

# Graider GitHub Instructions

# Repository Access Page

- The following instructions assume you have a GitHub account and have given your instructor your GitHub username.
- Go to the link for the Repository Access Page provided by your instructor.
- Select the line that corresponds to your information to be taken to your GitHub repository for the assignment.

## csc1110 — CSC1110: Software Development Week 6 Homework Repositories

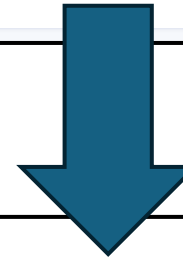
Term: 27s1. Assignment: Week 6 Homework (hw6).

Find your MSOE username below and open your repository.

If you do not see your username or cannot access your repository, contact your instructor.

### Section 111a

velez@msoe.edu



27s1-csc1110-hw6-adelavelez Private

main 1 Branch 0 Tags

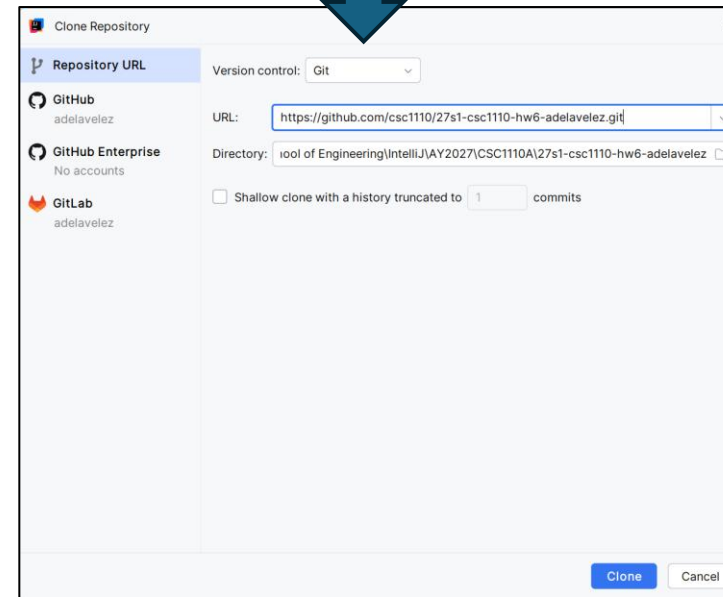
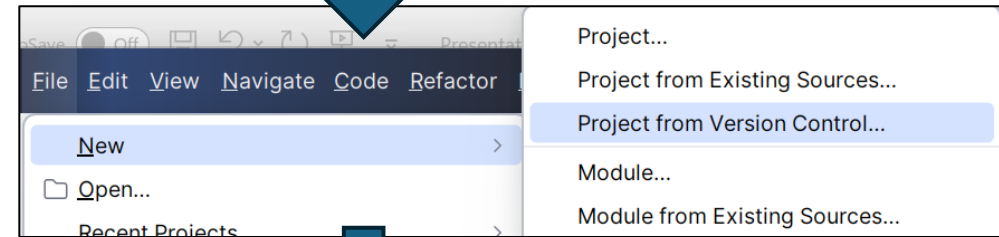
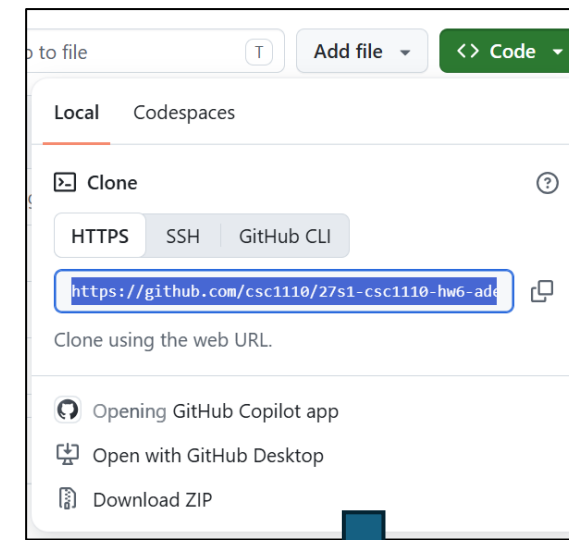
Go to file T Add file >> Code

adelavelez Configure Graider grading workflow db9463f · 2 days ago 2 Commits

|                   |                                    |            |
|-------------------|------------------------------------|------------|
| .github/workflows | Configure Graider grading workflow | 2 days ago |
| src               | Initial commit                     | 2 days ago |
| .gitignore        | Initial commit                     | 2 days ago |
| README.md         | Initial commit                     | 2 days ago |

