las.illinois.edu/fall2020/stat207/labs/01-intro.html>



REMOTE
netid
repository

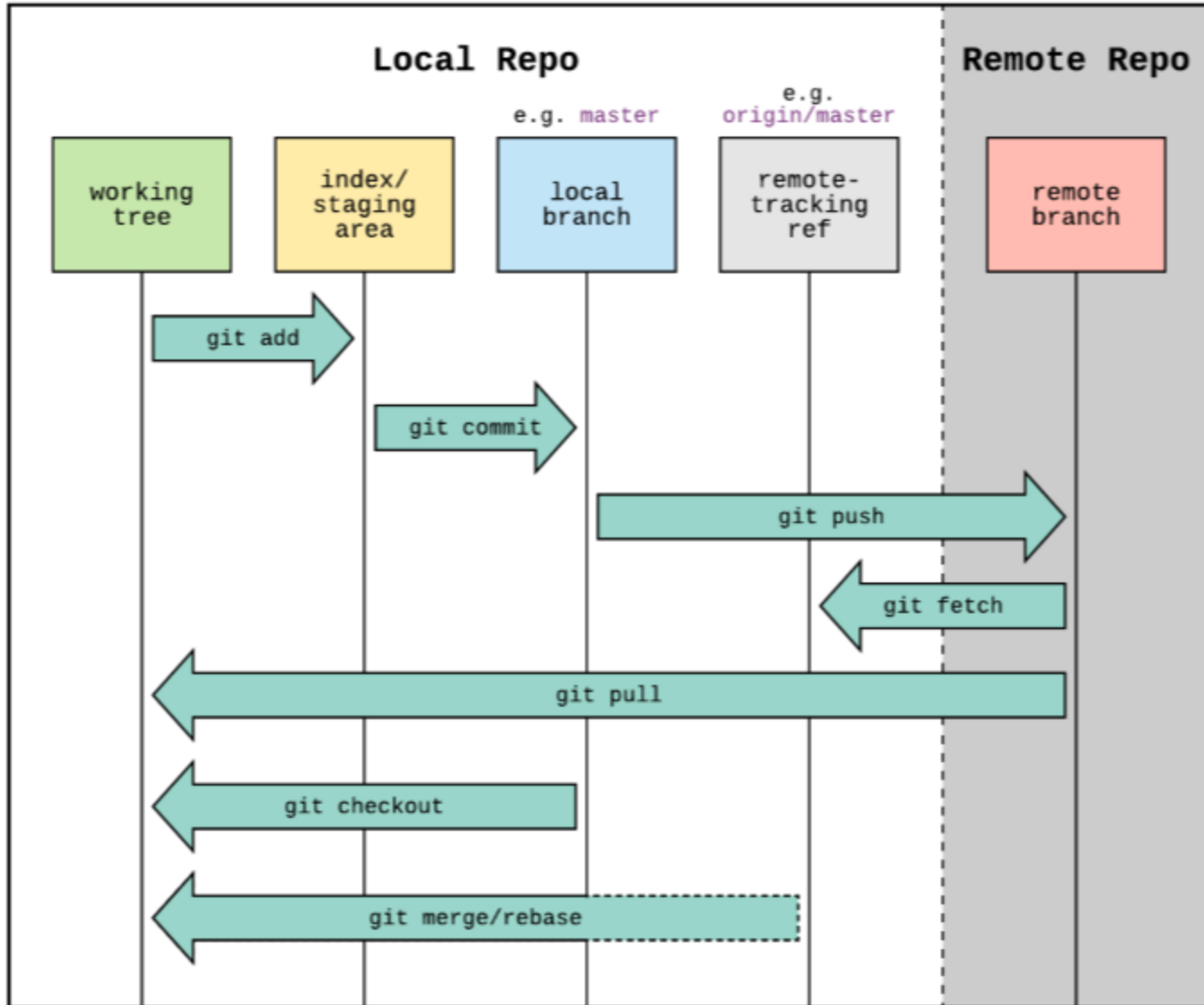
pull

After editing the contents of your **local current version of your netid repository** (like your lab assignments) you will do the following to submit them for saving/grading.

- **pull** any changes from your **REMOTE netid repository** to your **LOCAL netid repository**

LOCAL
netid
repository

Git Workflow, what do these git commands mean?

