DedicatedRAMs – Development Manual

Plunder Chess

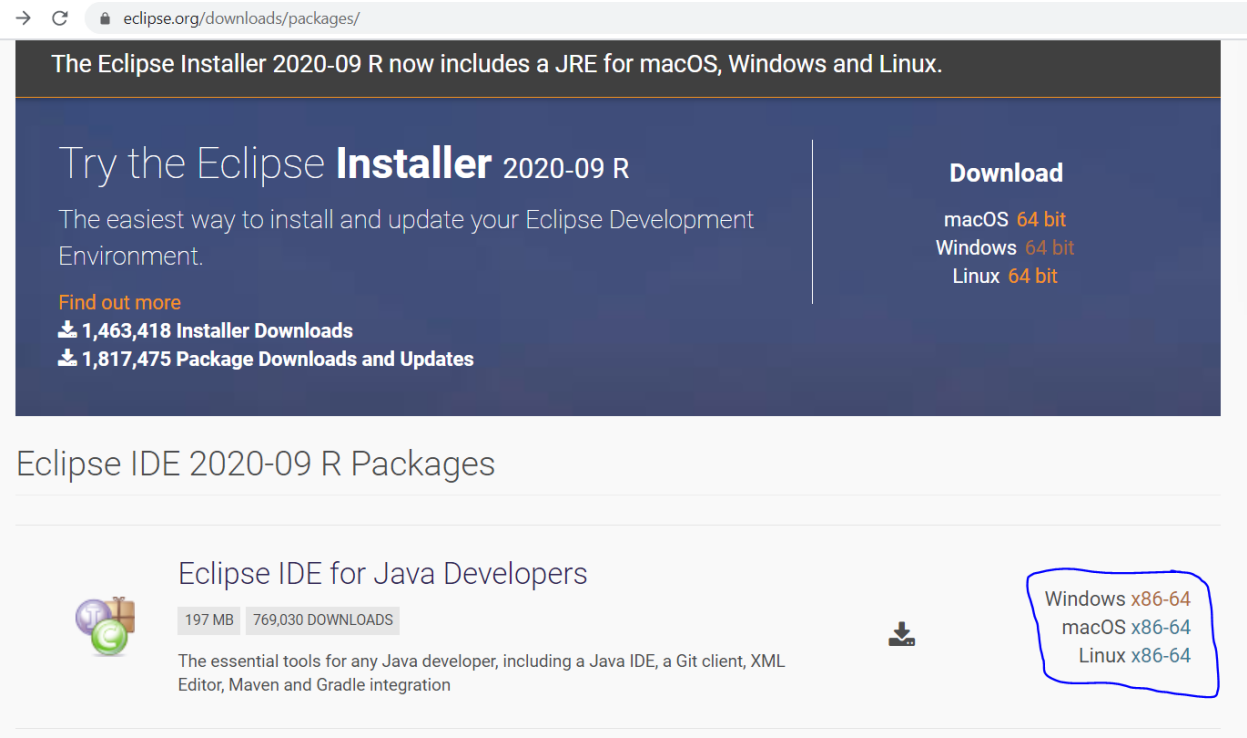
Version 1

Welcome to Development with the DedicatedRAMs as we build an online Plunder Chess game. In this manual, we will explain step-by-step how to set up the development environment we a working in, how to run our system as a developer, and how to run the code tests. This manual is meant for Windows though development can be done on Linux or MacOS systems. Let us jump right in!

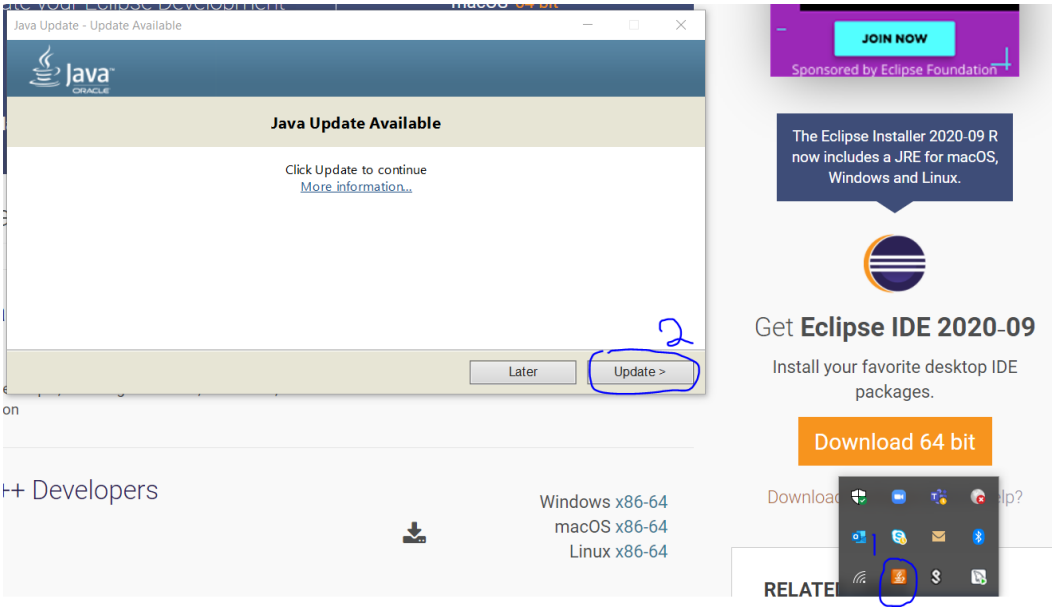
How to set up the development environment?

1. We utilize Eclipse as an IDE for developing code. If you do not have Eclipse installed on your computer, please install it.

- Installing Eclipse: Go to https://www.eclipse.org/downloads/packages/ and install the Eclipse IDE 2020-09 for Java Developers. If you have a previous version of eclipse, you should be able to use the older version still.

