How to setup & use VSCode with kronosheets
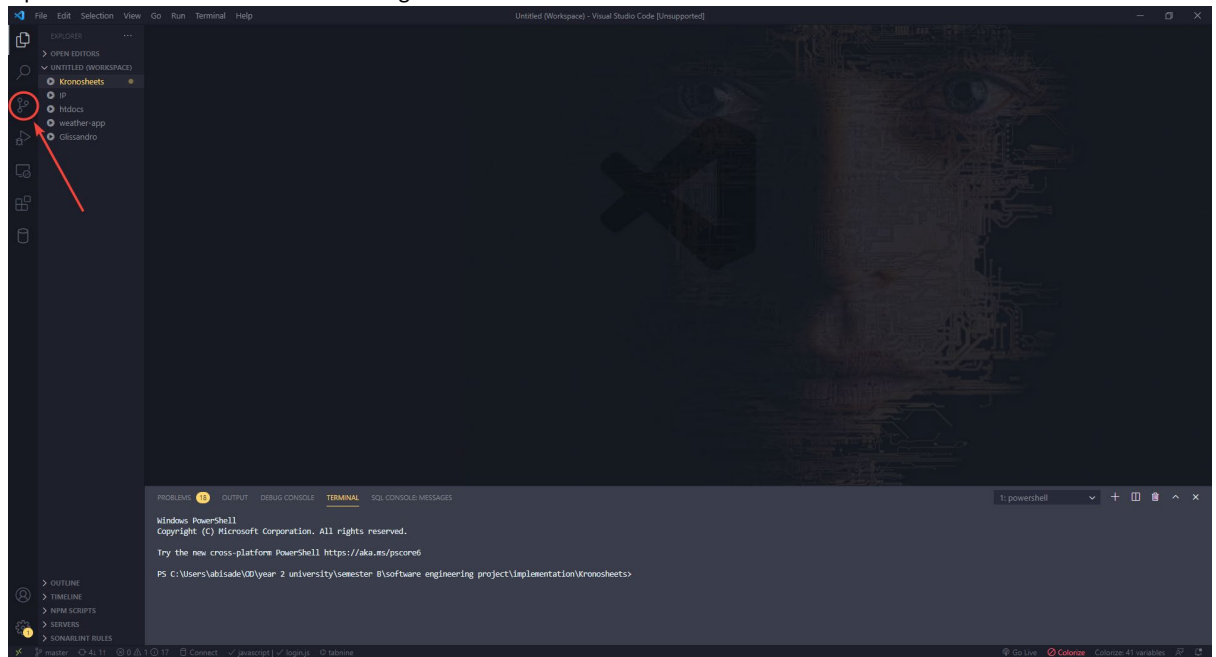
Contents
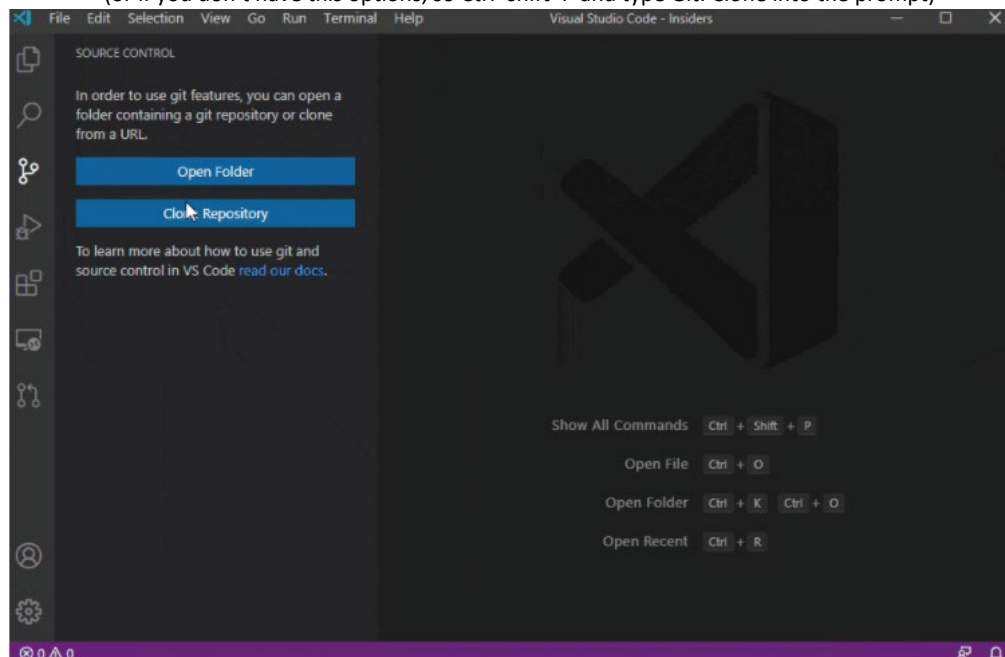
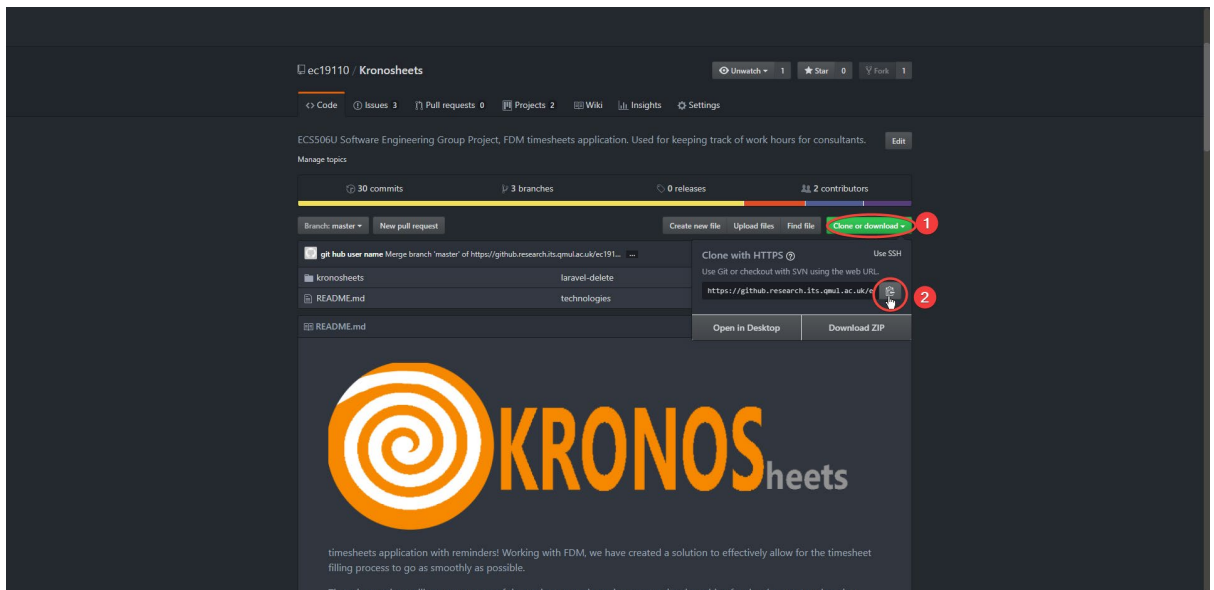## Initial cloning stage

1. Login to the https://github.research.its.qmul.ac.uk/ github and confirm that you can access Kronosheets repo
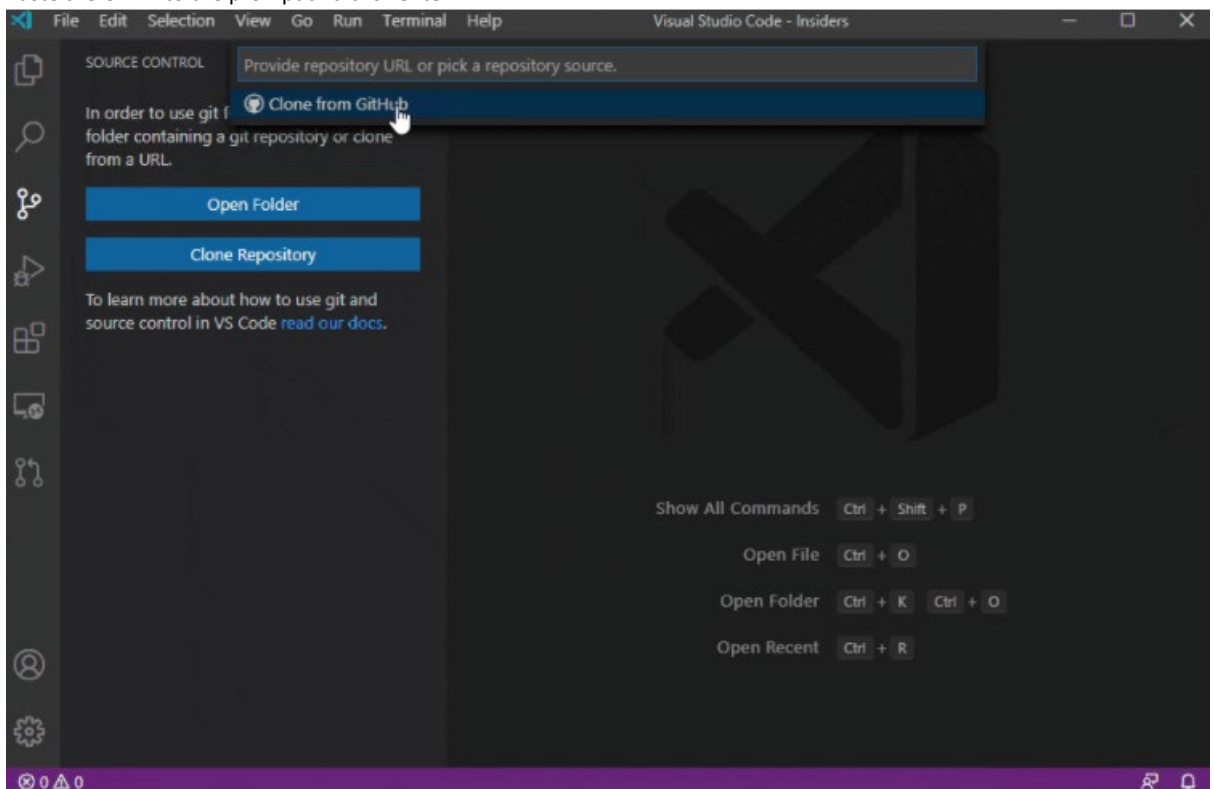2. Open Visual Studio Code and locate the github section