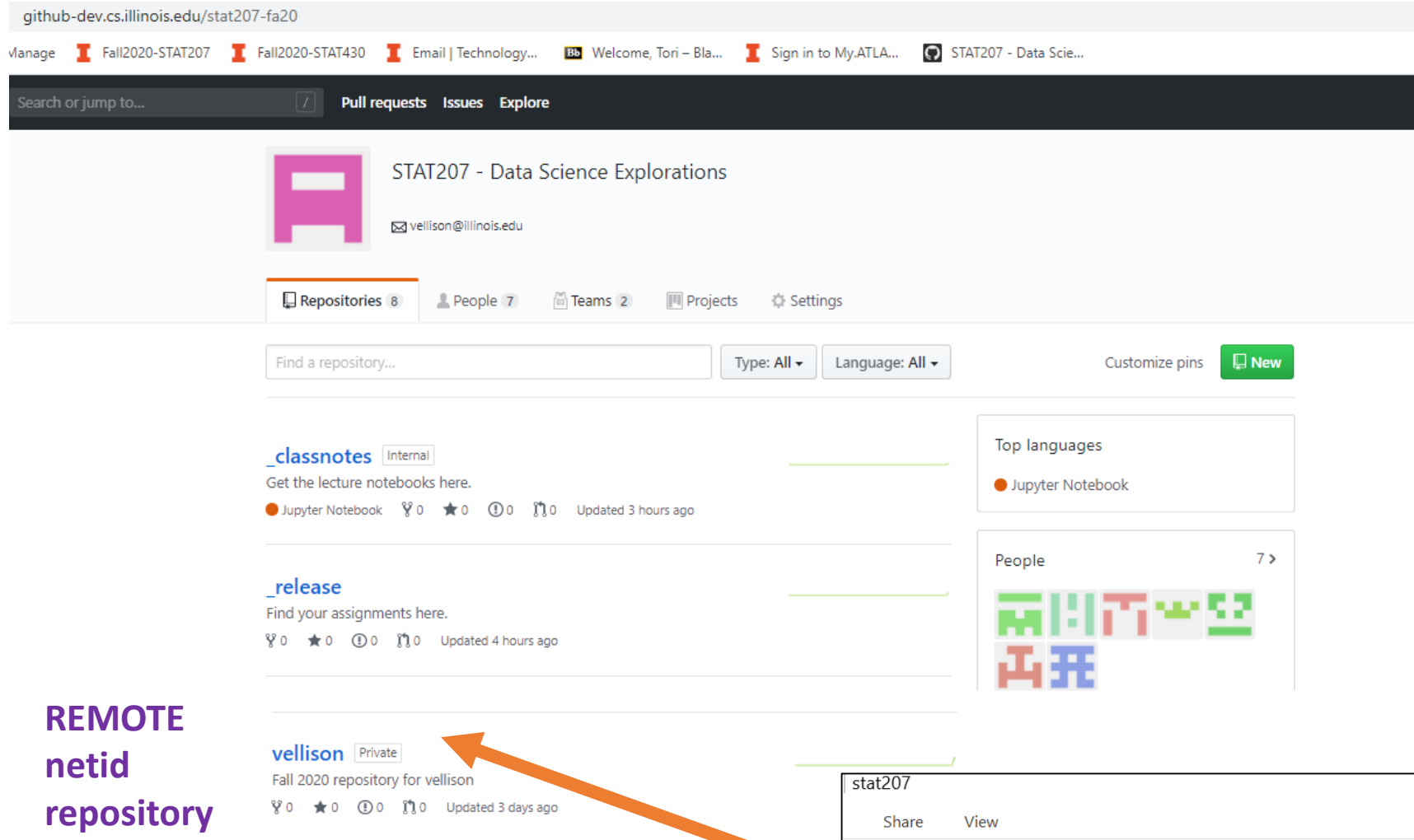


Commands:

- **git pull**
 - Pulls in (and automatically merges) any changes from the REMOTE repository into the LOCAL repository.

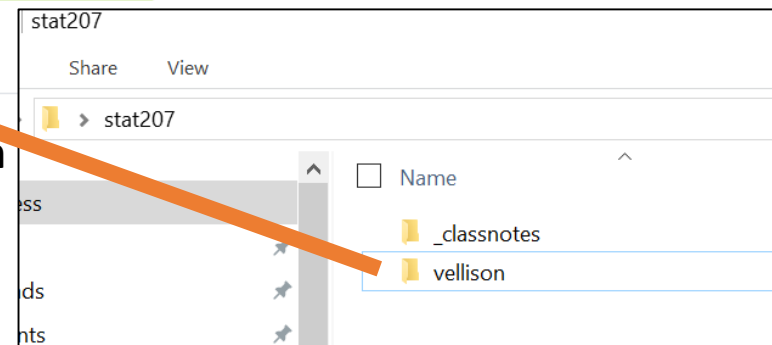
Our STAT207 Github Enterprise Organization