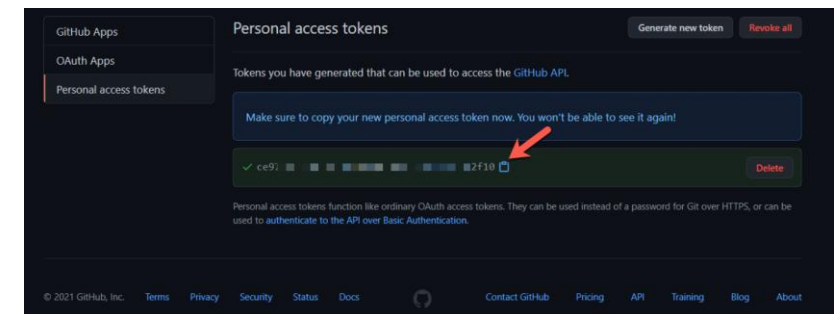
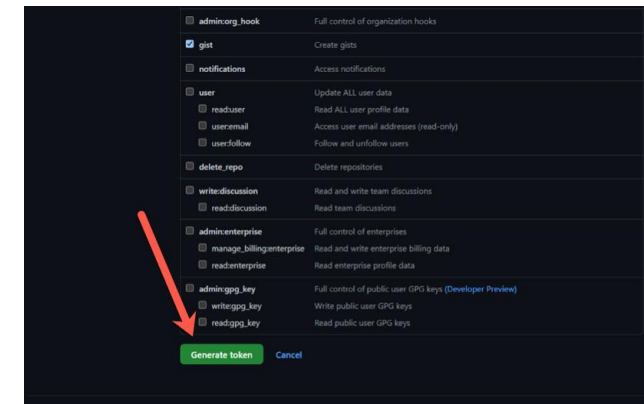
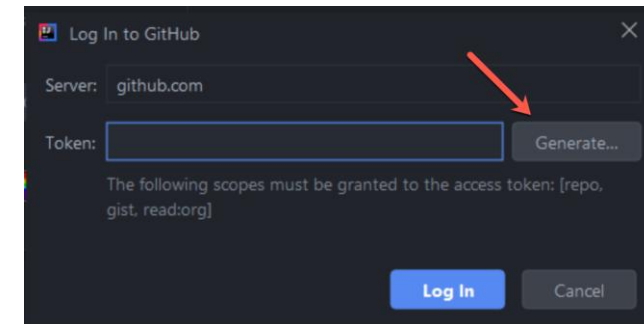
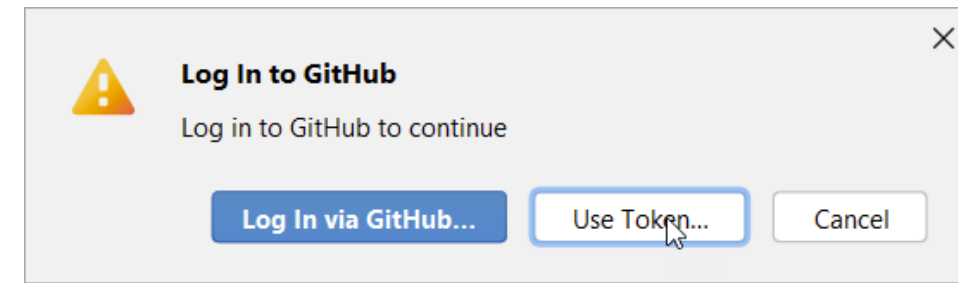
# Cloning Repository

- Click on the green Code button and copy the https URL.
- Go to IntelliJ, select File > New > Project from Version Control
- Paste the URL you copied into the URL field.
- Make sure you ok with the directory location that the project will be created in and then click Clone.