3. Click on "clone repository"
   - (or if you don't have this options, so Ctrl+shift+P and type Git: Clone into the prompt)


4. Copy the clone URL

5. Paste the URL into the prompt and click enter

6. After this, you will be prompted where to save the local files on your pc. Pick a folder you want (remember this for later use in step 9)

7. You will be prompted to log into your QMUL github *try to use the right username/password, or you'll have to take certain steps to reset the incorrect saved password on your system*
   I.   If you have inputted the wrong username/password follow these steps and then try again:
         i.   Click Start
         ii. Type: Credential Manager (On Windows 10, this is under "Start->Settings". Then search for "Credential Manager")
         iii. See the Windows Credentials Manager shortcut and double-click it to open the application.
         iv. Once the app is open, click on the Windows Credentials tab.
         v.   Locate the credentials that you want to remove/update, they will start with "git:" and might begin with "ada:"
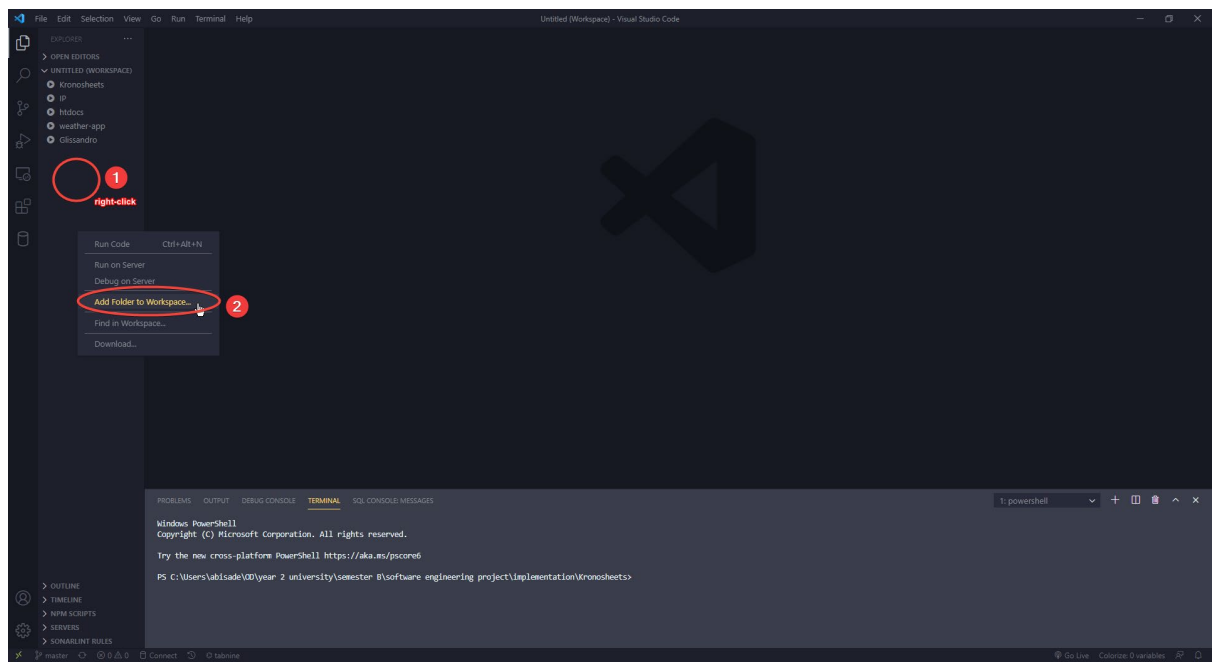
vi.   Click on the credential entry, it will open a details view of the entry.
vii. Click Edit or Remove as required and confirm.
viii. Wash, rinse, repeat as necessary.
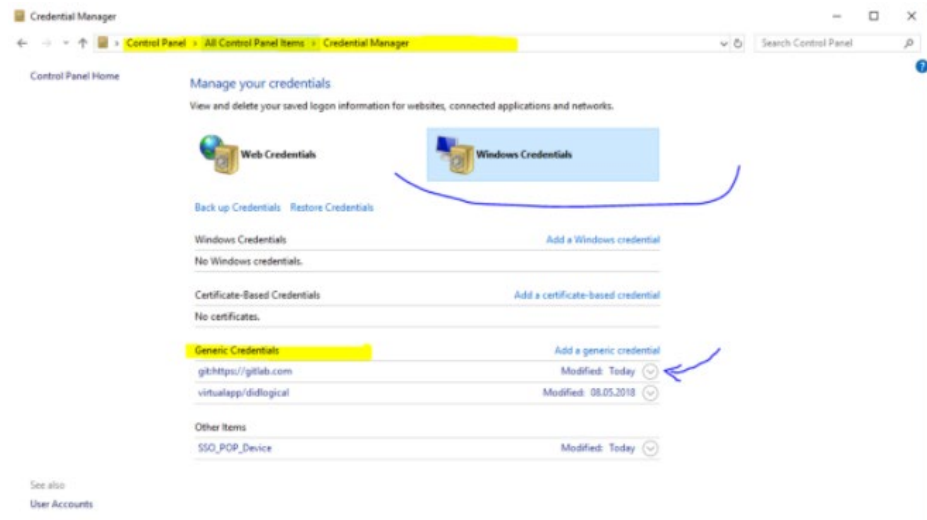
8. After this, go back to the explorer page in VSCode
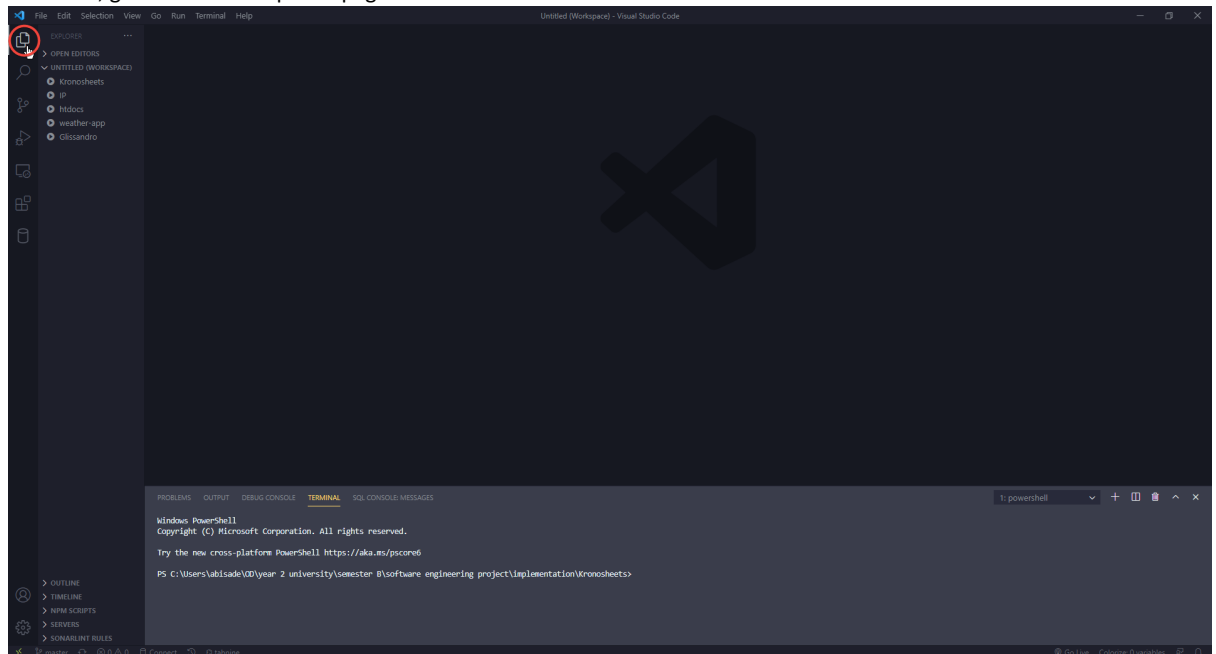
9. Right-click on your workspace tab and click add "folder to workspace" and browse to the cloned location
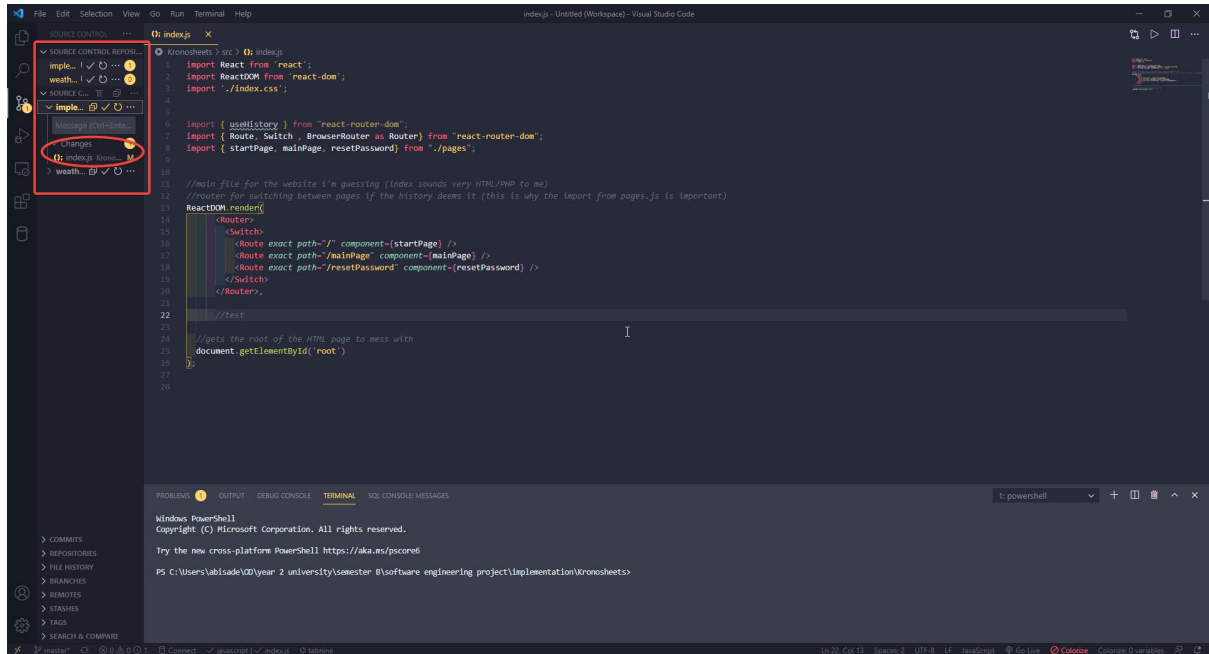
10. Profit