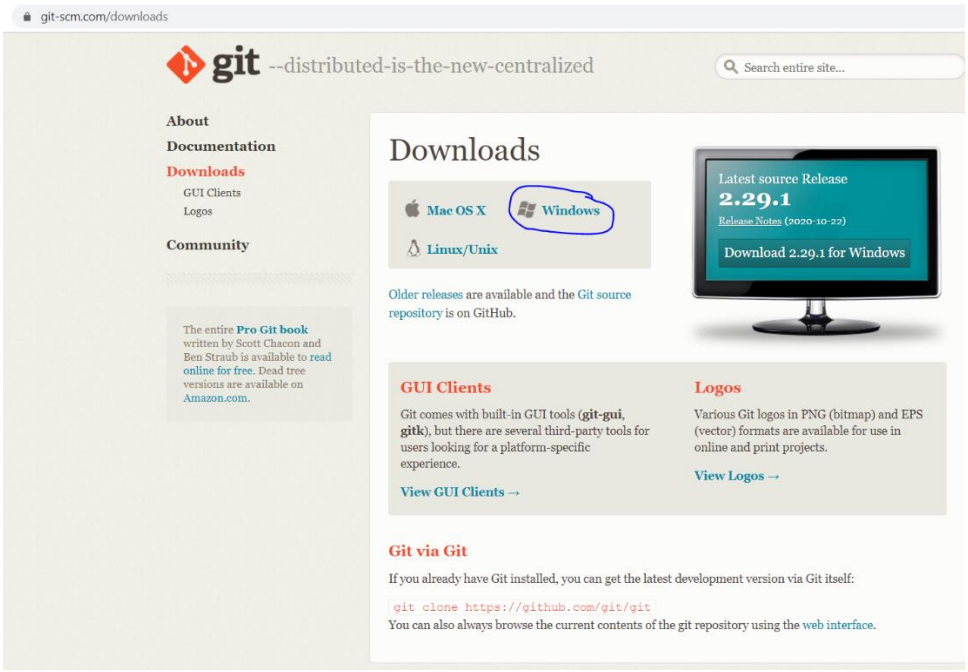


- Sometimes your Java version may need updated, if it does, see the java updater at the bottom right of your screen.



2. Now you have Eclipse installed on your computer and are ready to utilize git to grab our project from Github. With the right plugins, you can use git from Eclipse or install git on your computer and use the command line.

- Go to https://git-scm.com/downloads and click on the appropriate install on git. It will lead you through the setup for git.



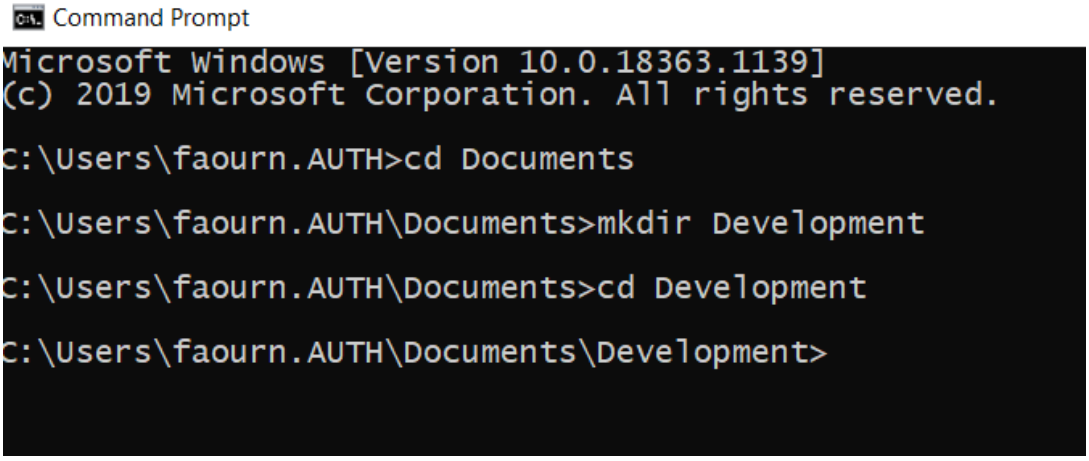
- Test that git is installed by going to the command prompt and typing the following into the command prompt:

```
C:\Users\faourn.AUTH>git --version
git version 2.28.0.windows.1
```

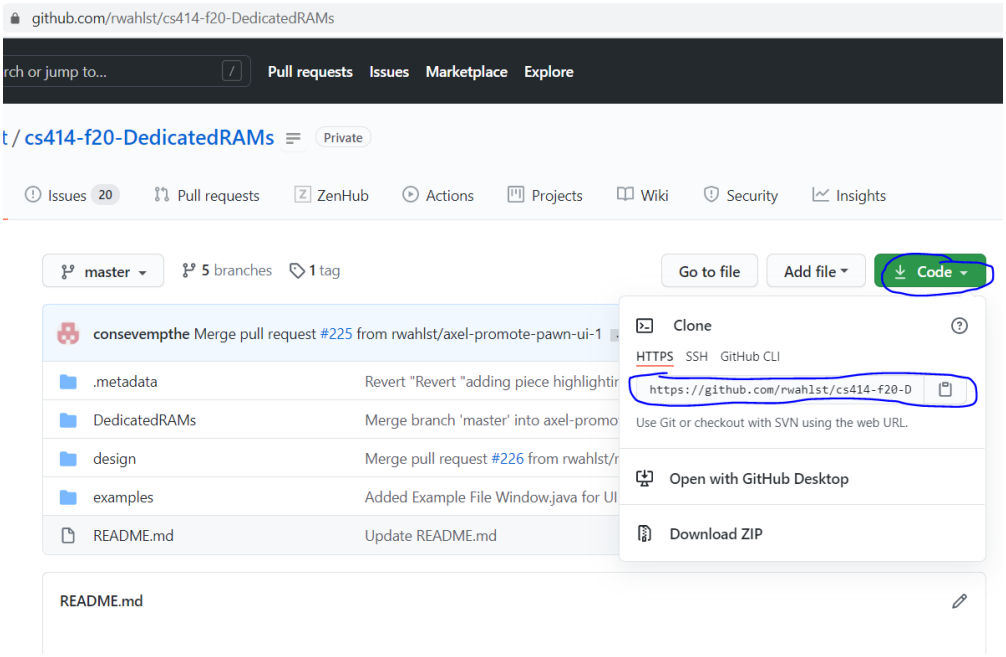
It should give you the version you have installed.

3. Time to clone the project.

- Open a command prompt and navigate to the folder you want the project to be in.



- Go to the DedicatedRAMs Github page, https://github.com/rwahlst/cs414-f20-DedicatedRAMs to get the HTTPS address for the repository.