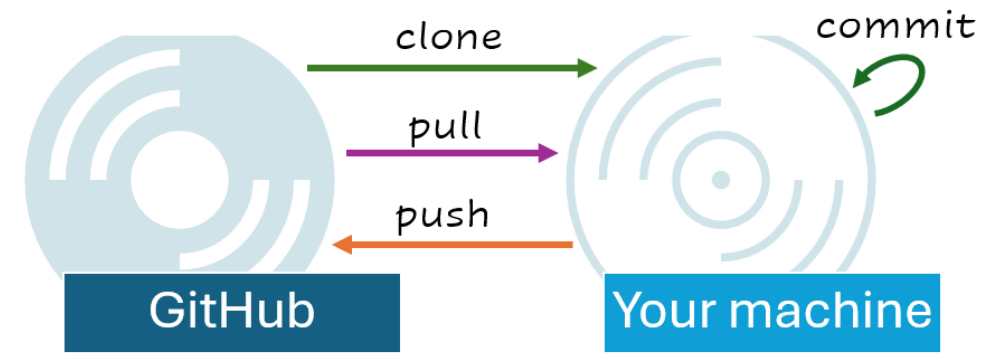


# Getting Access

- To allow IntelliJ to clone the repository locally to your computer, you must give it the authorization to access it the repository.
- The most reliable way to give authorization is with the “Use Token” method
- If you are logged into your GitHub account; this will take you to a page to generate a token that will give IntelliJ the authorization to access the repository.
- It is suggested that you set the expiration date (near the top of the page) to after the end of the semester.

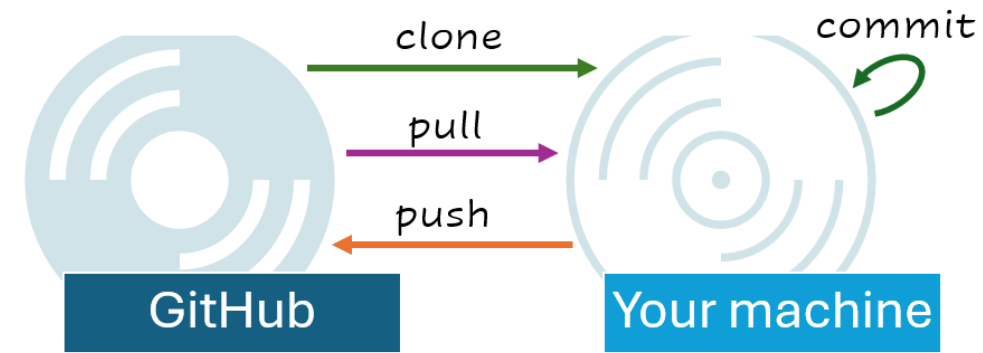


# Committing and Pushing Changes



- When you clone a repository into IntelliJ, you are making a local copy of the repository.
- A remote copy of the code still exists on GitHub at the URL you copied previously.
- When you add, remove, or modify a file on the local copy, you want to sync that change with the remote copy of the code.
  - The remote copy is what will be accessed and graded by your instructor.
- To do this you first must commit the changes and then push.

