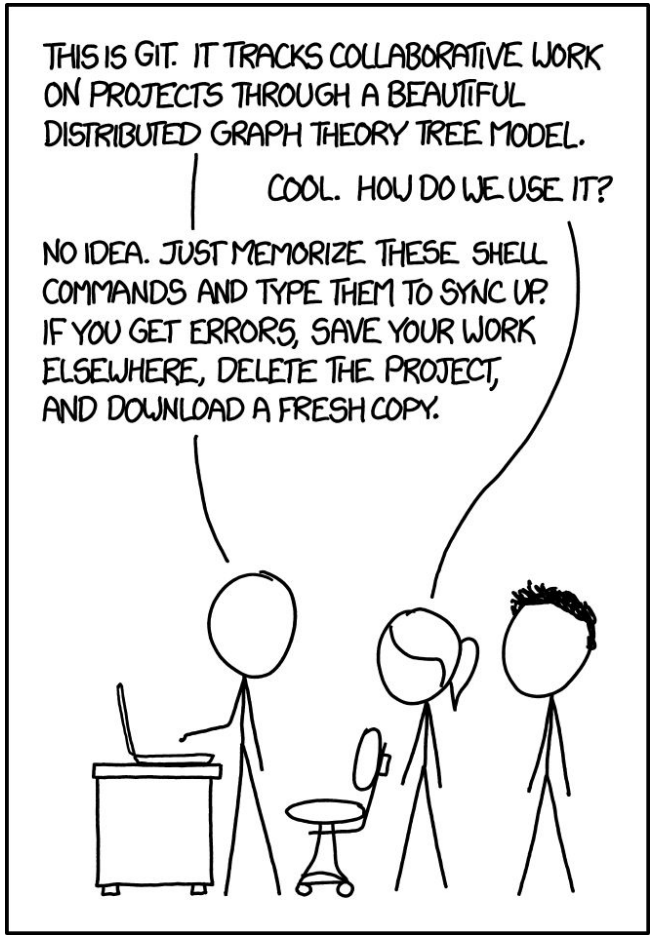


Image Source

Welcome to Week 13 Lecture 1!

Intro to Local Workflow: Terminal/GitBash, GitHub Desktop, and Jupyter Notebooks

Agenda

- Progress Check & Assignments
- **Today's Topics:**
  - GitHub / GitHub Repositories
  - Using GitHub Desktop
  - Using Jupyter Notebook
  - Walkthrough: Creating an Assignment Repo + Notebook
  - Querying MySQL with Python

- **Thursday's Topics:**
  - Using MySQL Workbench - Generating an ERD
  - Advanced SQL Queries
  - Database Administration
  - Exporting MySQL Databases

Assignments

- Core Assignments for this Week:
  - Queries: Sakila (Core)
  - Project 3 - Part 1 (Core)
  - Books (Core)

- Assignment Deadline - This Week:
  - Deadline is Friday at 9 AM PST.
  - If you have technical issues, email me (jirving@codingdojo.com) by 9 AM PST with:
    - Which assignment you're having trouble with
    - A description of your issues/errors.
    - If its an error, include screenshots!

Progress Check

- Local Python Installation.
  - Have you been able to install Python locally and your Setup your dojo-env successfully?
    - A) Yes!
    - B) No, I ran into problems.
    - C) No, I haven’t tried yet.

- For those that ran into problems installing Python + dojo-env, where did you get stuck?

Your New Tools

New Tools for Data Enrichment - Part 1

- **Your Terminal/GitBash:**
  - Used to start jupyter notebook
  - Used for navigating through your files, create new folders, etc

- **GitHub Desktop:**
  - Used to create new repositories and to sync your work.
  - Convenience options for launching terminal/GitBash, Showing a Repo in FInder/File Explorer, Viewing a Repo on GitHub, or opening a repo in VS Code.

- **Jupyter Notebook:**
  - A terminal-based program that launches a special notebook interface in your web browser.
  - Accesses files stored on your local hard drive.

New Tools for Data Enrichment - Part 2

- **VS Code (next week):**
  - A special text editor, but for code.
  - You will use to create/edit special files on your machine.
  - You won't need it much this week.

