Using PyCharm to manage repository

Install PyCharm

- Download from https://www.jetbrains.com/pycharm/download/

- Be sure to get the Community Edition

- Run the installer once the download is finished

Set-Up PyCharm

- Click the Configure drop-down then choose settings

- In the settings menu, search for GitHub. Enter your GitHub login and password and click Ok

- Change other settings as needed. I recommend changing the Keymap to an editor you are already familiar with
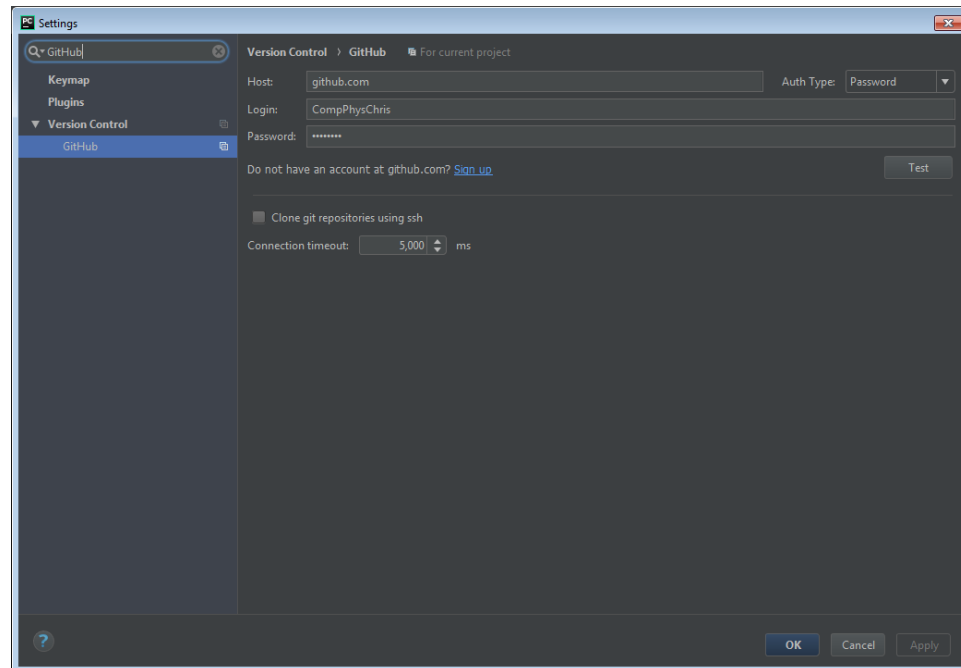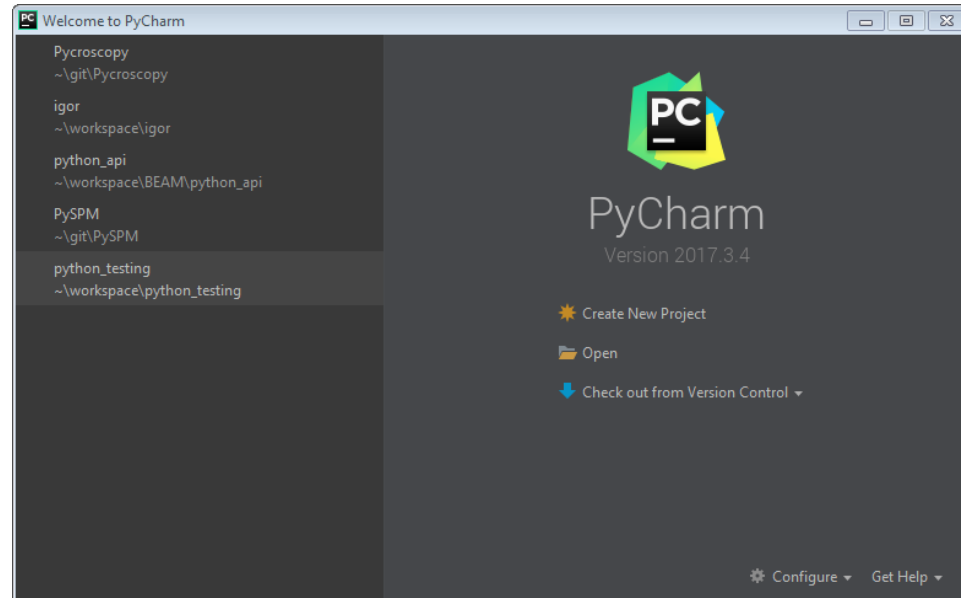
Basic Git Ideas/Commands