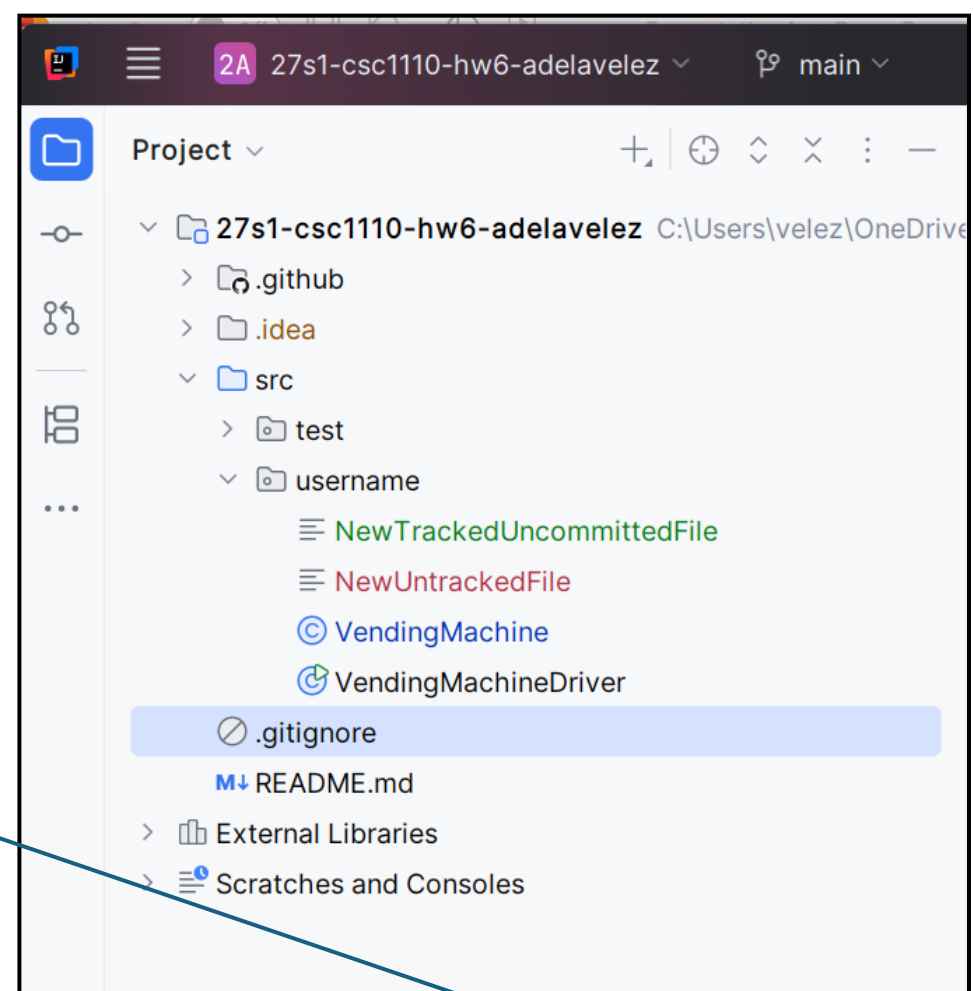
# Committing and Pushing Changes



- When you commit in IntelliJ, you are staging the changes to your repository, which is the local copy of the code.
- Pushing will take your committed changes and attempt to sync them with the remote repository.
- Committing is like getting all the items ready for a car ride and pushing is getting in the car and leaving.
- You can commit as many times as you want before doing a push.
  - It is recommended to push often during the development of a project because if your machine were to fail, you would at least have an updated copy of the project on GitHub.

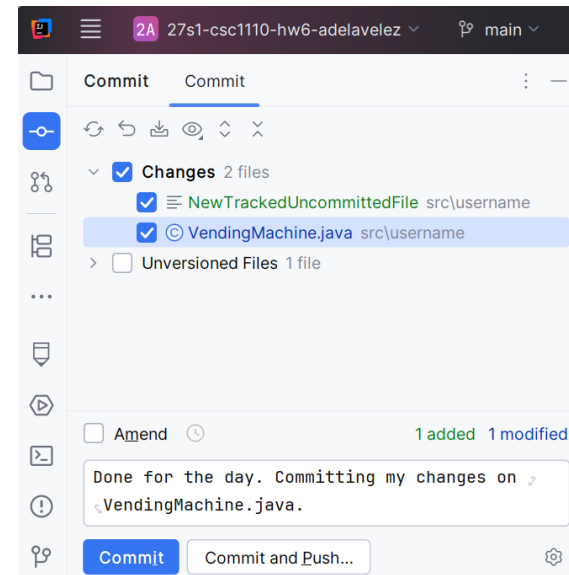
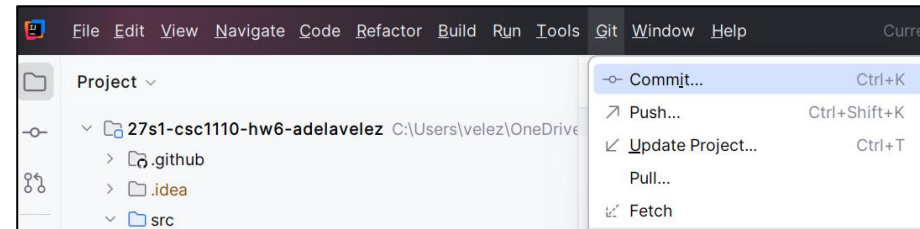
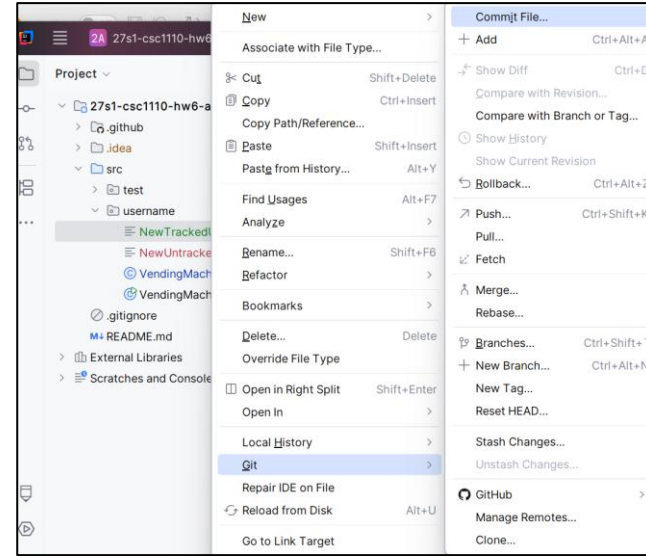
# File Changes