- **MySQL Server (next class):**
  - Used to host and connect to your own MySQL databases on your local machine

- **MySQL Workbench (next class):**
  - Used to provide a graphical user interface (GUI) for our database administration tasks.
  - Designing Databases/ERD
  - Importing/Exporting databases to/from our MySQL server.

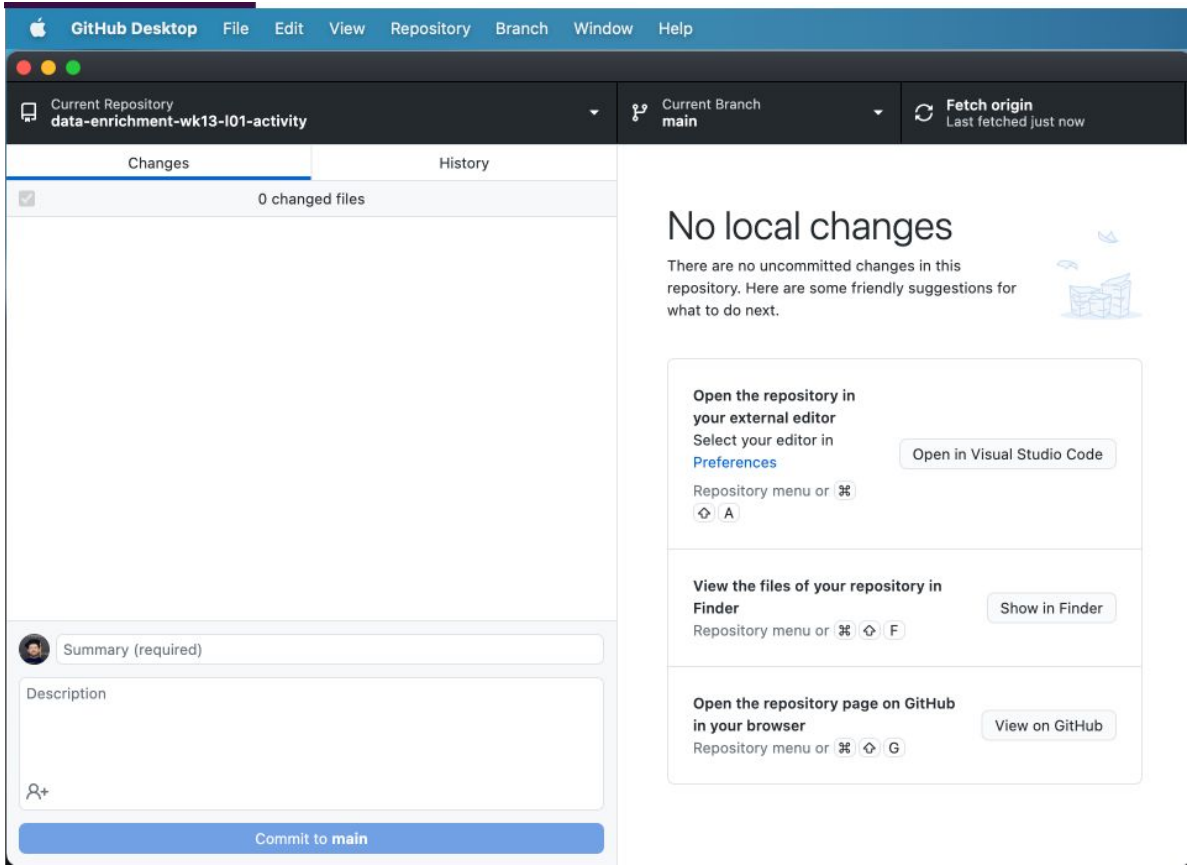
GitHub & GitHub Desktop

GitHub Repositories & Version Control

- A GitHub repository is a **version-controlled folder** containing code and related files.
  - By “version-controlled”, we mean that GitHub **tracks every change made to all files within a repository.**
    - **The changes are saved as a snapshot of each file at that point in time, called a** “commit”.
    - This allows us to retrieve any previous snapshot of our work, if we need to.