- Clone – download a new repository to the local machine
- Branch – different versions of the same repository.
  - Local branch – only exists on your computer
  - Remote branch – exists in the GitHub repository
- Commit – record all changes to the local branch
- Pull – get the latest Commits from the remote branch and apply them to the local branch
- Push – copy all new Commits from the local branch to the remote branch
- Stash – store all unCommited changes to the local branch
- UnStash – apply the changes in a Stash to the local branch
- Conflict – two changes from Commits or a Stash both attempt to alter the repository in an incompatible way
- Merge – choose how to implement Conflicting changes

Clone the Repository

- Click check out from Version Control and choose GitHub
- Repository URL: https://github.com/pycroscopy/pycroscopy.git
- Click Clone
- The repository will now be download into a new directory
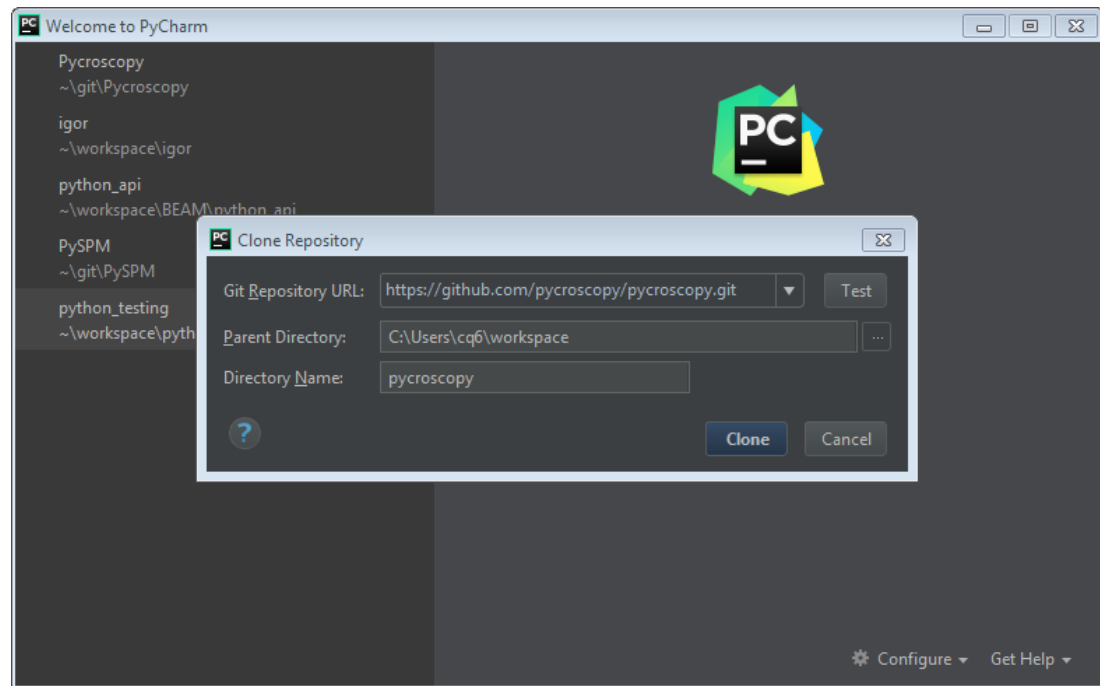
Git in PyCharm
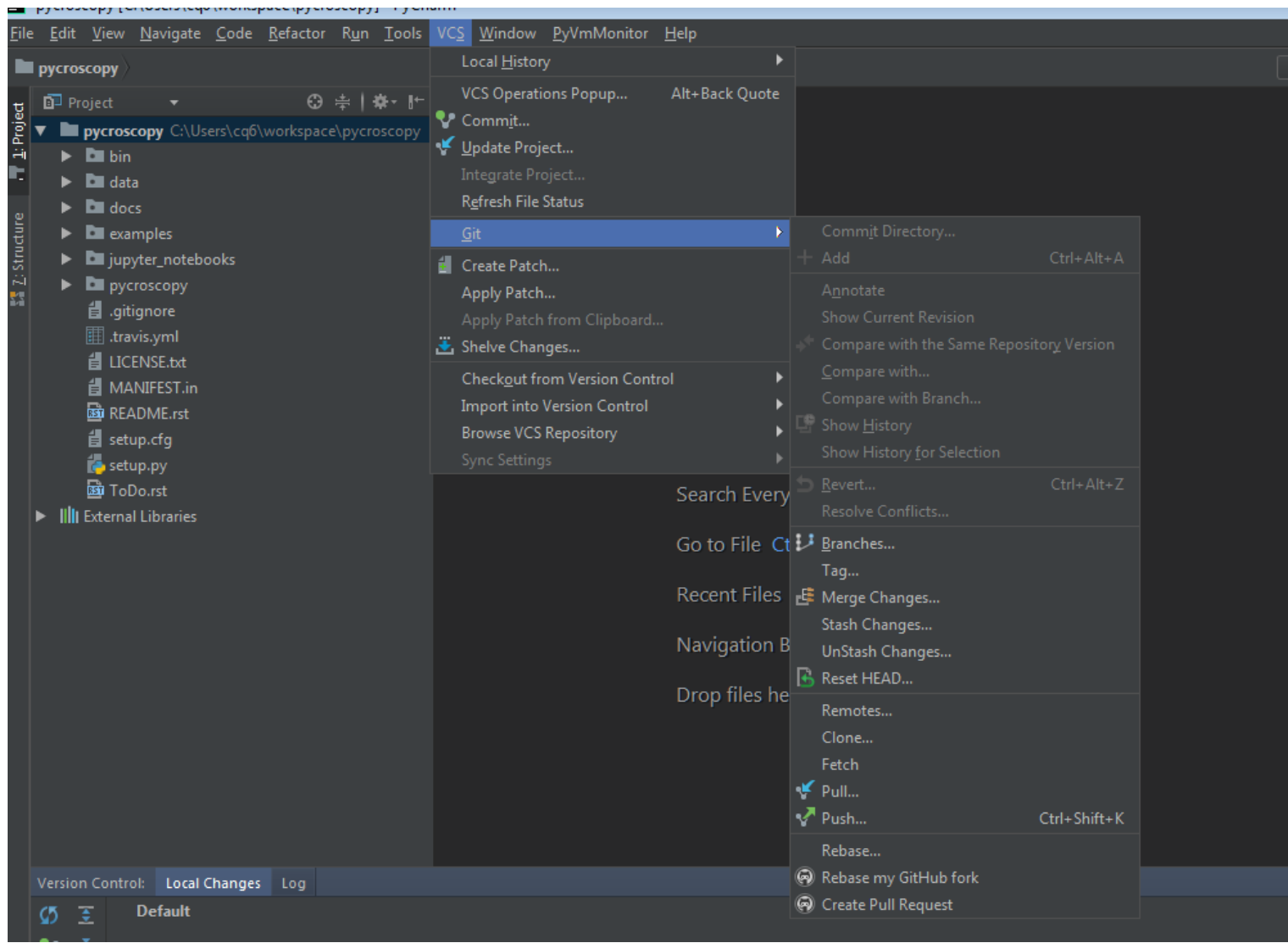
Change Branch