Commit-Pushing and pulling from the github

Disclaimer: if you are behind on the git, you will have to pull first before you can push (or you could force push alternatively)

After making a change, in the git tab there will automatically be a change appearing. These are the changes that we would like to push to the github.
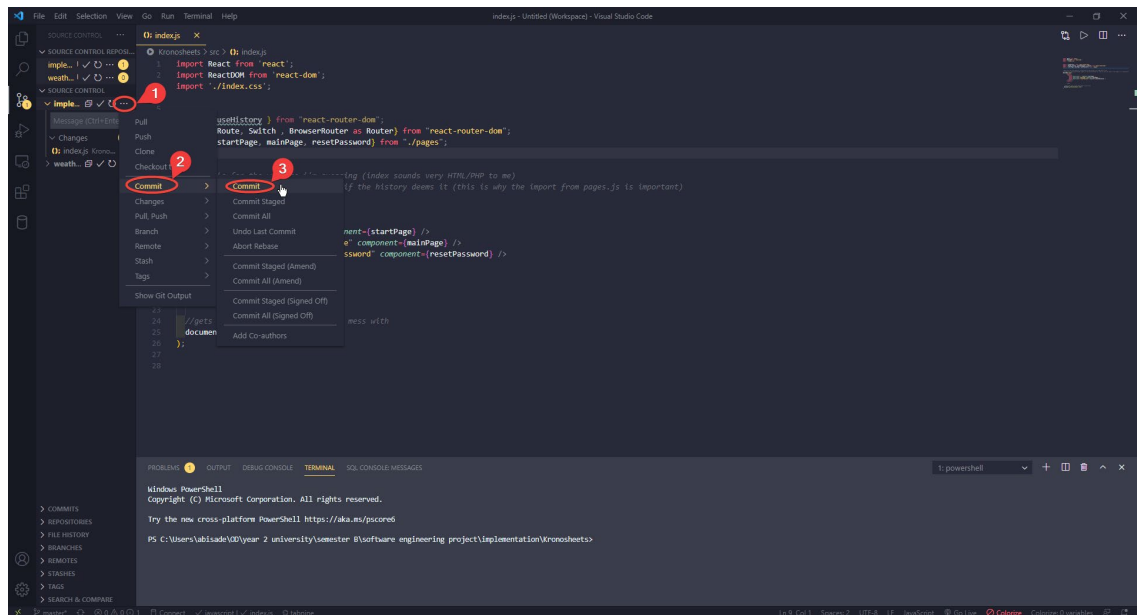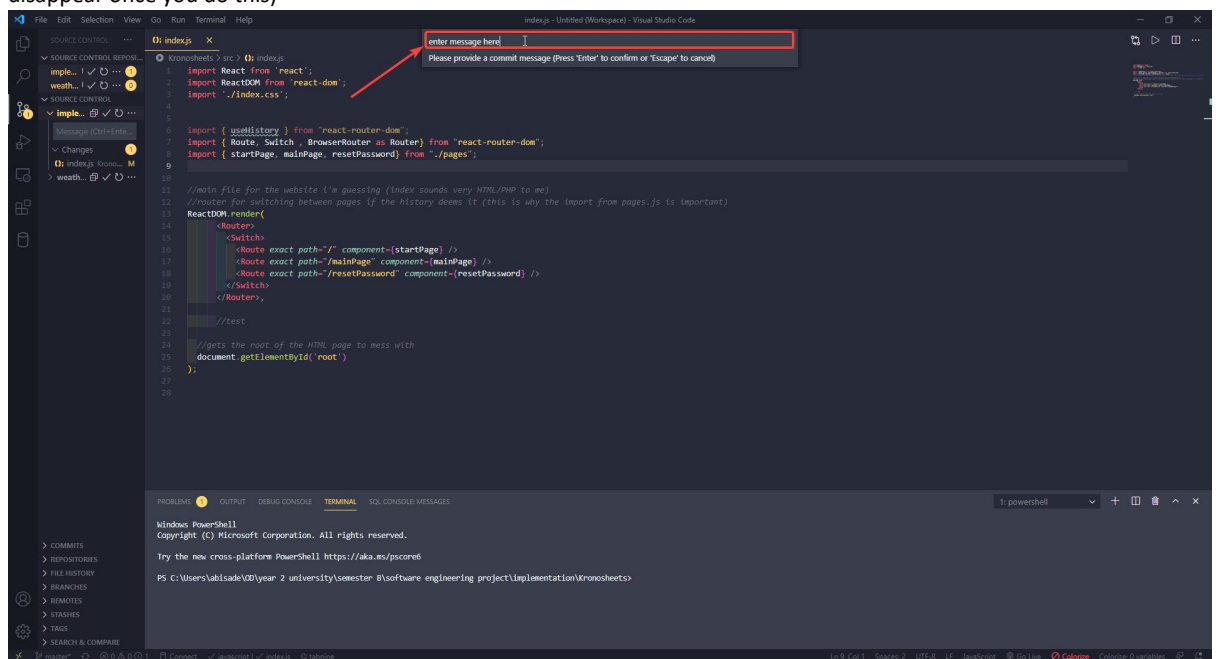

To do so, follow these steps:

1. Click the commit button (under the correct git sources i.e. Kronosheets)
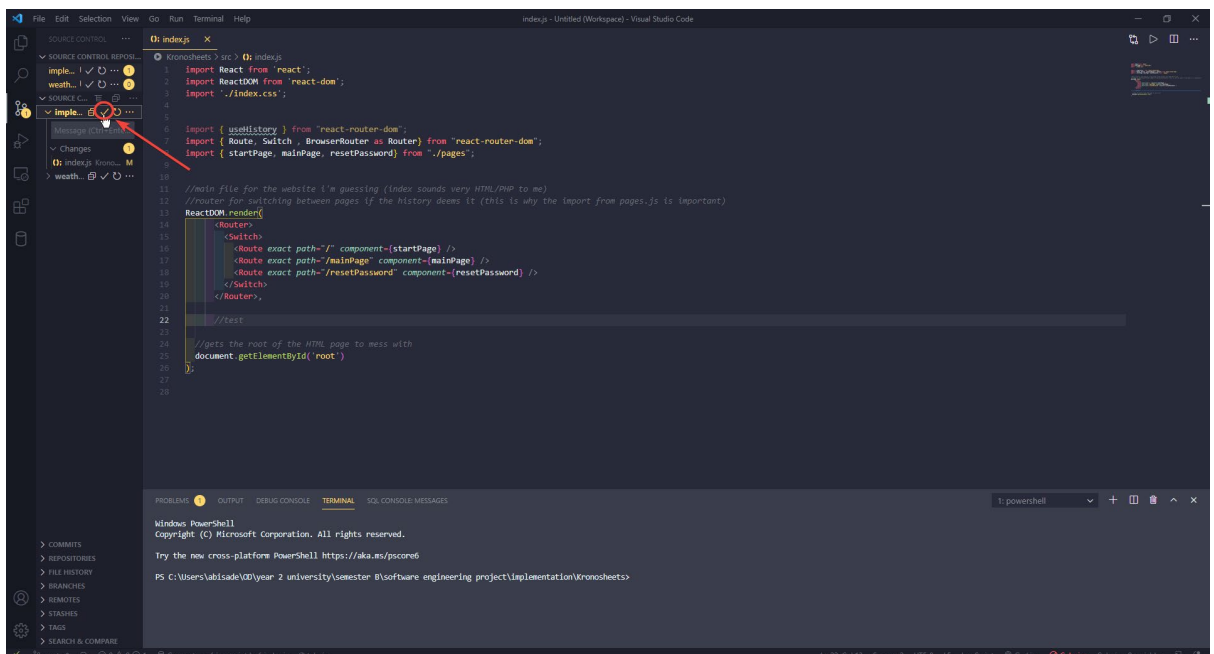
    a. Alternatively, you can use the … option to manually select the type of commit you want:
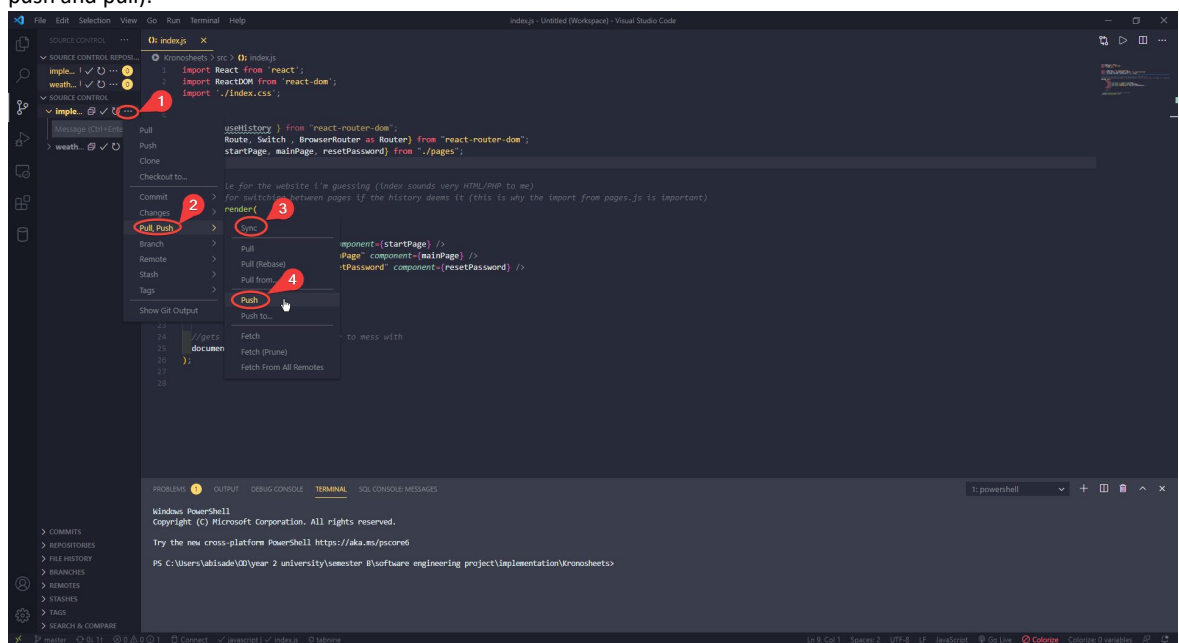
2. After selecting commit, enter the commit message you want associated and click enter (the changes should disappear once you do this)