- In the command prompt you opened, run the following command where the https address is the one you copied from the repository:

```
C:\Users\faourn.AUTH\Documents\Development>git clone https://github.com/rwahlst/cs414-f20-DedicatedRAMs.git
Cloning into 'cs414-f20-DedicatedRAMs'...
remote: Enumerating objects: 67, done.
remote: Counting objects: 100% (67/67), done.
remote: Compressing objects: 100% (58/58), done.
remote: Total 2956 (delta 20), reused 19 (delta 6), pack-reused 2889
Receiving objects: 100% (2956/2956), 20.62 MiB | 10.86 MiB/s, done.
Resolving deltas: 100% (1695/1695), done.

C:\Users\faourn.AUTH\Documents\Development>
```

- Navigate into the cloned project using the following command:

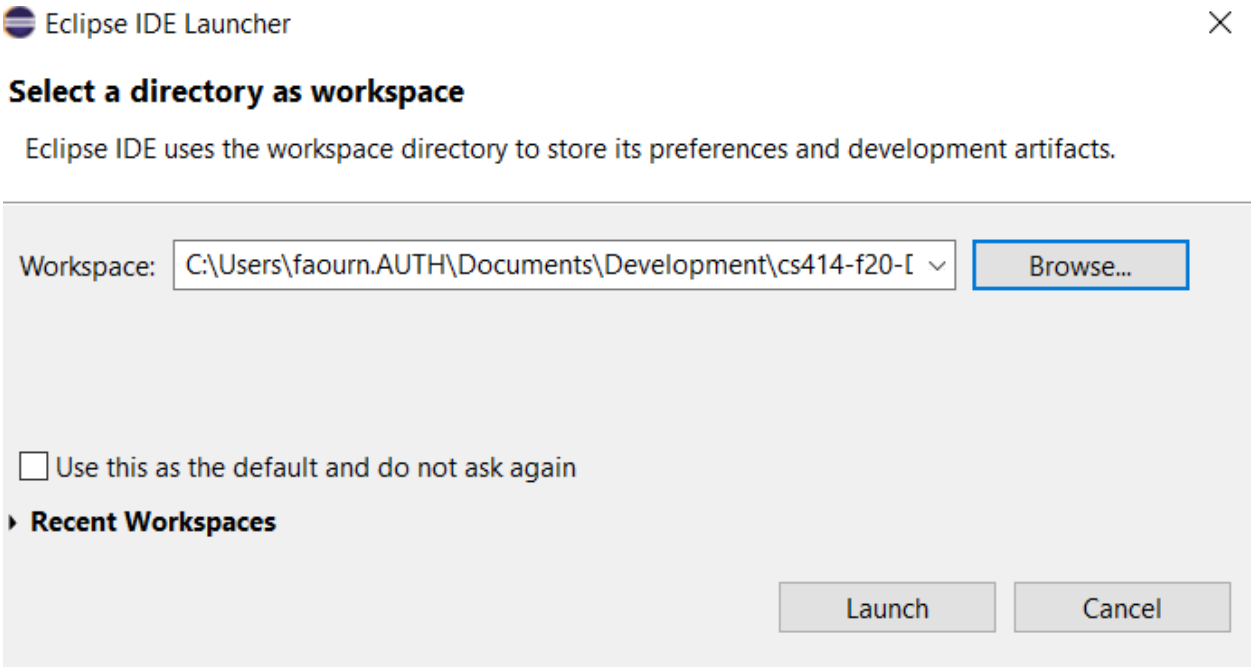
```
C:\Users\faourn.AUTH\Documents\Development>cd cs414-f20-DedicatedRAMs

C:\Users\faourn.AUTH\Documents\Development\cs414-f20-DedicatedRAMs>
```

- Run git branch to see the current branch and other local branches.

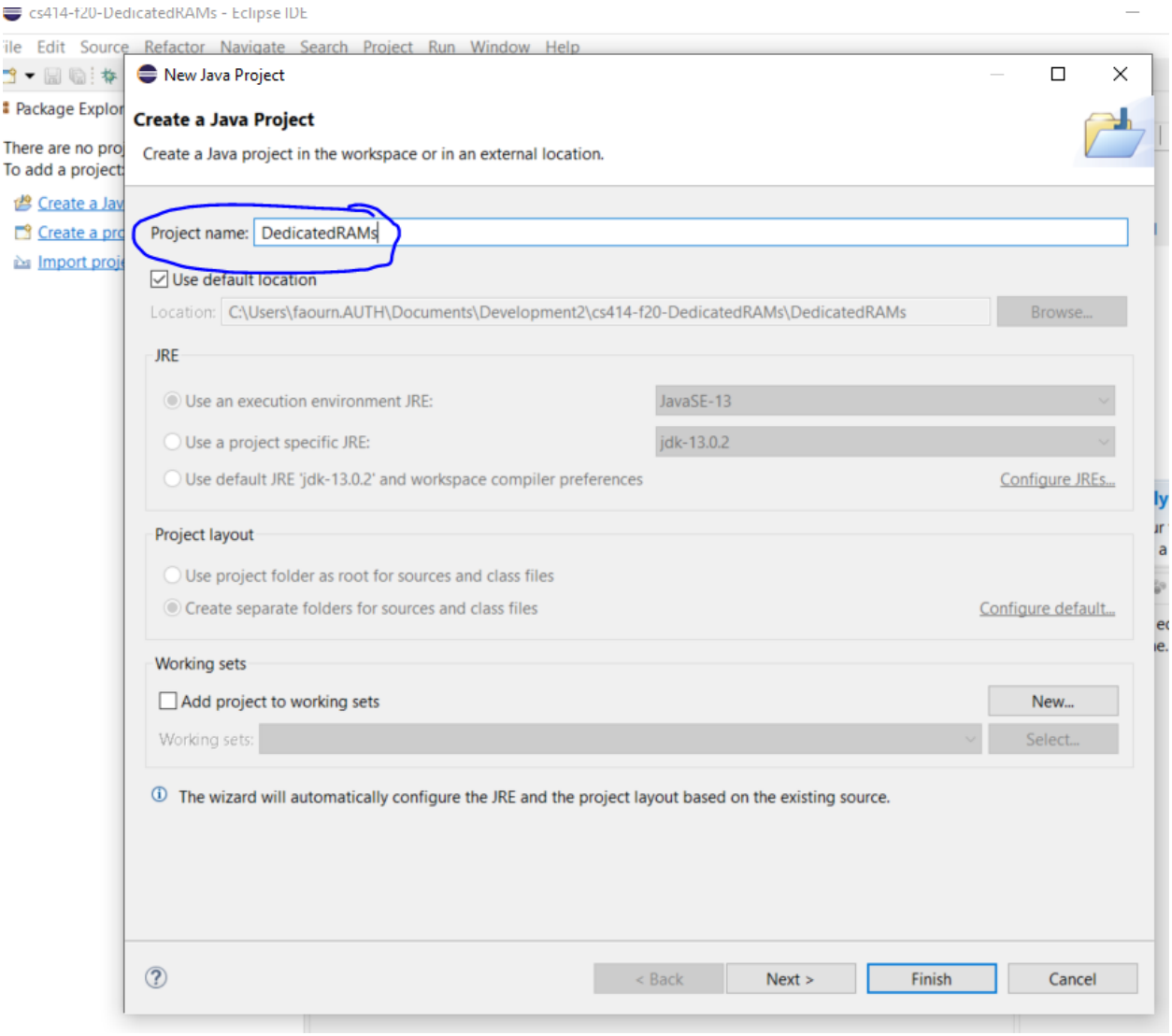
```
C:\Users\faourn.AUTH\Documents\Development\cs414-f20-DedicatedRAMs>git branch
* master

C:\Users\faourn.AUTH\Documents\Development\cs414-f20-DedicatedRAMs>
```