- Click branches
- Local branches are those that exist on your computer
- Remote branches are those on GitHub
- You can download a remote branch by clicking checkout
- You can switch between local branches by clicking the new one and selecting Checkout
- You may need to Stash or Commit local changes before you switch braches
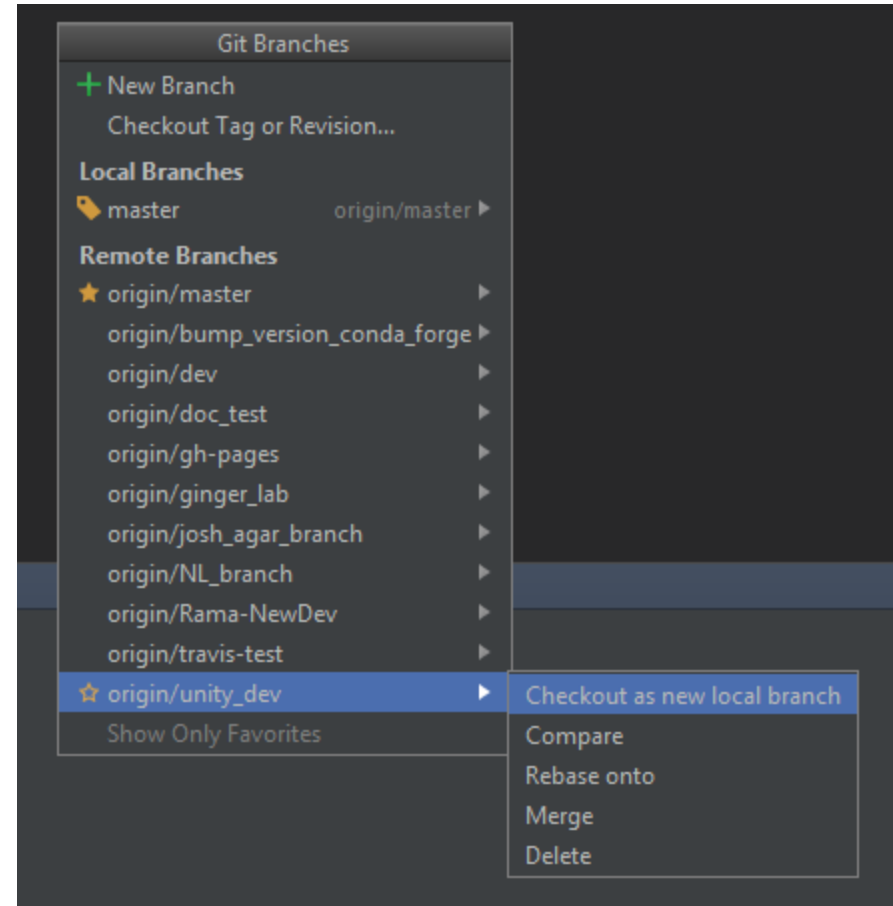
Updating the Current Branch