3. Now we can push the changes. You can do this by going down to the push/pull option
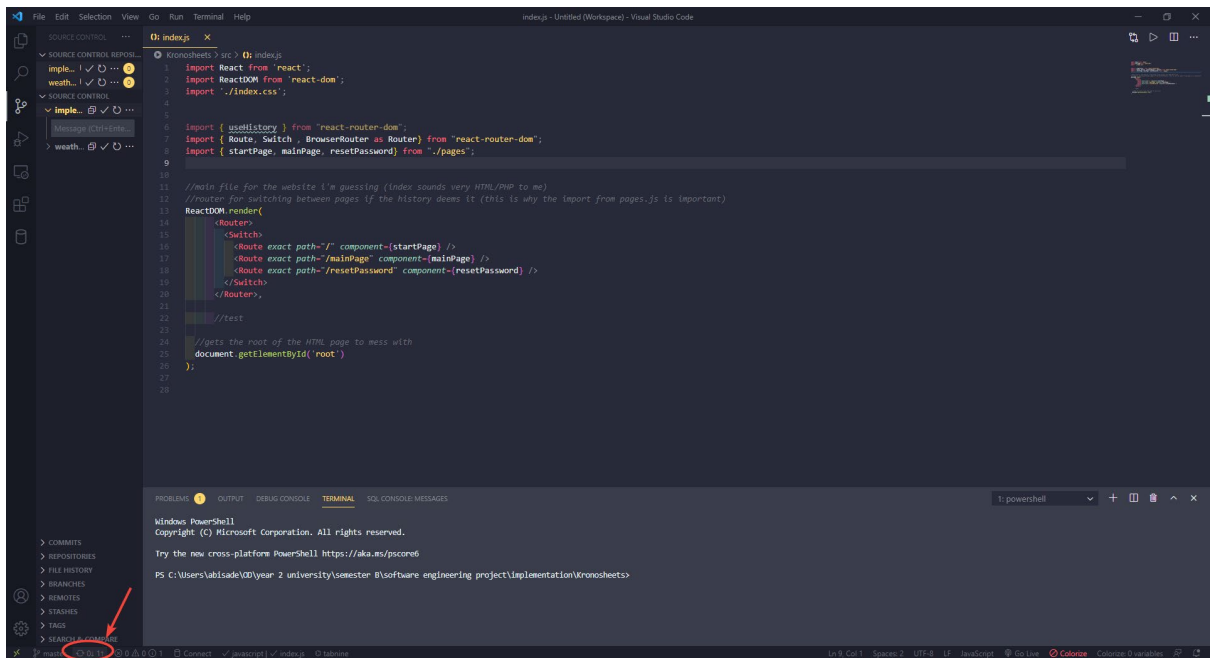
a. Alternatively, you can use the … option to manually select the type of push you want (or sync [3] to push and pull):

Setting up XAMPP

To verify the connection to the back-end, I utilise the XAMPP application to make a local PHP server. Follow the steps to set up the environment for communication with the backend.

1. https://www.apachefriends.org/index.html
   download XAMPP from the link above for your specific OS
2. Follow the set-up wizard for the XAMP application and run it once it has finished
3. Launch XAMPP
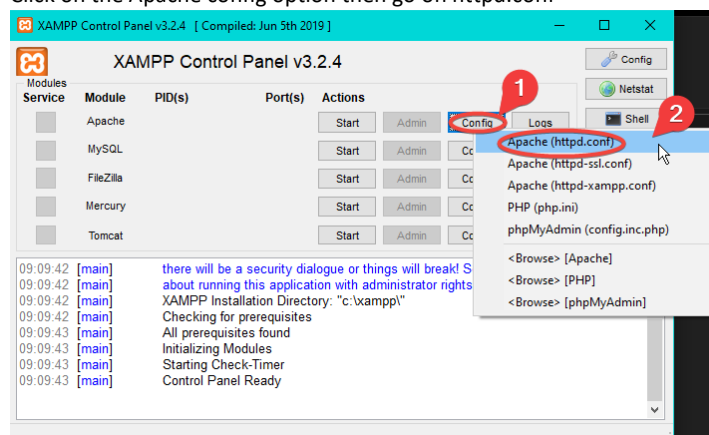4. These are the main modules we will be using
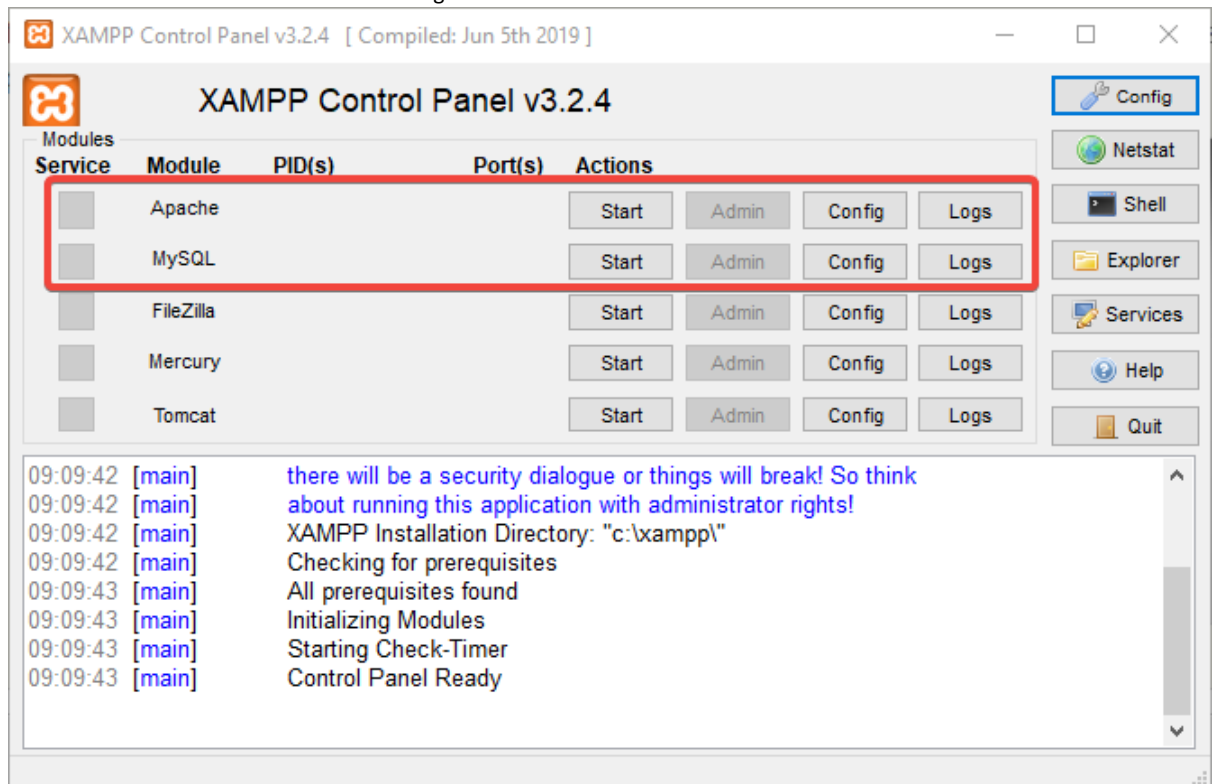
Apache is the PHP server itself, and mySQL is the database
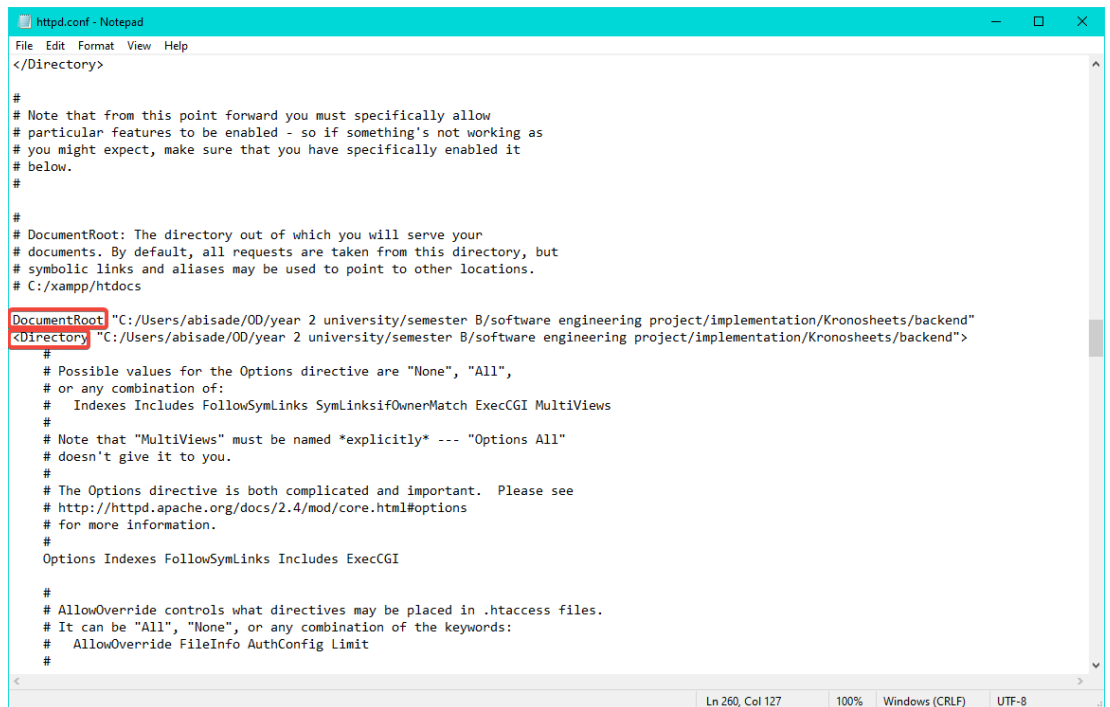When we start the Apache server, the PHP that runs is in the location
C:\xampp\htdocs
Or wherever you saved XAMPP to during installation.