- We will use GitHub Desktop to **create new repositories or to clone repositories that already exist.**

GitHub Desktop Interface



To Open Repo in Jupyter:

1. Click on the “Repository” menu on the menu bar/top of the window.
2. Select “Open in Terminal”/”Open in GitBash”
3. In the terminal/GitBash window, start jupyter notebook by running: “jupyter notebook” or “jnb” (if you added the alias as shown in step 3. Setting dojo-env as your default)

Your Terminal/GitBash

Important Commands

- **"pwd"** (print working directory): show the current folder name
- **"ls"** show the files in the current folder

- **"cd"** change directory.
- **"cd ..**" navigate up one folder.
- "cd ~" navigate back to your user folder
  - OR **"cd /Users/your-username/"**
- **"mkdir folder-name"** make a new folder named folder-name.

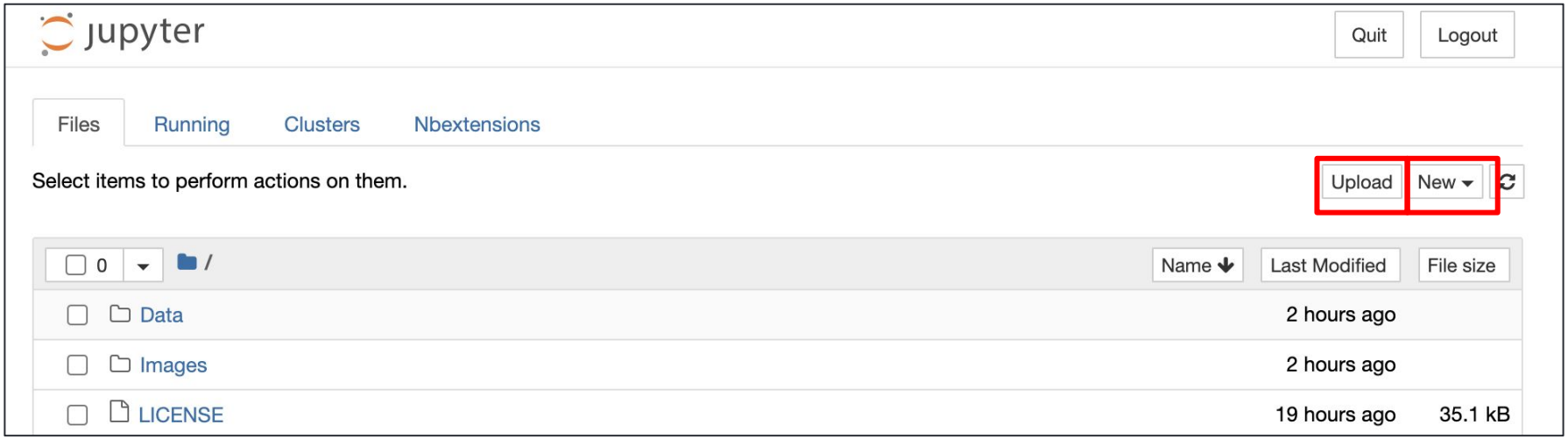
- **"jupyter notebook"**: launches jupyter from the current folder.
  - Or **"jnb"** if you followed all of the setup instructions.

- **"conda activate dojo-env"**: activates your dojo-environment.
  - Should happen automatically if you followed all of the instructions.