- Getting the latest changes is easy, just click Pull
- If you have local changes that you haven't Commited yet, you should Stash Changes before Pulling
- After the Pull completes, you can UnStash
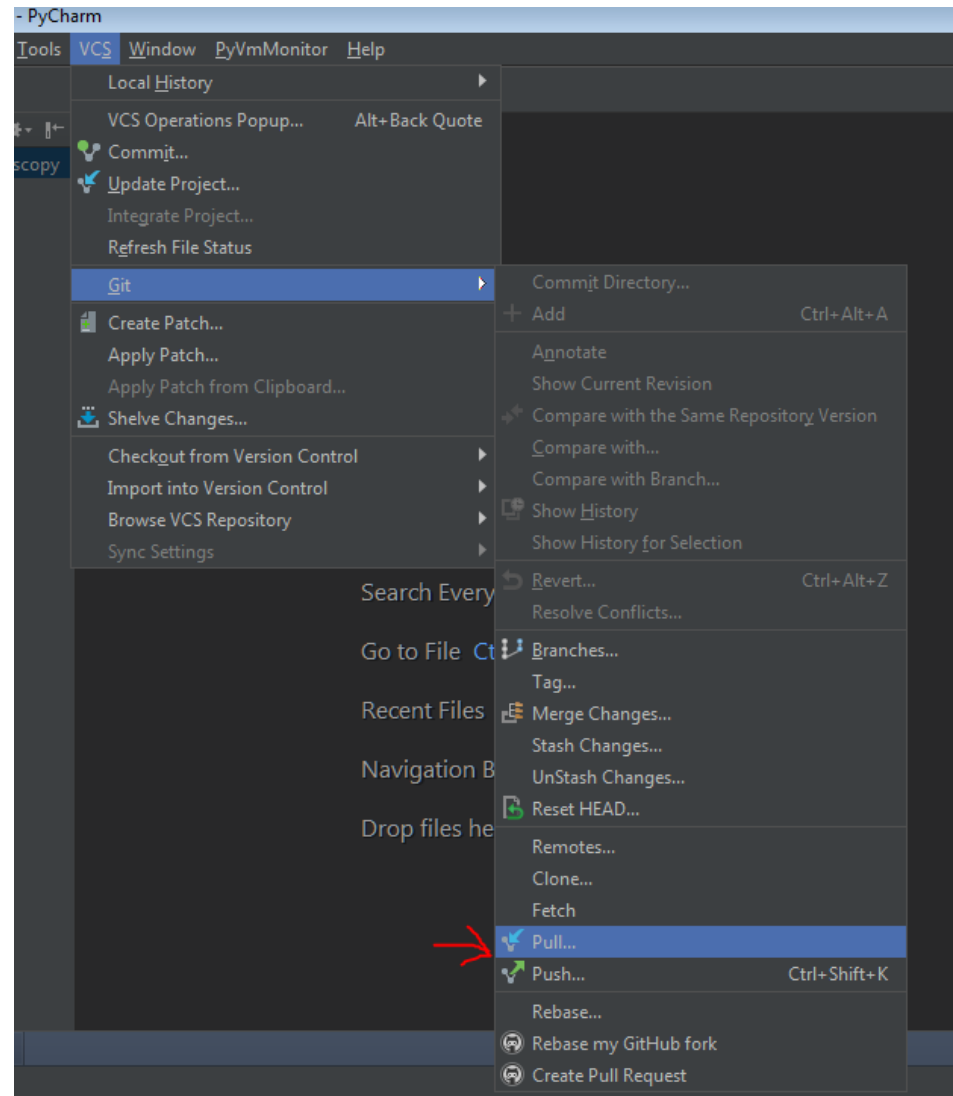
Commit Changes