- Whenever a file has been changed or added, IntelliJ should change the color of the filename to indicate there are changes not committed.
  - Green means just added and not committed and blue is not committed.
- Whenever you add a file to the repository, IntelliJ will ask you if you want to add the file to Git.
- If you do not add the file to Git, then its changes can not be committed and it will not be synced to the remote repository.
  - The filename will show up in red.
- You can always right-click a file, go to the Git option, and select Add to add it to Git



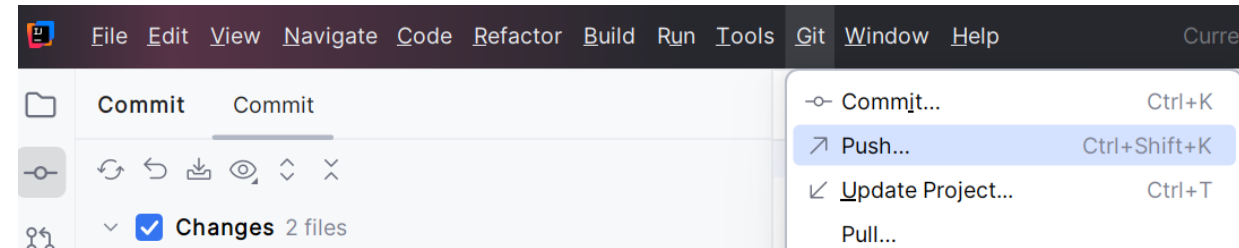
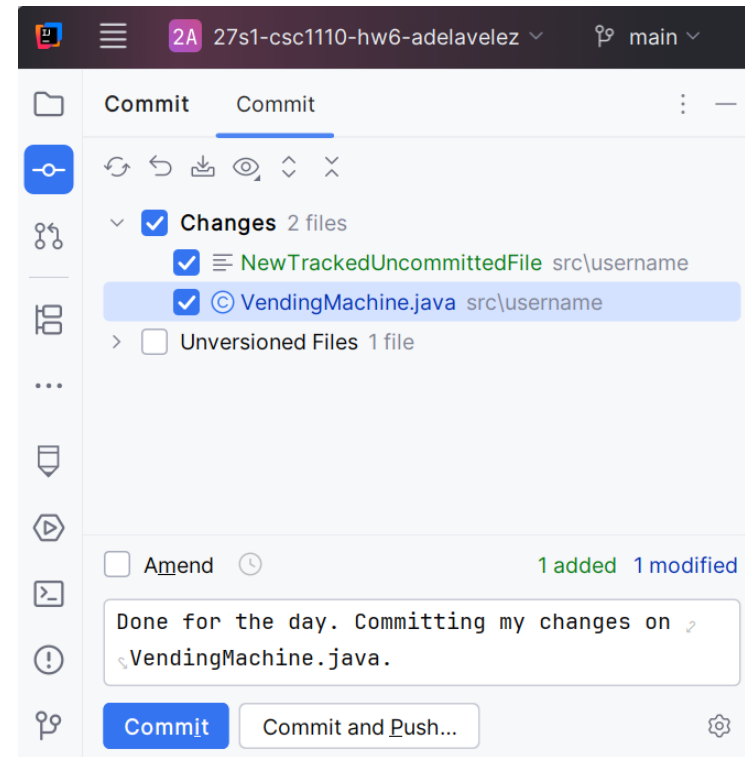
# Committing

- To commit a file, you can
  - Right click the file, select Git, and select commit
  - Click on the -o- symbol on the left panel or press Ctrl-K to bring up the commit window
  - Go to the horizontal bars > Git > commit
- All these methods should take you to the commit window where you can select the files to commit and provide a commit message.
  - From here click commit to commit the changes.