<http://courses.las.illinois.edu/fall2020/stat207/labs/01-intro.html>



REMOTE
netid
repository

add, commit, and push

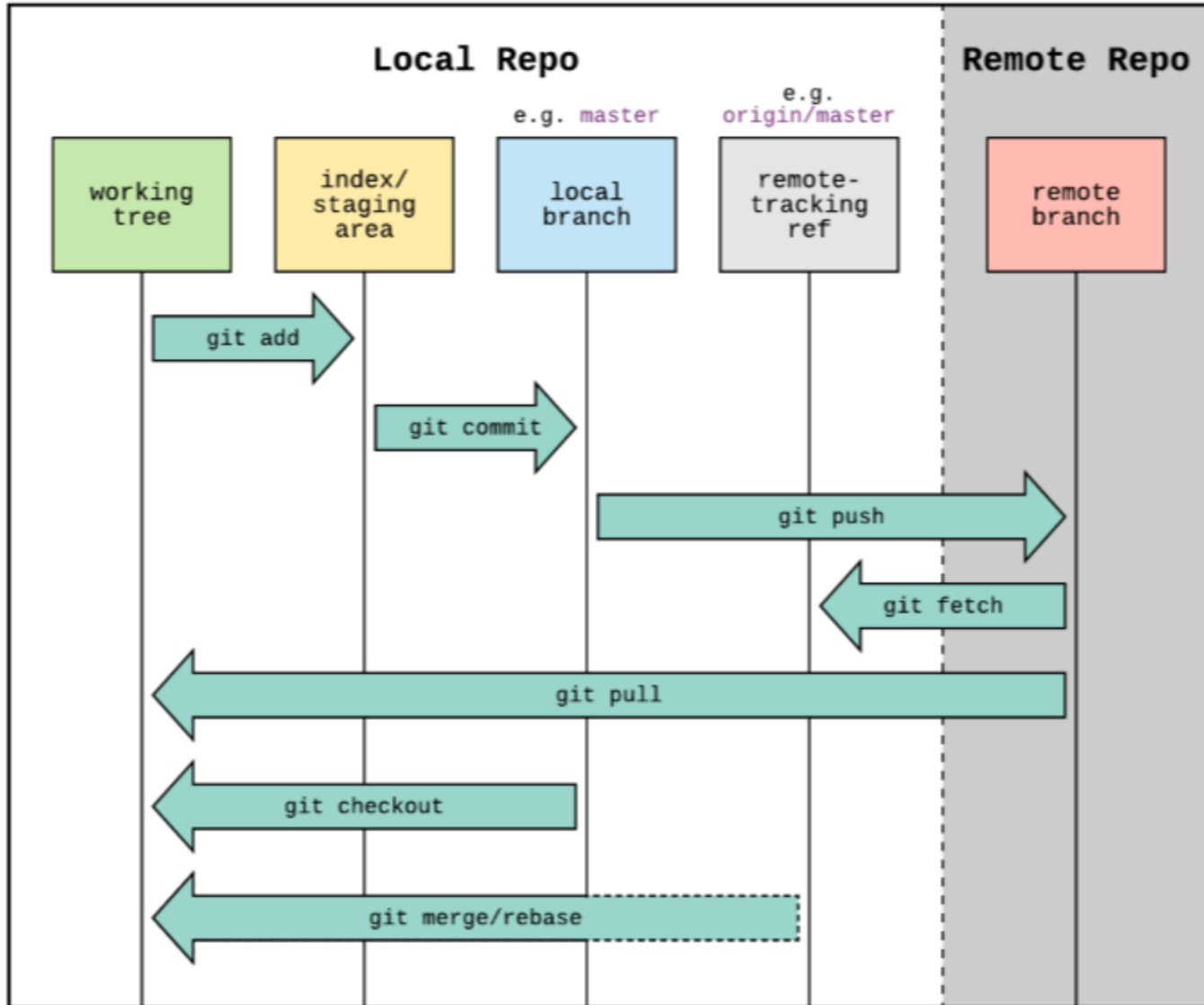


LOCAL
netid
repository

After editing the contents of your **local current version of your netid repository** (like your lab assignments) you will do the following to submit them for saving/grading.

- **pull** any changes from your **REMOTE netid repository** to your **LOCAL netid repository**
- And then **add, commit, and push** the changes from your **LOCAL netid repository** to your **REMOTE netid repository** for grading/saving.