- You can see all local changes in the version control tool window.

- To commit your changes click the Commit button in the Version Control window

- Or click Commit under the VCS menu
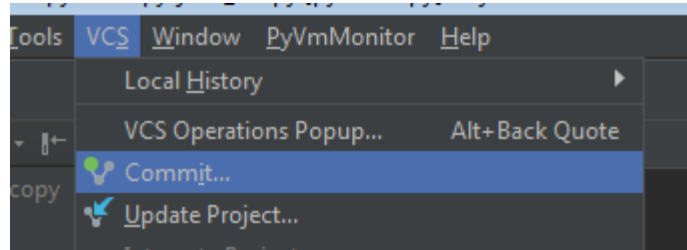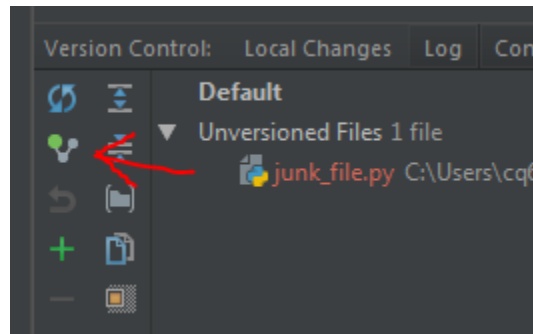
Commit 2

- In the Commit window, you will see the affected files
- You must choose which file(s) you want included in the Commit by clicking the check-box(es)
- If you click a file, you can see the differences from the local version
- Type in a short message describing the changes you made then click Commit
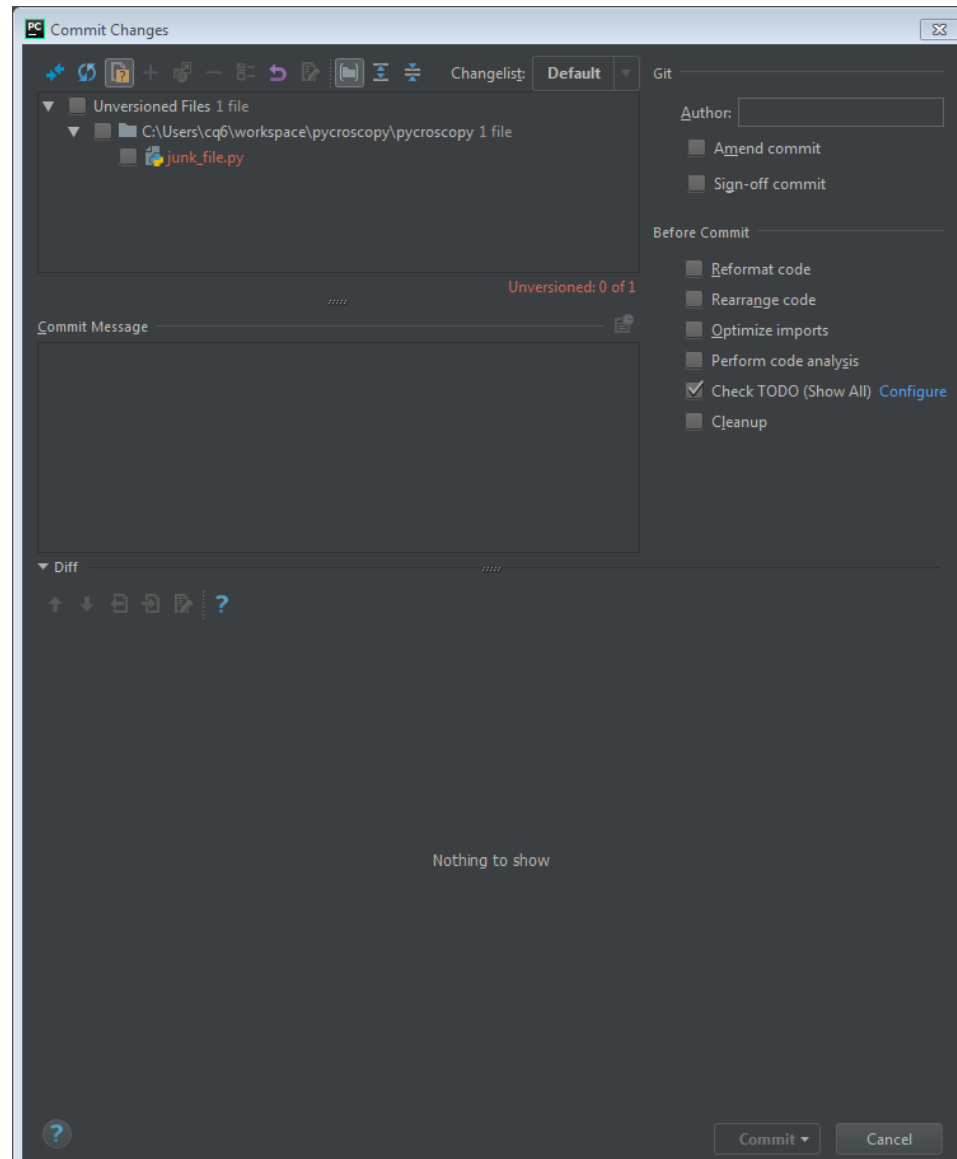
Push Changes