5. (optional) if you want to use another directory to launch your PHP files (not htdocs)
   a. Click on the Apache config option then go on httpd.conf

   b. Scroll down or control F to find the [DocumentRoot] and <Directory> options in the text file that opens up

c. Change both directories to the absolute paths to the file location you want to run PHP file from

d. Save the text file and when you run Apache, it should run from that location (this will not work if the path is invalid)

6. Now when we start the server we can go to localhost (it should auto open the web browser and go to localhost:3000)

7. You can now set up a database at http://localhost/phpmyadmin/server_databases.php?server=1 (make sure XAMPP is running)

8. If you don't know how to use the UI, just paste SQL statements into the SQL tab to get your DB running
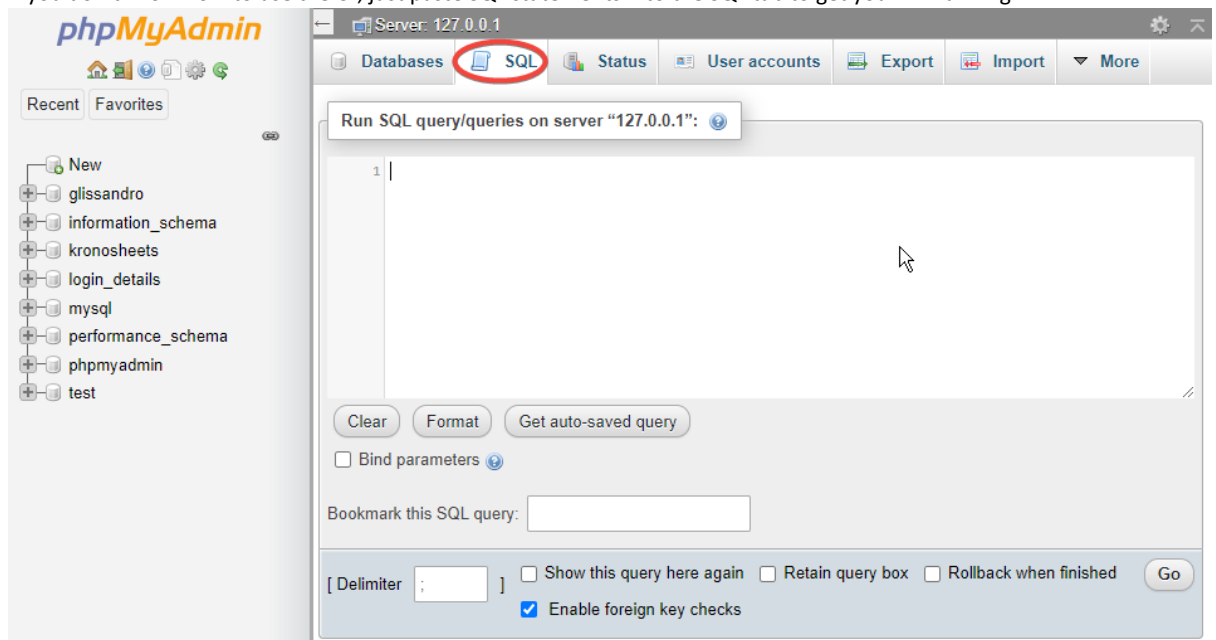
Using the database with VSCode

I use the visual studio code extension "SQLTools" to create interact with the database in an integrated environment, you can install and use it by doing the following:

1. Select the extensions tab on the side panel of VSCode
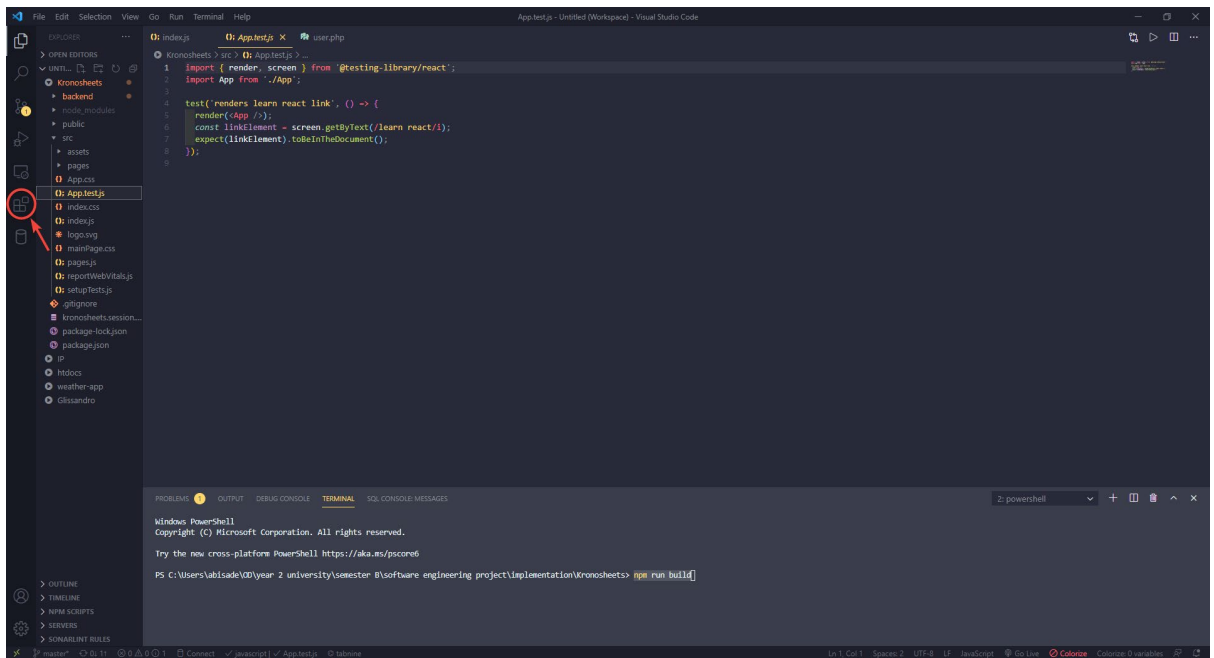
2. Search "sqlTools" and install the extension
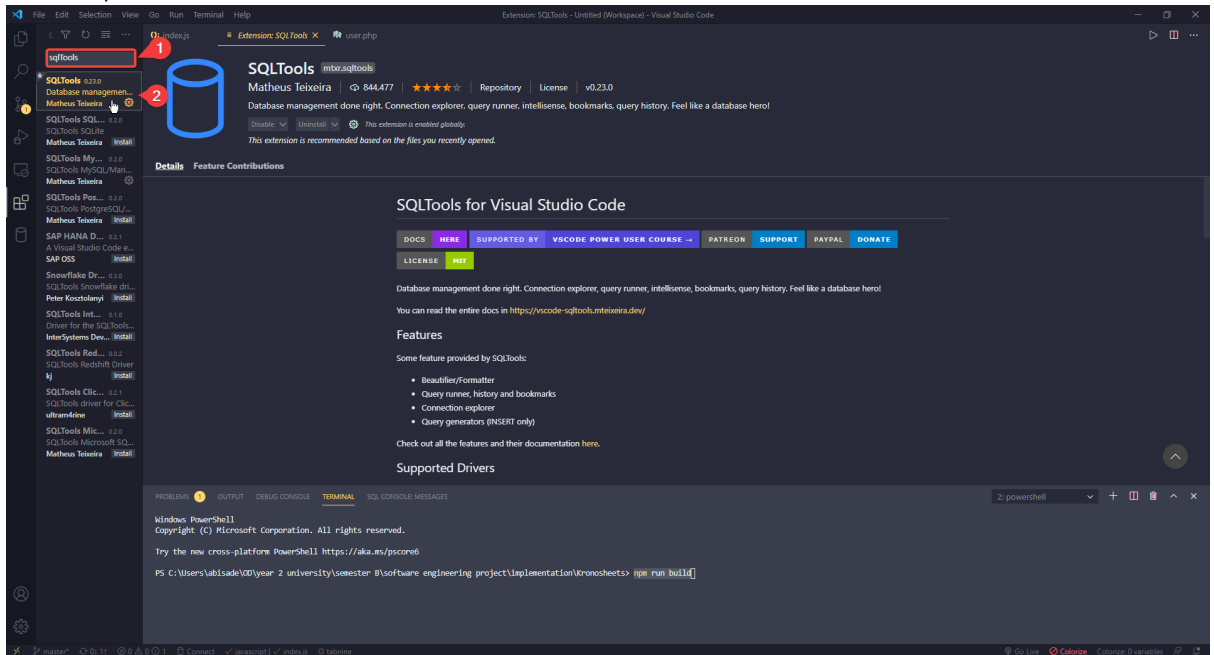
3. After the installation, there should be another panel appearing on your side panel. Click on that
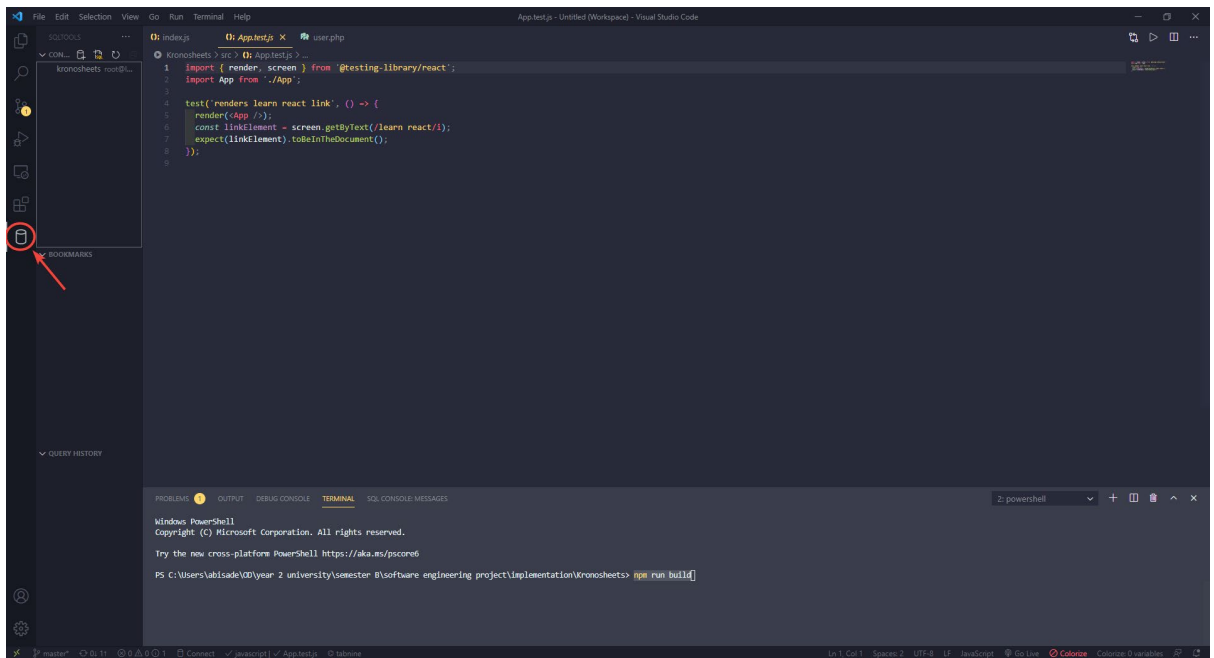
4. When on the extension options, go to "Add New Connection"
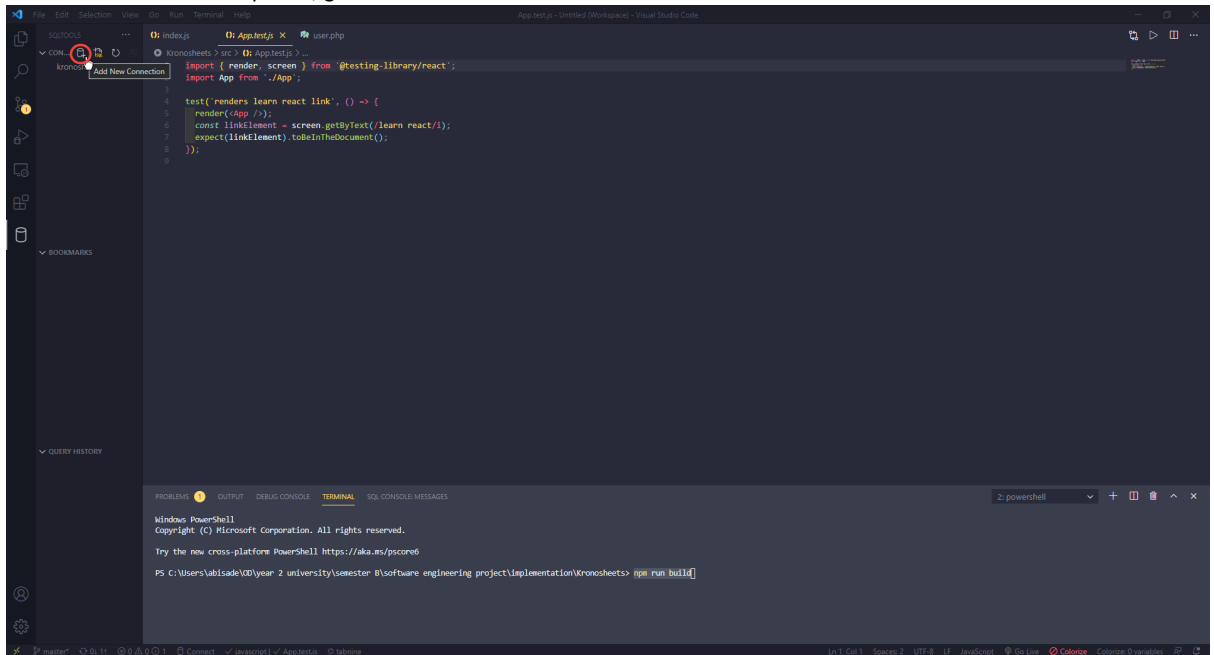
5. You will be greeted with a screen with less options than the following
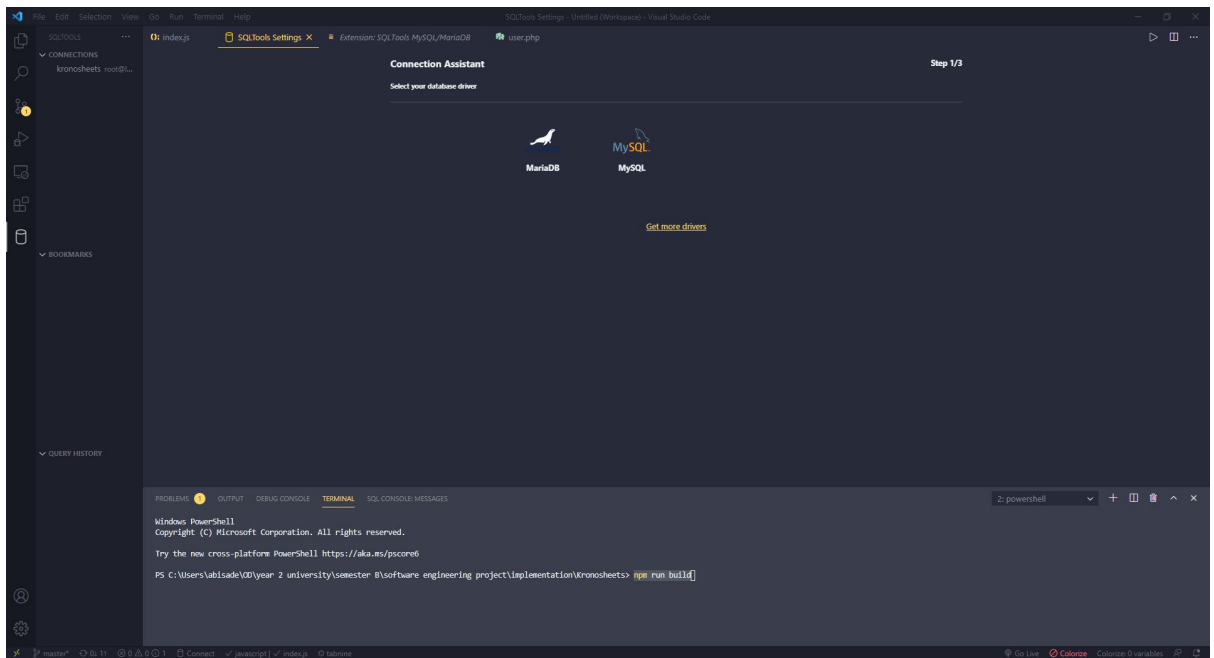
This is because we need to select the "get more drivers" option
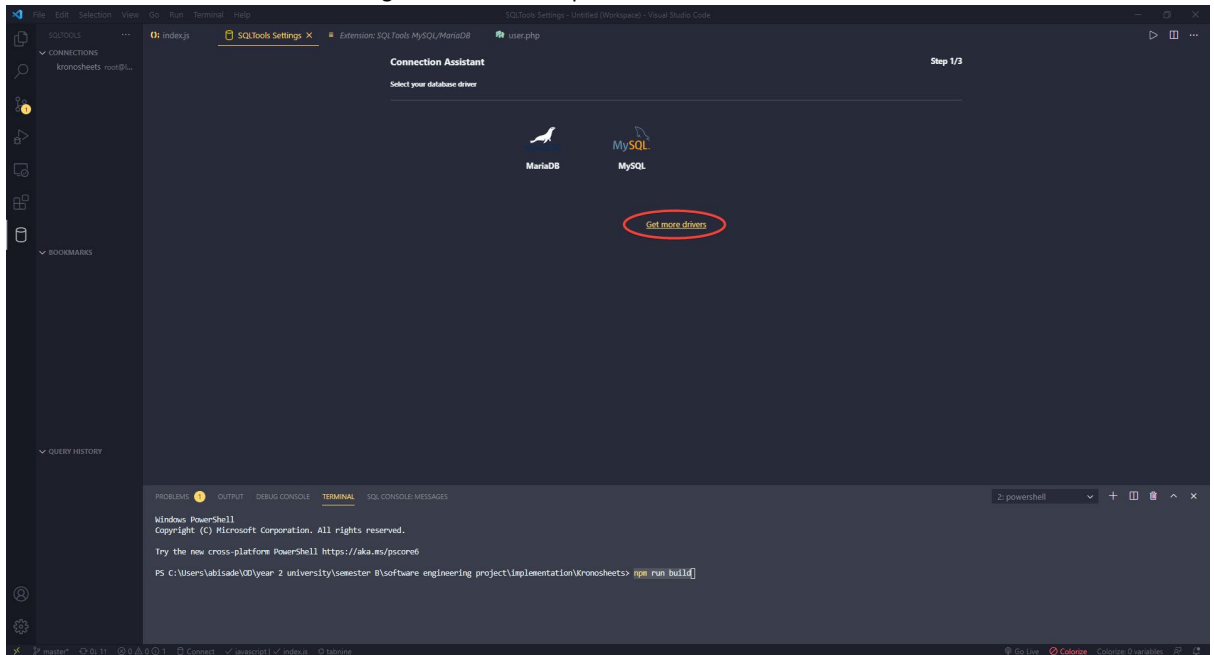
6. This is because we do not have the drivers necessary to connect to mySQL databases yet. The "get more drivers" option should bring you back to the extensions page. Select the mySQL/MariaDB extension and install it
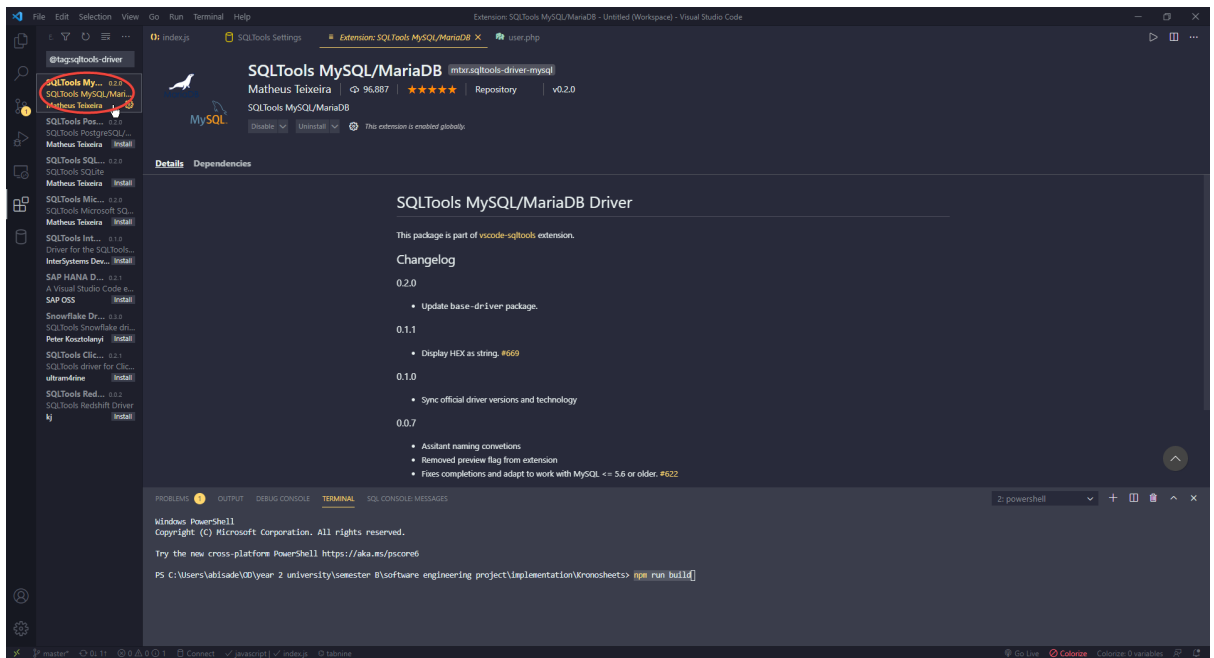
7. Go back to the SQLTools tab and you should see the new mySQL option now, click on that
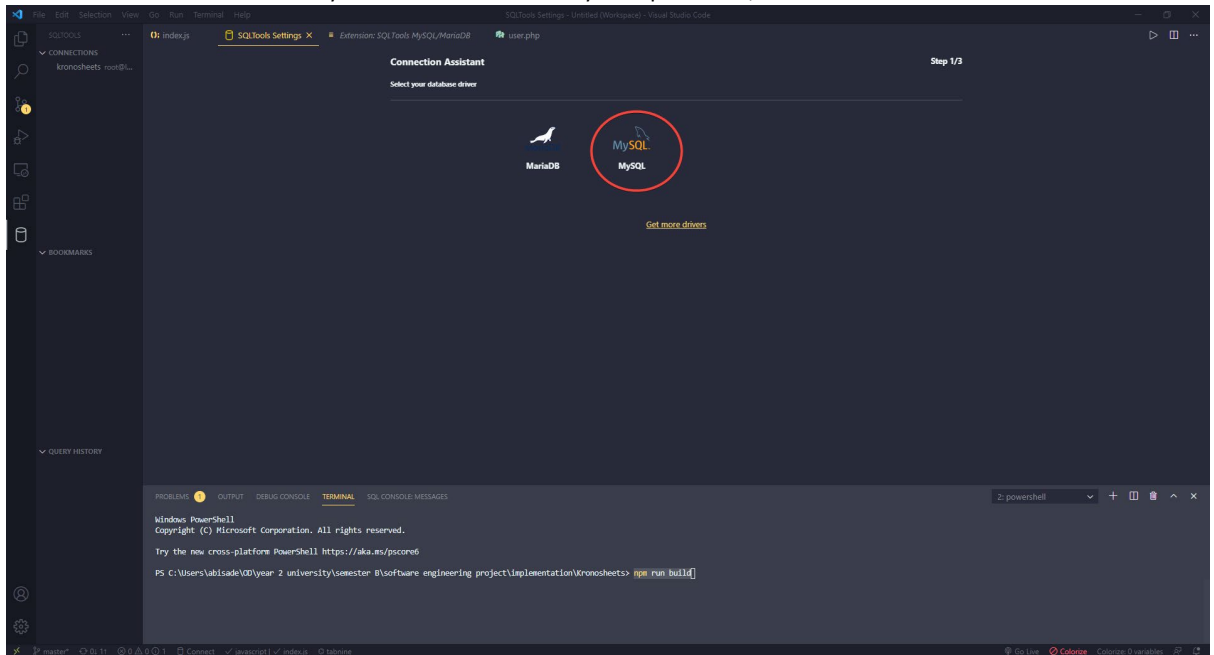