You know have the project. Be sure to follow all version control common practices. No developing on master. Always create a new branch before developing and pushing to the remote repository.

4. Open Eclipse from the start menu and set the workspace to the repository you just created:

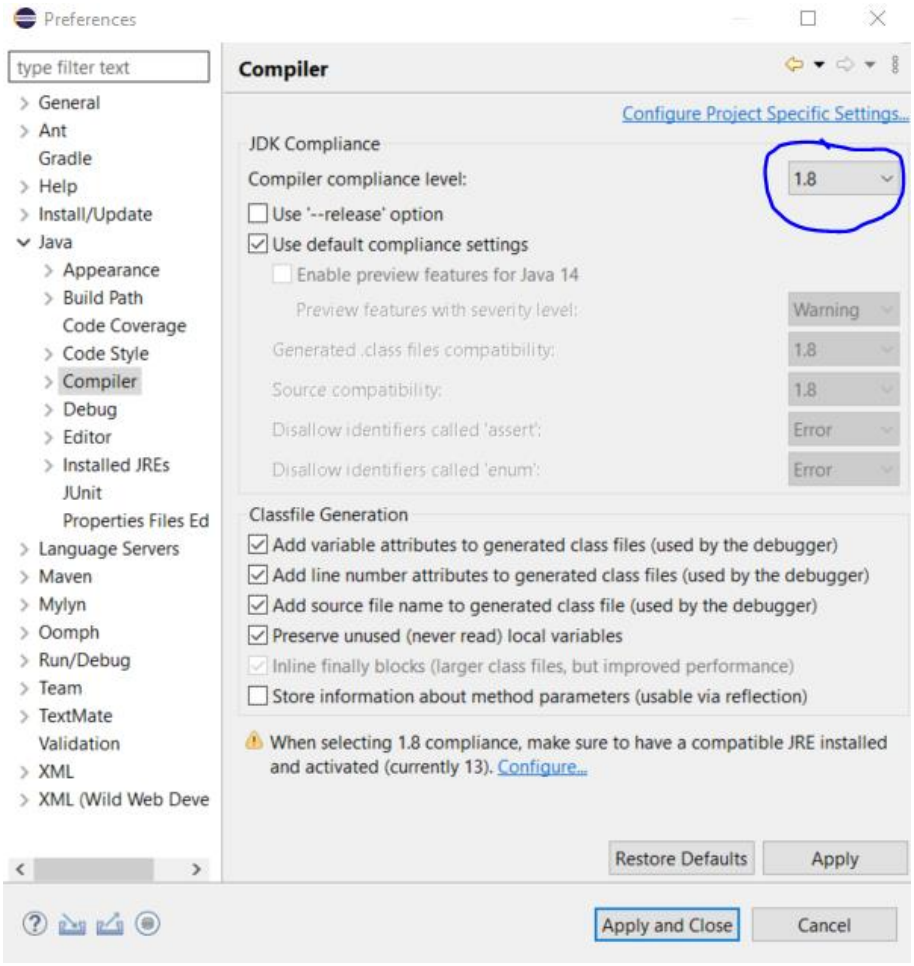


- Create a Java project from the package explorer in Eclipse. Name it: “DedicatedRAMs”



- Click “Next”, “Finish”, and “Don’t Create” for the module-info.java.

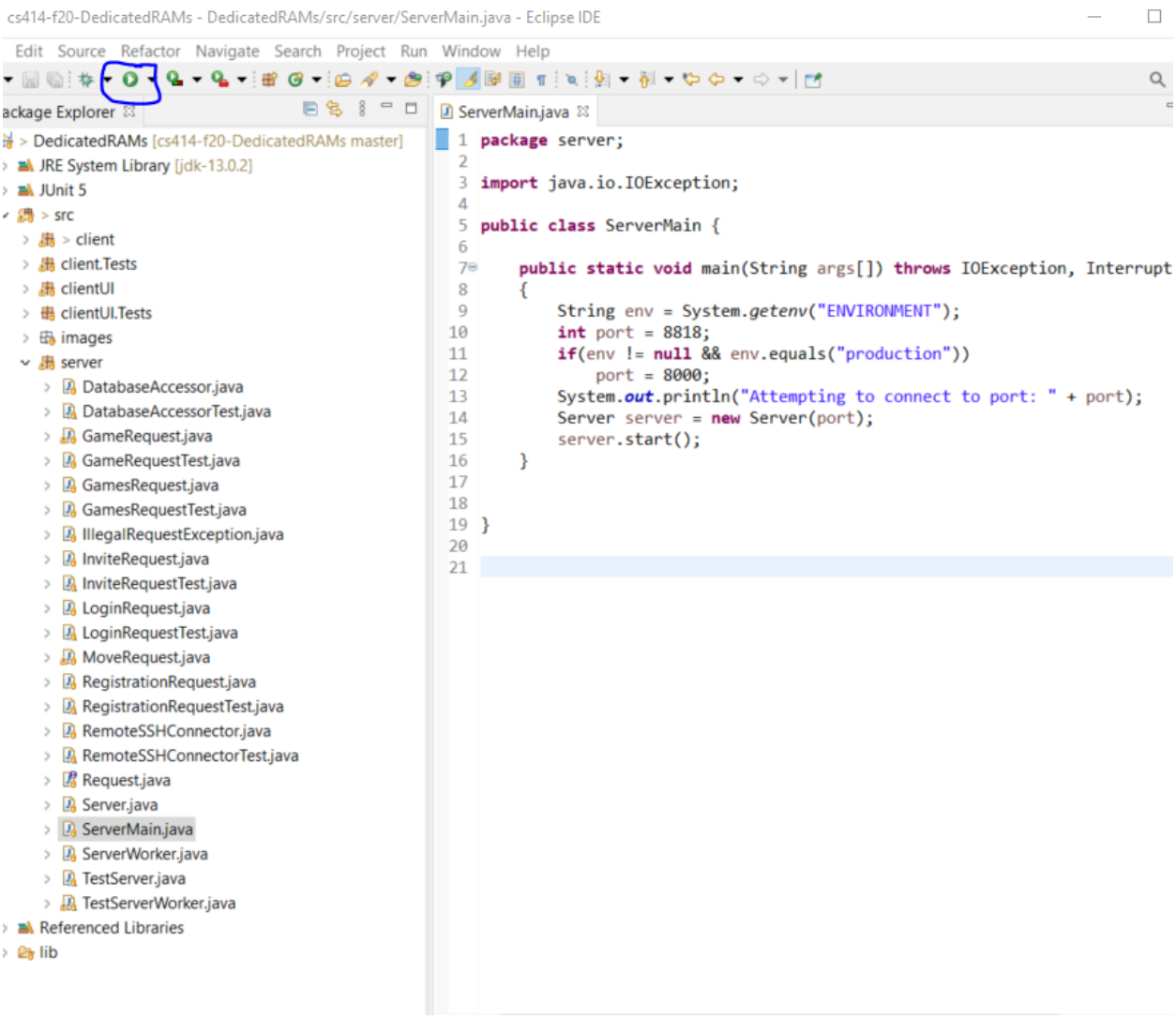
5. For compatibility, go to “Window” -> “Preference” -> “Java” -> “Compiler” and set Complier compliance level to 1.8. Then click “Apply and Close”.