Git Workflow, what do these git commands mean?

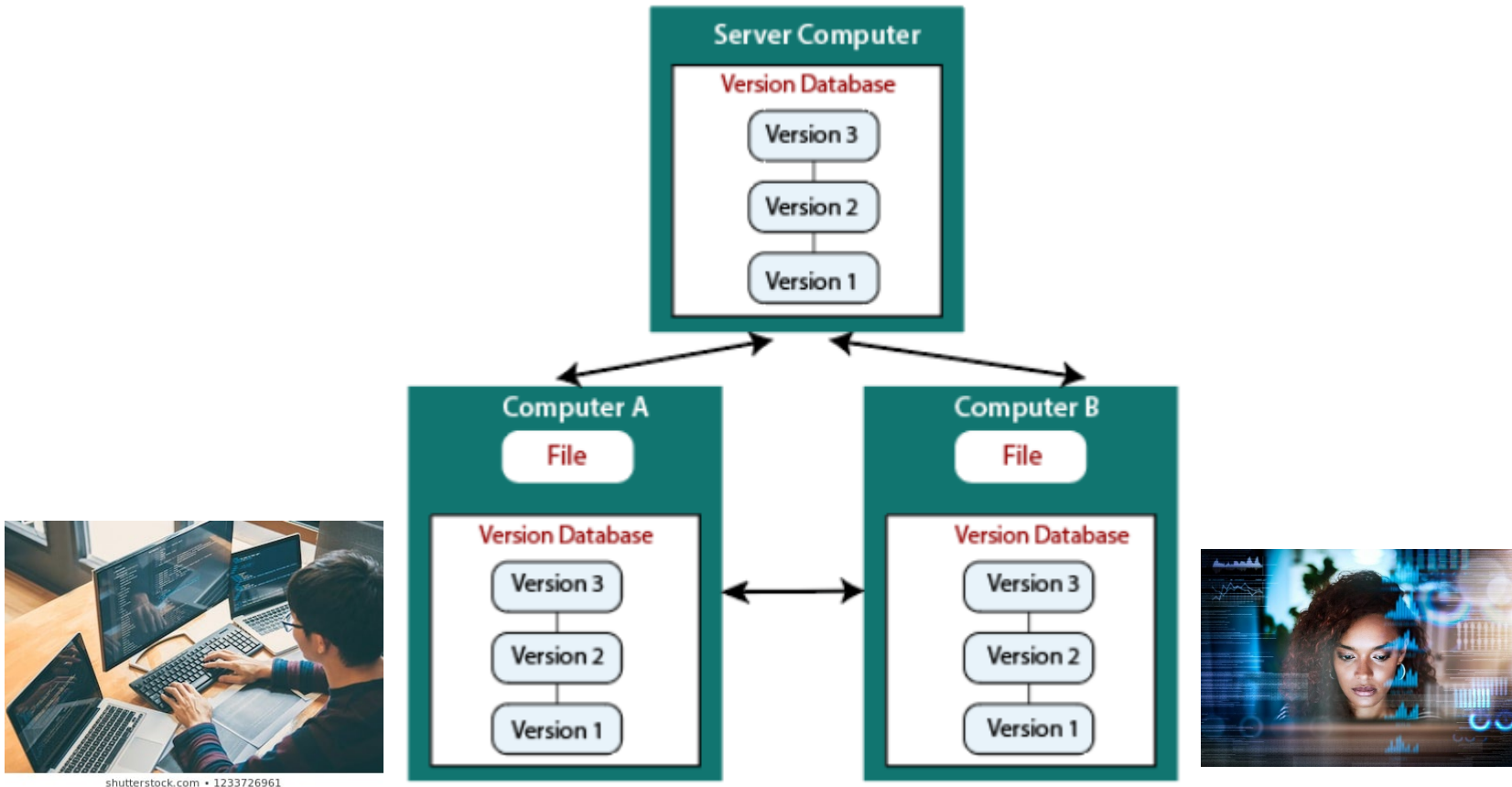


Commands:

- **git add -A**
 - Adds your changes to the **index/staging area** (this is a “rough draft space.”)
- **git commit -m “message explaining your changes”**
 - Adds a “snapshot of your project” (with changes) to the local branch along with your message.
- **git push origin master**
 - Pushes/overwrites these changes you made back to the REMOTE repository.

What is Git?

- A Distributed Version Control System



<https://www.javatpoint.com/git-version-control-system>