- Before Pushing, Commit any changes you want to Push, then Pull the latest changes from the Repository.

- To Push, select VCS-Git-Push

- You will get a window showing all unPushed Commits

- Click Push and your commits will be applied to the current remote branch
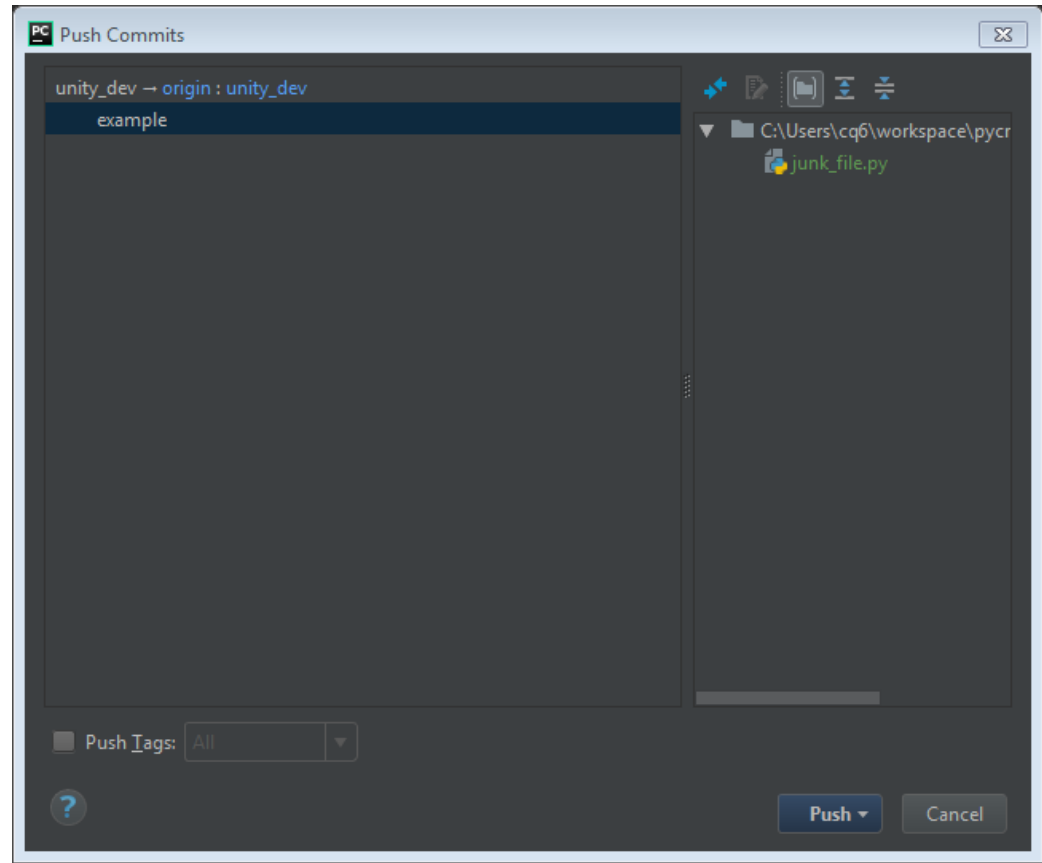
Conflicts and Merging