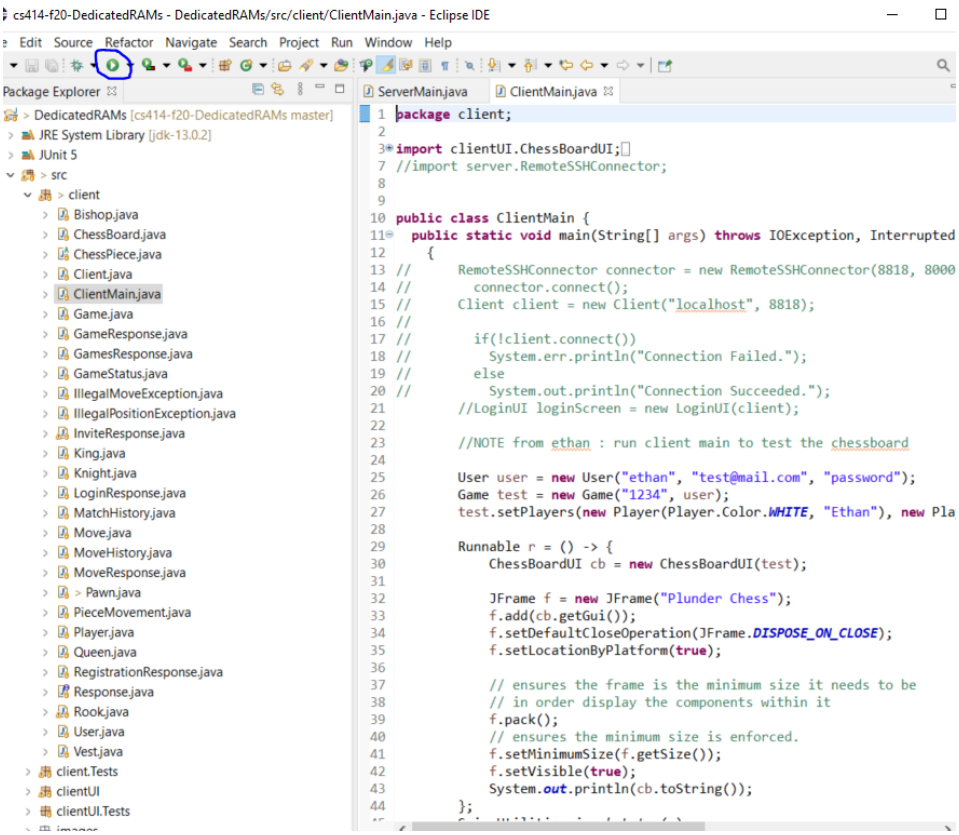


The project is now set up and ready for development. Contact team members with additional questions.

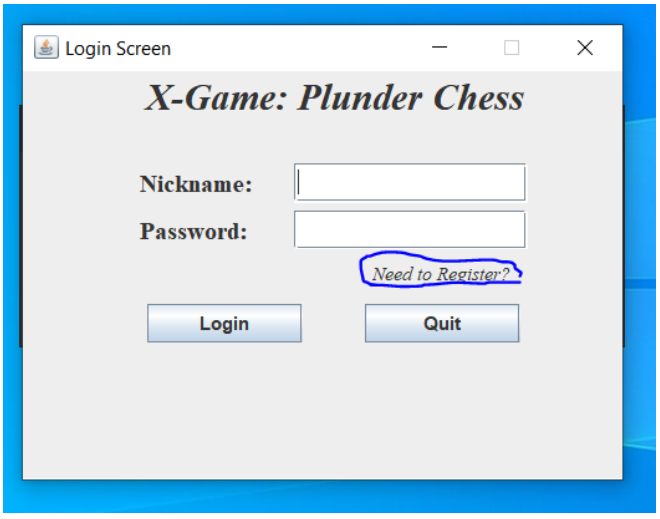
How to run the system once the project is setup?