8. Fill out the connection settings with the following (I believe these are the defaults for an XAMPP database)
   Connection name: [any name you want]
   Connection group: [leave blank]
   Connect using: Server and Port
   Server Address: localhost
   Port: 3306
   Database: [the name of the database you created]
   Username: root
   Use password: Use empty password
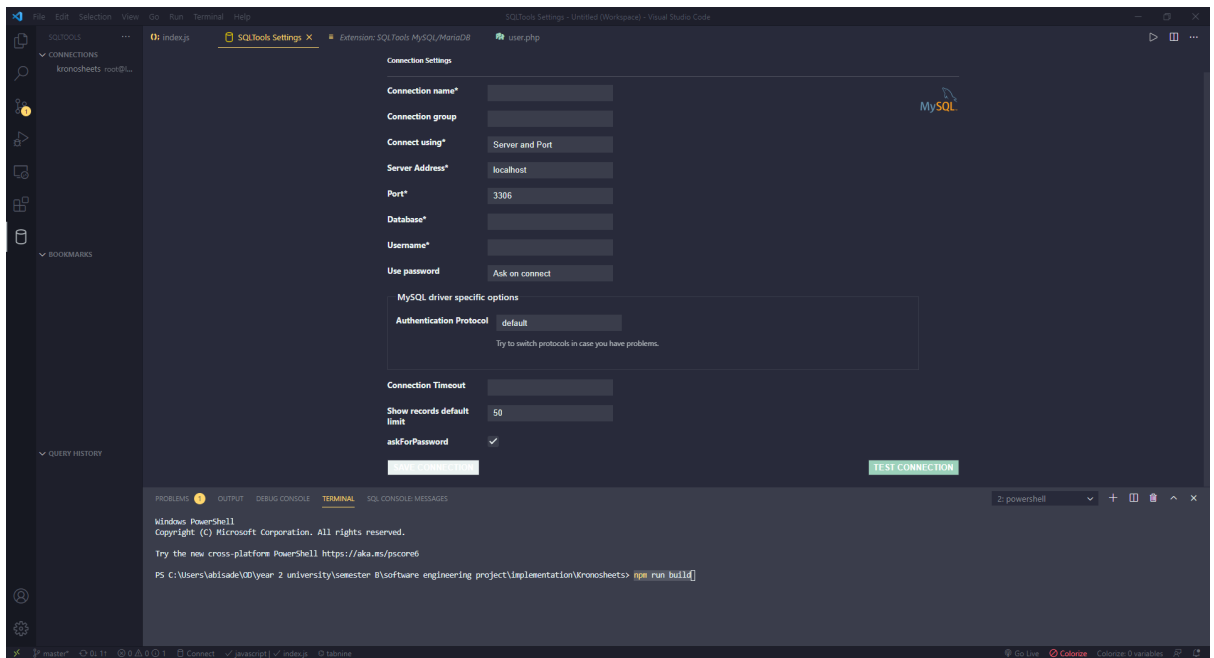
9. Now we can click test connection. It should connect, if not try fiddling around with the settings (you may have the database name or user/password wrong
    a. If you are truly stuck, you can go to C:\xampp\phpMyAdmin\config.inc.php
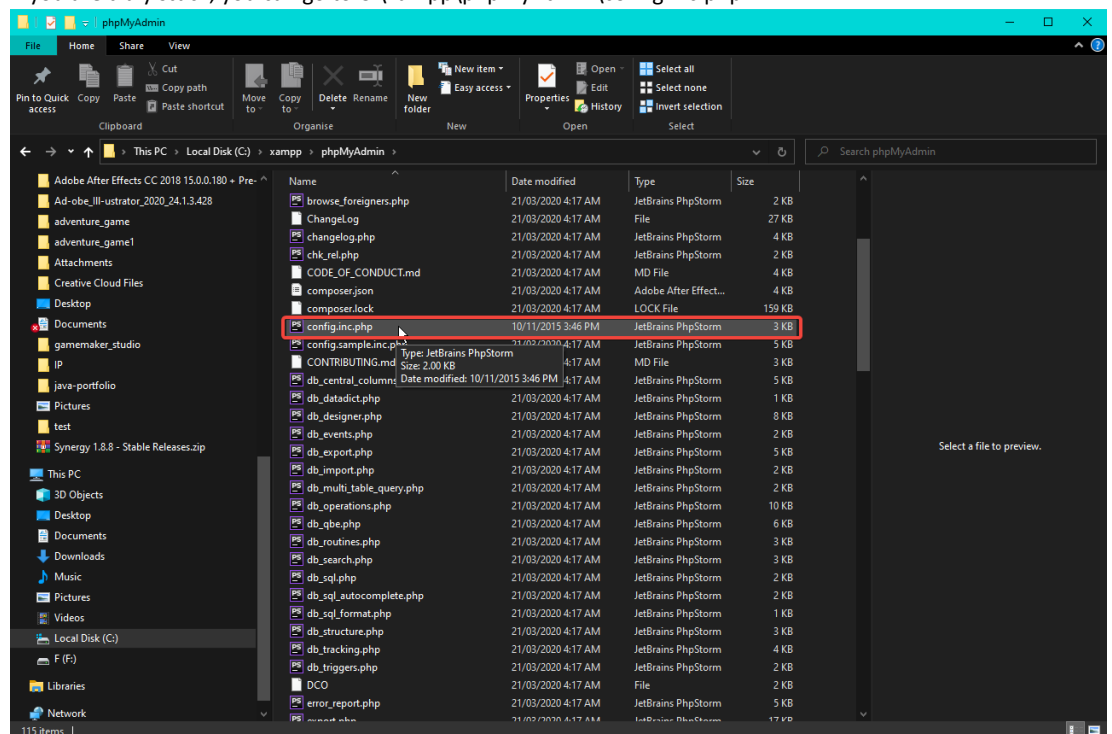
    b. After opening this file you will see something like:
       $cfg['Servers'][$i]['user'] = 'YOUR USER NAME IS HERE';
       $cfg['Servers'][$i]['password'] = 'AND YOU PASSWORD IS HERE';
       This should be your username/password
10. then save the connection
11. now you can view your database from VSCode
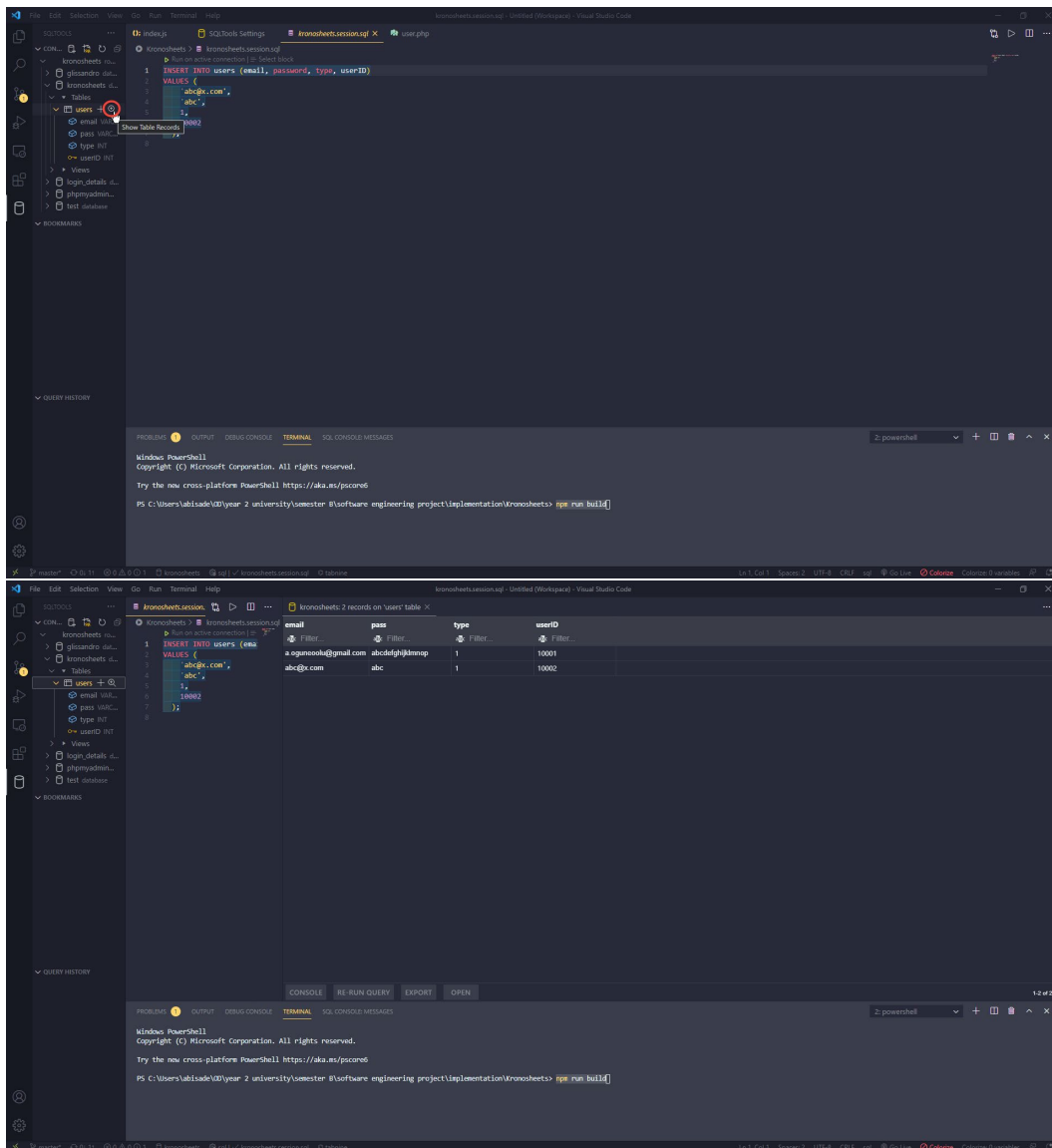
and edit it with SQL commands!
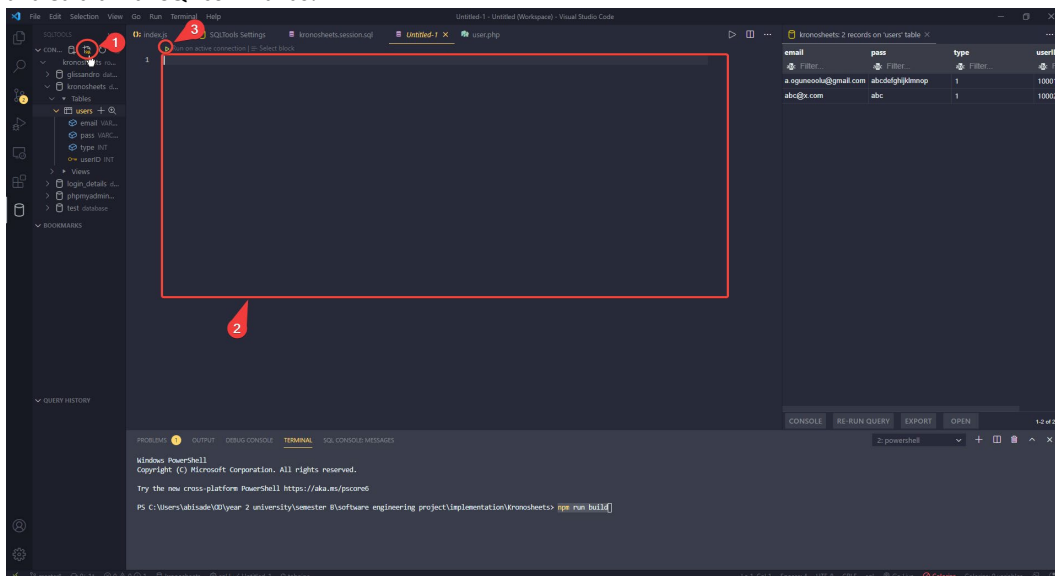
Running the react-app