- If a Conflict occurs, you will see a new window

- To resolve the Conflicts, click Merge to open the three panel Merge tool
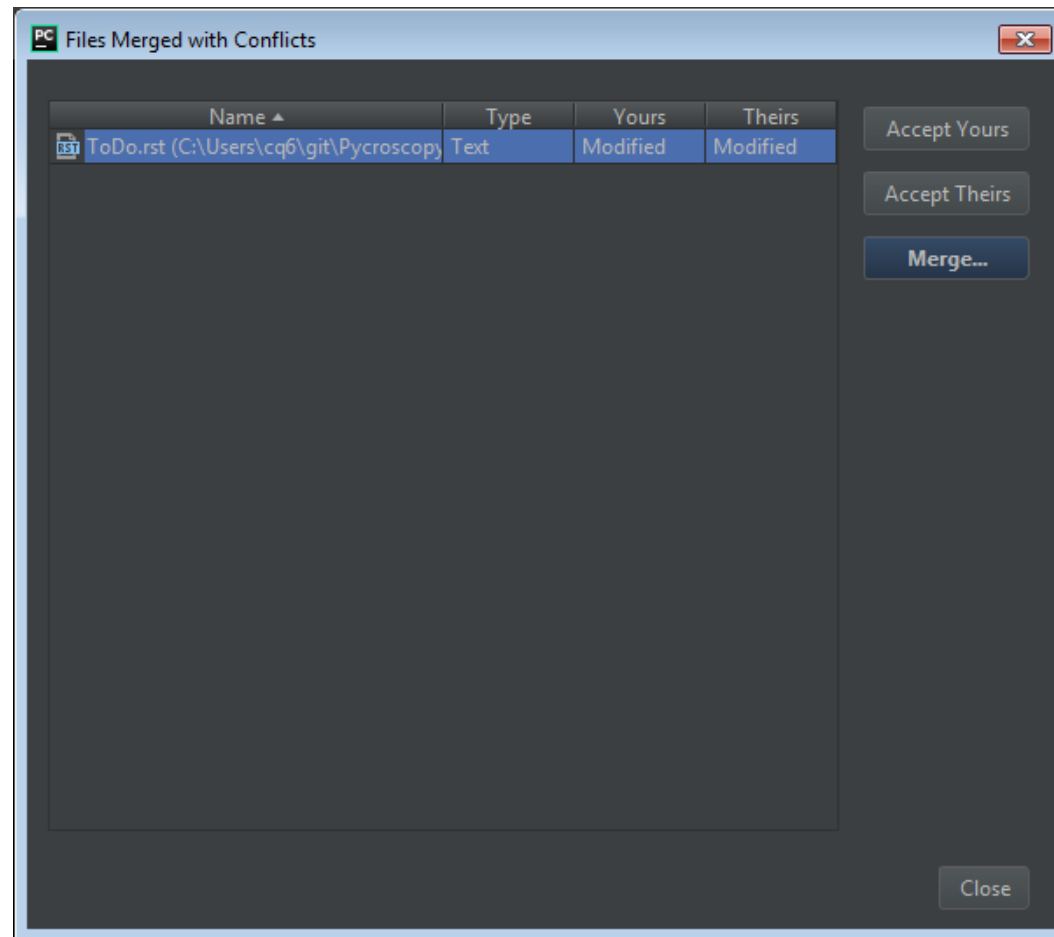
Merging

- The left panel shows your changes, the right shows the other changes, and the middle shows the current result
- You can choose either left or right for any conflicting region or manually enter a third option by typing in the middle panel
- Once you are done, click Apply and Commit the Merge changes
- Don't forget to push the Merge Commit once you are done