1. To run the system locally, you can open ServerMain.java and ClientMain.java within Eclipse. To run the Server locally, click “Run” with ServerMain.java open. To run the Client, click “Run” with ClientMain.java open. Refer to the two figures below:



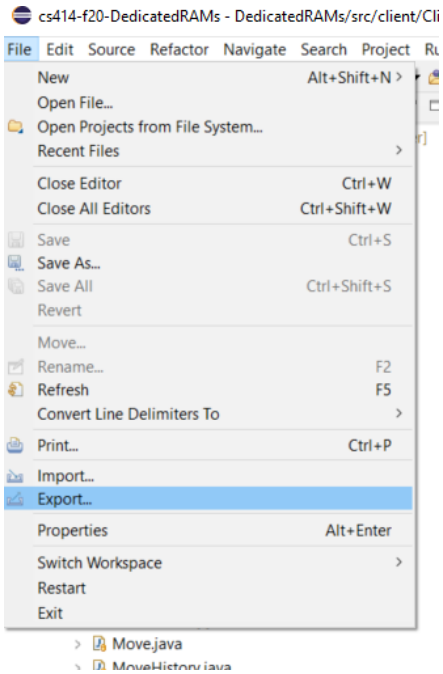


- The Client will open the login page. You can register by clicking on the “Need to Register” button or login with preexisting login information. See figure below:

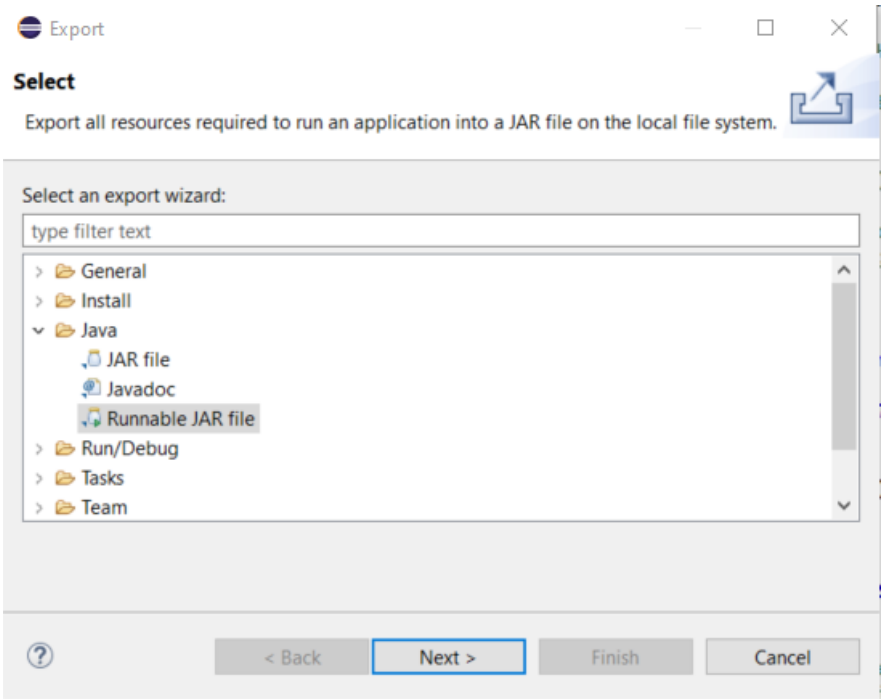


2. To run the Server on the CS computers, you need to export ServerMain as a jar file. See the figure below:

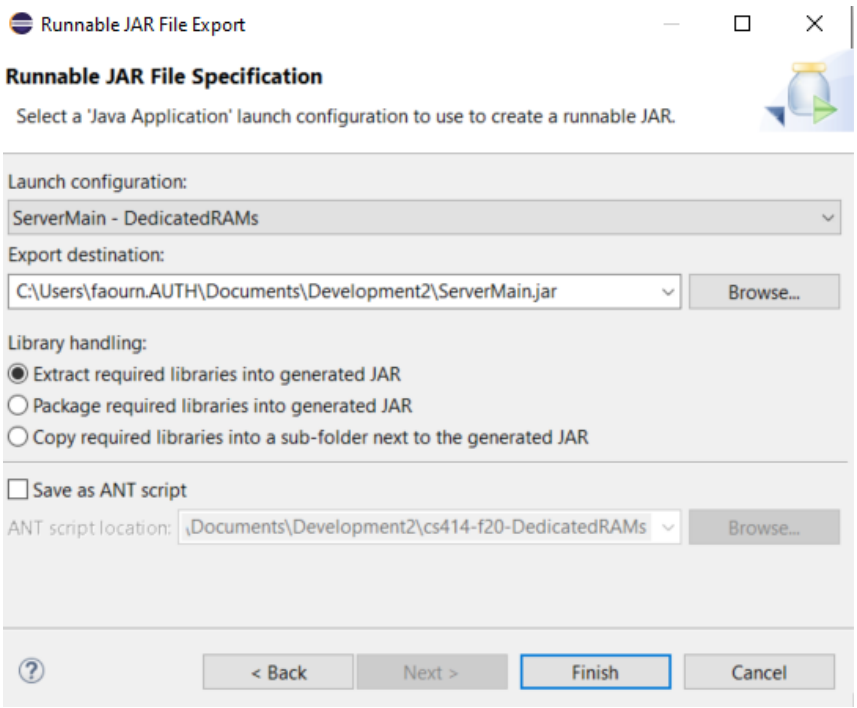
- Start by going to “File”, “Export”.



- Select “Runnable JAR file” under Java and click “Next”.



- Select ServerMain – DedicatedRAMs as the Launch Configuration. Select your export destination and click “Finish”, “OK”, and “OK” to complete the export.



- Use the command prompt to send the .jar file created to our server space on concord.cs.colostate.edu. Replace the username “najyfaou” with your username.

```
C:\Users\faourn.AUTH\Documents\Development2>scp ServerMain.jar najyfaou@concord.cs.colostate.edu:/s/bach/a/class/cs414/cs414e
najyfaou@concord.cs.colostate.edu's password:
ServerMain.jar                                100% 4821KB   1.2MB/s   00:04
```

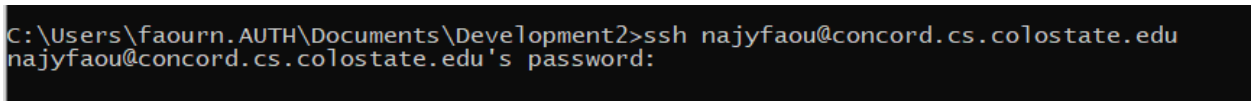
- ssh into concord.cs.colostate.edu and run the Server using the following commands. Replace the username “najyfaou” with your username.

