Steps for using git with JupyterLab:

1. Go to the Anaconda Navigator and install ‘git’ and ‘jupyterlab-git’.
   1. I think this package may come from a different conda channel (conda’s way of organizing packages). You may need to click the ‘Channels’ button and add ‘conda-forge’, a channel that has a lot of common packages in it
   2. This can also be installed through the terminal with the command ‘conda install -c conda-forge jupyterlab-git’
2. Go to the extensions tab in Jupyterlab (it looks like a puzzle piece), click the button to enable extensions, and search for ‘git’. Install the extension called ‘@jupyterlab/git’
3. Close and re-open JupyterLab from Anaconda Navigator. There should now be a new widget on the left side that looks like a diamond. This is the classic ‘git’ symbol
4. Using the folder widget on the left, navigate to the place where you want to clone a repository
5. Now click on the Git widget (the diamond) on the left.
   1. The next steps can also be done via the ‘Git’ tab at the top of JupyterLab
6. Click on the button that says ‘Clone a Repository’
7. Paste the link for the forked repo that you want to clone (instructions for where to get that link are in the sep document)
8. Click the ‘Clone’ button
9. That repo should be copied to that location in your file system.
10. Once you make changes to the items in that repo. JupyterLab will keep track of those changes on your local machine, but they aren’t yet saved in the online repository.
11. The steps for saving something to an online git repository is stage > commit > push
12. Saving your changes to the online Github repository is called ‘pushing’ your changes. To do this:
    1. From within the folder where the repo is on your machine in JupyterLab, click on the git tab on the left
    2. The files that have changed will be listed under the ‘Changed’ category
    3. If you have created any new files, they will be listed under the ‘Untracked’ category
    4. To stage changes (prepare them to be committed), hover over the file and click the ‘+’ symbol. That file name should move to be in the ‘Staged’ section
    5. To commit all of your currently staged changes, make a commit message down in the summary box at the bottom (answering the question, ‘what kind of changes did you make/why?’) and then press the blue ‘Commit’ button
    6. You’ll notice that this adds an orange dot to the button at the top that looks like a cloud with an arrow pointing up. Click this to finally push/send your committed changes to your online Github repository

ghp\_zs0M176jZcEFHcteIEzBxAImK98Yrw1UE8o8

Resources for learning more about git and version control:

* [Introduction to Git - Training | Microsoft Learn](https://learn.microsoft.com/en-us/training/modules/intro-to-git/)
* [Learn the Basics of Git in Under 10 Minutes (freecodecamp.org)](https://www.freecodecamp.org/news/learn-the-basics-of-git-in-under-10-minutes-da548267cc91/) – walks through the ideas of git in general (pushing, pulling, staging, committing, etc.)