- **"code .**" opens the current folder in VS Code

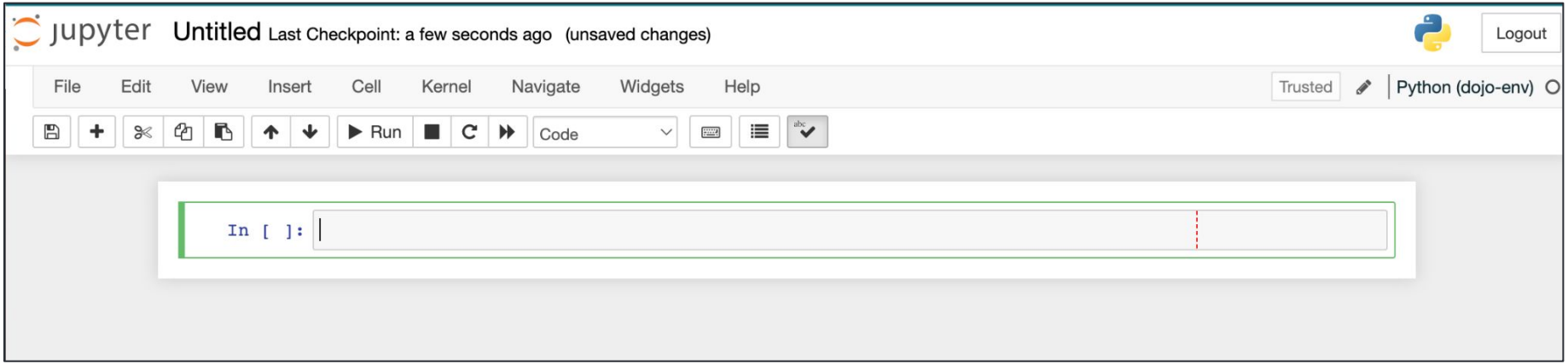
Jupyter Notebook

Jupyter Notebook Interface - Files View



- To create new Notebook:
  - Click New > Python (dojo-env)
  - A new “Untitled” notebook will open in a new tab.
- To add a file to the same folder as your repo:
  - Click Upload> Select the File > Confirm the filename and hit Upload

Jupyter Notebook Interface - Editor



CodeAlong Activity: Creating a New Project Repository

Using GitHub, GitHub Desktop, & Jupyter Notebook together

CodeAlong - Part 1: Start a New Project

- **For today's activity, you are going to be practicing working with GitHub Desktop, Jupyter Notebook, and local files.**
    - 1. Use the terminal to create a new DataEnrichment folder in your Documents.
    - 2. Use GitHub Desktop to create a new repository called something like "Wk13 Local Workflow Practice"
    - 3. Use GitHub desktop to launch a temrinal/gitbash window in the same folder as your new repo.
    - 4. From your terminal/gitbash, start the jupyter notebook server
    - 5. Download the files from this share url to your Downlaods folder.
        - Data science salaries 2021.csv
    - 6. Upload the files to a new "Data" folder in your repository.
    - 7. Create and open a new jupyter notebook.
    - 8. Rename the notebook "CodeAlong"

CodeAlong - Part 2: Do Your Work/Analysis

- 1. Inside your Jupyter Notebook, import the required packages:

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
```

- 2. Read in the .csv using pandas as a new dataframe using pandas.
- 3. Display the .info() and .head() of the new dataframe.

CodeAlong - Part 3: Shutting Down Jupyter

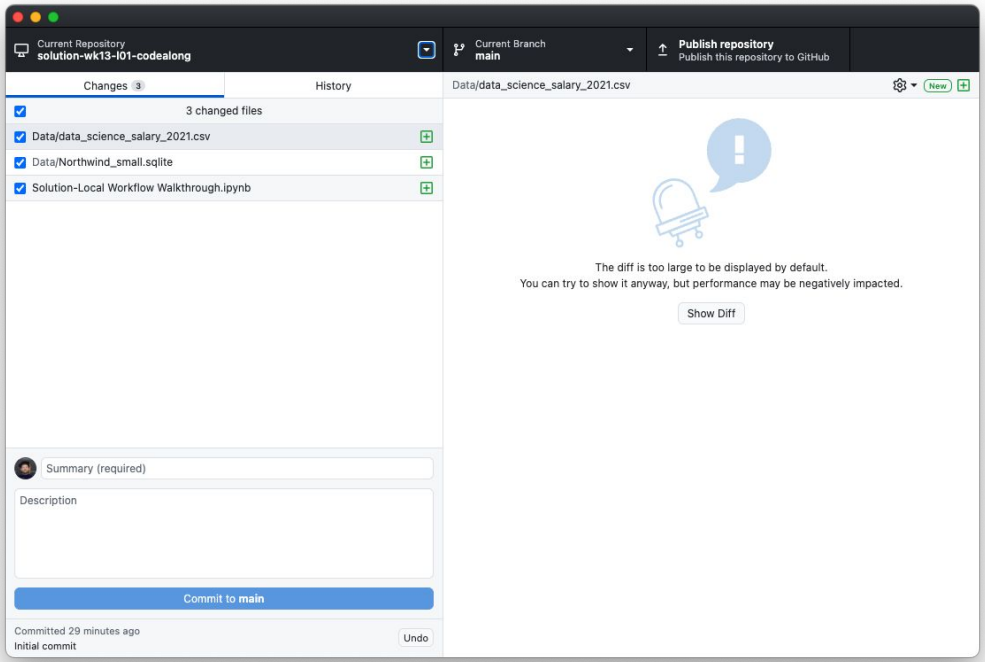
1. In your Jupyter Notebook:
    a. Click File > Save and Checkpoint (or the 💾 icon)
    b. If you're done working for the day:
        i. Click File > Close and Halt. Close the web browser tab (if it doesn't close on its own).

2. On Jupyter's File View Page:
    a. Click the **Quit** button on the top right
    b. Close your browser.

3. In your GitBash/Terminal:
    a. If the terminal stopped running the notebook and is waiting for input, close the window.
    b. If the terminal is still running jupyter: hit Control + C to force-quit.

4. Return to GitHub Desktop

CodeAlong Part 4: Commit Your Changes



- Enter a commit message in the "Summary (required)" field.
- Click "Commit to Main"
- Click Publish repository;
    - UNCHECK KEEP THIS CODE PRIVATE!

- Click Repository > View on GitHub to confirm you can see your work on the web version of your repository.

Part 5: Continue Working on the Repository

1. Open GitHub Desktop and select the repository you want to work on.
    a. Click on the “Current Repository” button on the top-left and select the correct repo from the drop down menu.
2. Click on the Repository menu > Open in Terminal/GitBash
3. In your terminal/ GitBash:
    a. Type “jupyter notebook”/”jnb” to start jupyter.
    b. Click on the notebook you want to work on.
4. Answer the following question using Pandas and/or visualizations:
    a. What is the mean 'Avg Salary(K)' according to “job_title_sim”?
5. Once you’ve produced your answer, save your work, shut down jupyter, and save and push a new commit to your repo.

Nice job!

Questions?

Local Workflow Review

Reference Slides

Steps to Start a New Project

1. Create a new repo
   a. In GitHub Desktop: Click File > New Repository.
   b. Give it a name and choose where to save it.

2. Open a Terminal/GitBash window in repo's folder.
   a. In GitHub Desktop > Click the Repository menu > Open in Terminal (or Open in GitBash)

3. Start Jupyter Notebook:
   a. Type "jupyter notebook" or "jnb" in your terminal/GitBash.
   b. Jupyter Notebook's File view will open in your web browser.

4. Create a new notebook:
   a. Click the "New" button in top right corner of Jupyter.
   b. Select "Python (dojo-env)"

5. Do your work/analysis in your new notebook.

Steps to Finish a Session

1. In your Jupyter Notebook:
   a. Click File > Save and Checkpoint (or the 💾 icon)
   b. Click File > Close and Halt. Close the web browser tab (if it doesn't close on its own).

2. On Jupyter's File View Page:
   a. Click the Quit button on the top right
   b. Close your browser.

3. In your GitBash/Terminal:
   a. If the terminal stopped running the notebook and is waiting for input, close the window.
   b. If the terminal is still running jupyter: hit Control + C to force-quit.

4. Return to GitHub Desktop
   a. Enter a commit message in the "Summary" field on the left sidebar.
   b. Click the blue "Commit" button.
   c. Click "Push Origin"

Steps to Continue Working on a Notebook

1. Open GitHub Desktop
    a. Click on the “Current Repository” button on the top-left.
    b. Find and click on the repo you want to work on.

===== *BELOW ARE SAME STEPS FROM Starting a New Project* ====

2. Open a Terminal/GitBash window in repo’s folder.
    a. In GitHub Desktop > Click the Repository menu > Open in Terminal (or Open in GitBash)

2. Start Jupyter Notebook:
    a. Type “jupyter notebook” or “jnb” in your terminal/GitBash.
    b. Jupyter Notebook’s File view will open in your web browser.

3. Create a new notebook:
    a. Click the “New” button in top right corner of Jupyter.
    b. Select “Python (dojo-env)”

4. Do your work/analysis in your new notebook.