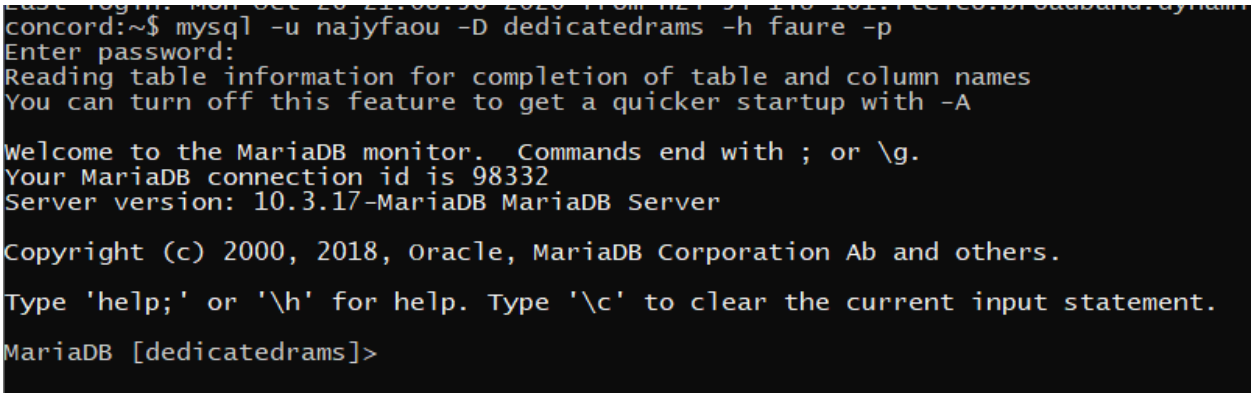
```
C:\Users\faourn.AUTH\Documents\Development2>ssh -t najyfaou@concord.cs.colostate.edu "cd /s/bach/a/class/cs414/cs414e ; bash"
najyfaou@concord.cs.colostate.edu's password:
concord:/s/bach/a/class/cs414/cs414e$ export ENVIRONMENT=production
concord:/s/bach/a/class/cs414/cs414e$ java -jar ServerMain.jar
Attempting to connect to port: 8000
Connected
Port Fowarded!
Ready to accept connection...
```

The Server is now running on the CSU computer. Any Client should be able to connect to the Server from different computers while the Server runs on the concord. It is not set up for other computers. Use concord.cs.colostate.edu unless it is unavailable, in which case, contact the other team members to make connection changes.

The Client can now be run locally with local and remote server connection. Use local development for Server until testing remote server.

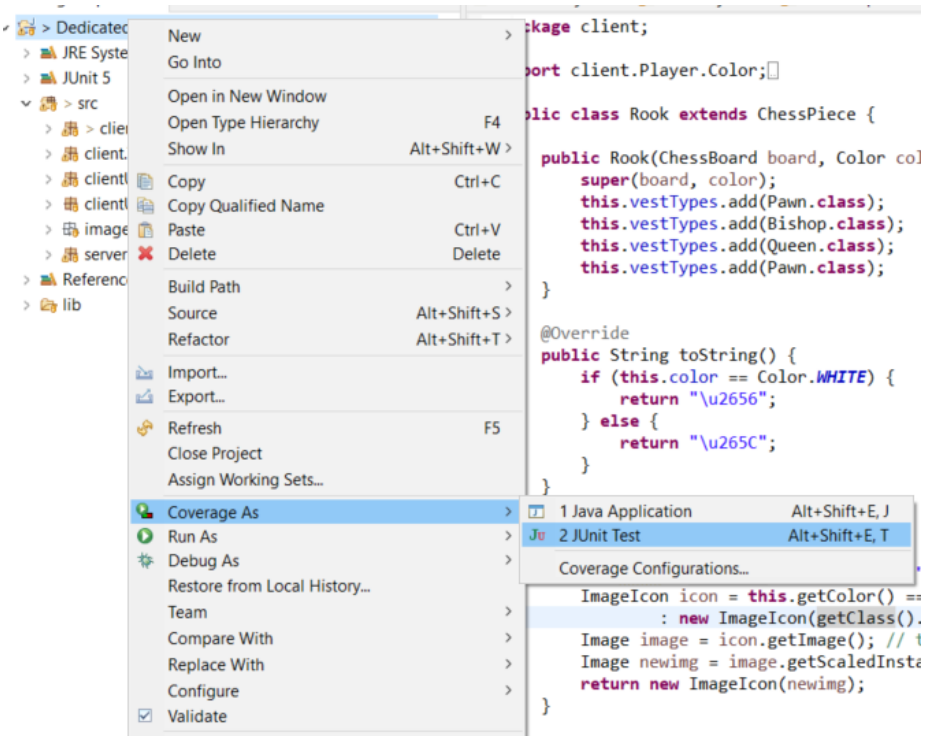
3. Accessing the database directly. Along with running the system, a developer may want to access the database to perform changes to it directly. Use the following commands with your username and password to access the database.



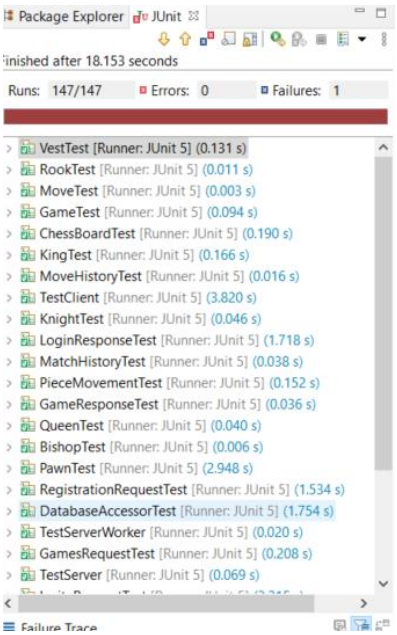


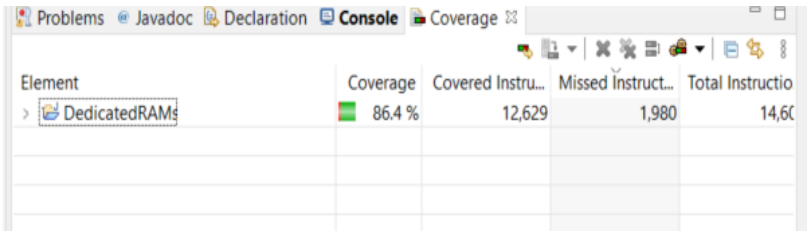
How to run tests in Eclipse?

1. Running all tests is simple. You can right click on the project, go to “Coverage As” and select “Junit Test”. This will run all tests.



- Tests results and a Coverage percent will appear upon completion of the tests. An example can be seen here:





How to make changes to the code?

1. First, be sure that you are working in a separate branch from master. It is easy to overwrite and break master if you develop in it.