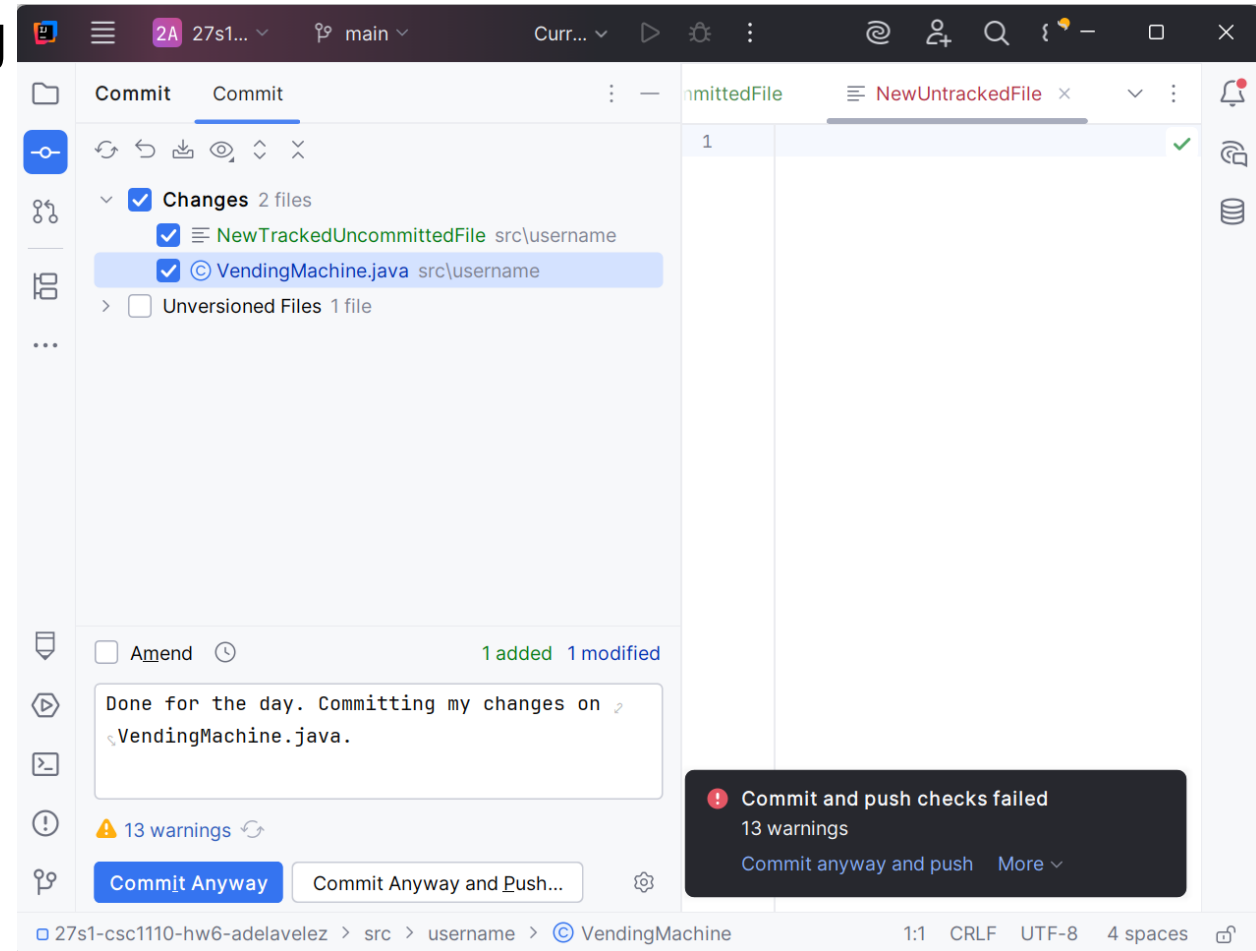
# Pushing

- To push the committed changes, you can either click “Commit and Push” to do both actions at once.
- Or select the “Push” option via the Git menu to push already committed changes.



# Push Warnings!!!

- Sometimes when you push, IntelliJ will try to be helpful and tell you that the code you are about to push has warning or errors.
  - It's fine to push code with warnings or errors especially if you are saving your progress as you are working on the project.
- You must select the “Push Anyway” or “Commit Anyway and Push” option to get IntelliJ to actually push the code.



# Pushing

- Pushing should pop up the window shown which will show you the changes to be pushed.
  - Click on the button to push.
- Once you have pushed your changes, go to the URL for the repository to verify that the remote copy was updated.
- You should see that the repository was changed recently and the commit message matches the commits you just pushed.