- Using the git plugin for Eclipse or git in the command prompt as shown before, create a new branch. Familiarize yourself with git commands if you have little prior experience.

```
C:\Users\faourn.AUTH\Documents\Development2\cs414-f20-DedicatedRAMs>git checkout -b development_fix
Switched to a new branch 'development_fix'

C:\Users\faourn.AUTH\Documents\Development2\cs414-f20-DedicatedRAMs>
```

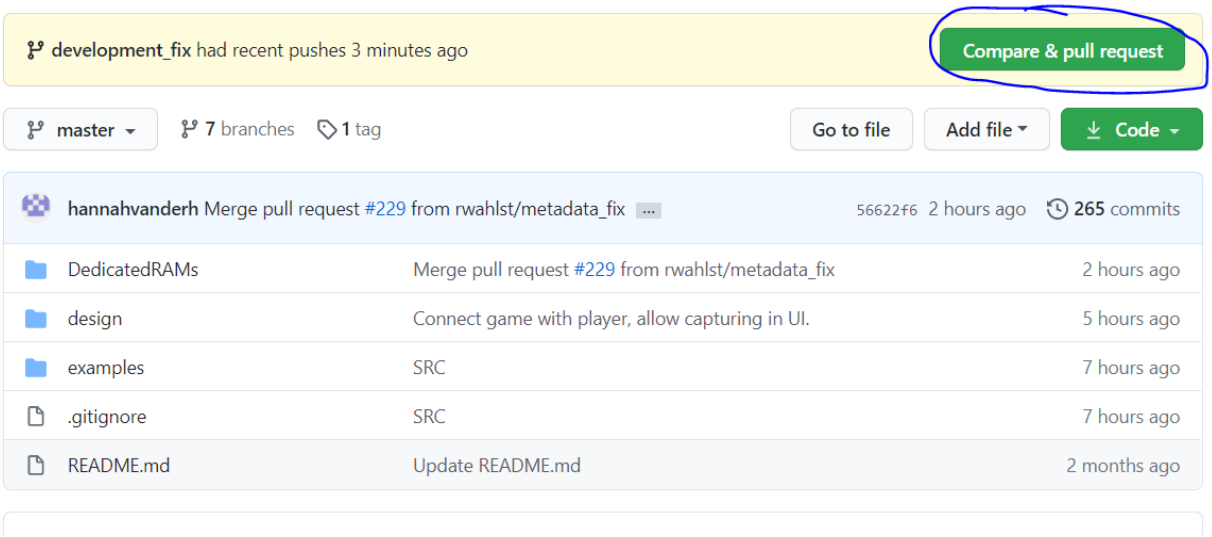
- Change files in Eclipse.
- Add and commit files to git using the following commands with your branch.

```
C:\Users\faourn.AUTH\Documents\Development2\cs414-f20-DedicatedRAMs>git add .

C:\Users\faourn.AUTH\Documents\Development2\cs414-f20-DedicatedRAMs>git commit -m "Comment about commit content"
[development_fix 65f5d53] Comment about commit content
 Committer: Faour <najy.faour@hp.com>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:

    git config --global --edit

After doing this, you may fix the identity used for this commit with:

    git commit --amend --reset-author

 2 files changed, 6 insertions(+), 8 deletions(-)

C:\Users\faourn.AUTH\Documents\Development2\cs414-f20-DedicatedRAMs>
```

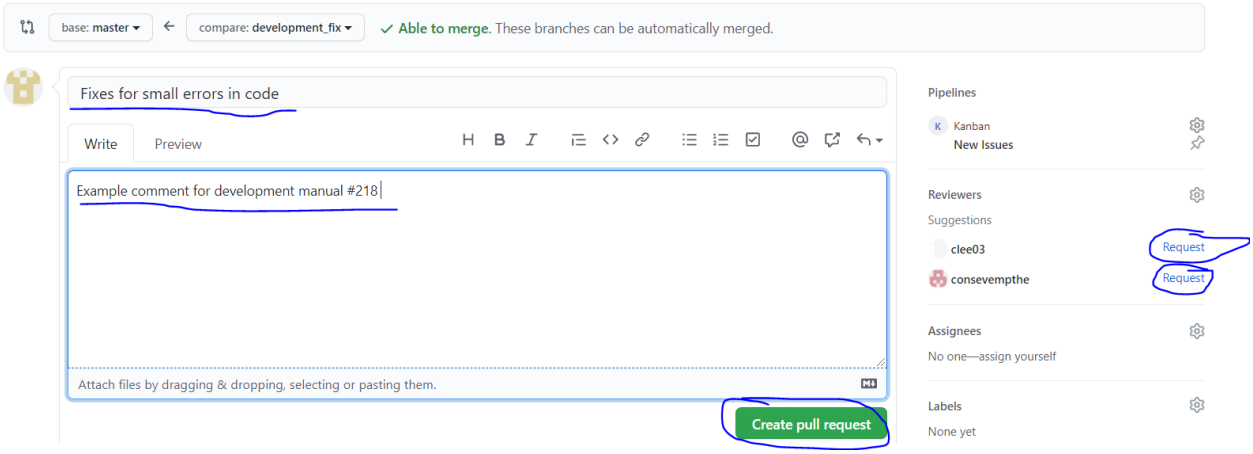
- Once you are ready to send the changes to Github to be merged or reviewed. Push your branch. See below:

```
C:\Users\faourn.AUTH\Documents\Development2\cs414-f20-DedicatedRAMs>git push origin development_fix
Enumerating objects: 15, done.
Counting objects: 100% (15/15), done.
Delta compression using up to 12 threads
Compressing objects: 100% (8/8), done.
Writing objects: 100% (8/8), 680 bytes | 680.00 KiB/s, done.
Total 8 (delta 6), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (6/6), completed with 6 local objects.
remote:
remote: Create a pull request for 'development_fix' on GitHub by visiting:
remote:      https://github.com/rwahlst/cs414-f20-DedicatedRAMs/pull/new/development_fix
remote:
To https://github.com/rwahlst/cs414-f20-DedicatedRAMs.git
 * [new branch]      development_fix -> development_fix

C:\Users\faourn.AUTH\Documents\Development2\cs414-f20-DedicatedRAMs>
```

- Your branch should be on Github now. If you wish to merge the changes, go to Github and create a pull request for your branch. See the below pictures for an example pull request creation. **Do not merge the pull request yourself! Always request reviews and have at least 1 other person review and merge your pull request.**



- Be sure to title your pull request appropriately, leave a nice comment, and request reviews before creating the pull request.



With that, good luck! These should get you started. If you have no experience with Github or git, here are some resources:

https://git-scm.com/docs

https://guides.github.com/activities/hello-world/

https://lab.github.com/

Utilize the internet and team member as **well!**