This is where I am not totally sure.

I assume you only need to do the following:

Initial run steps:

1. Right click on the folder where the react app is and click open in integrated terminal
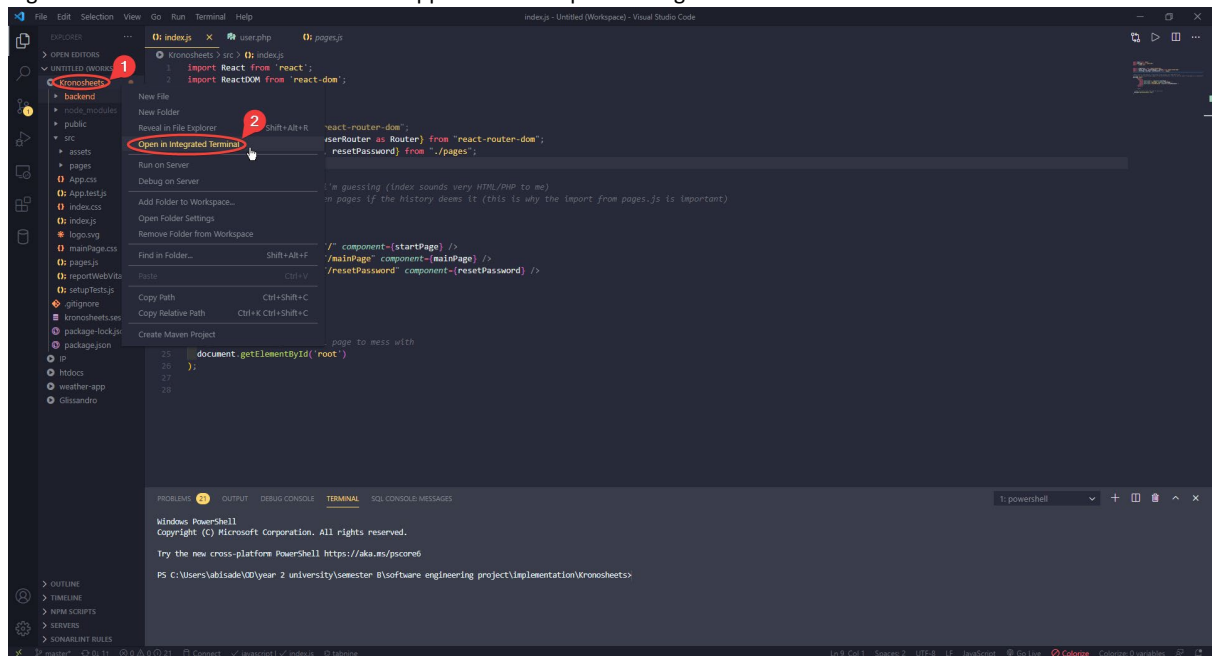
2. Type "npm install" to install all dependencies
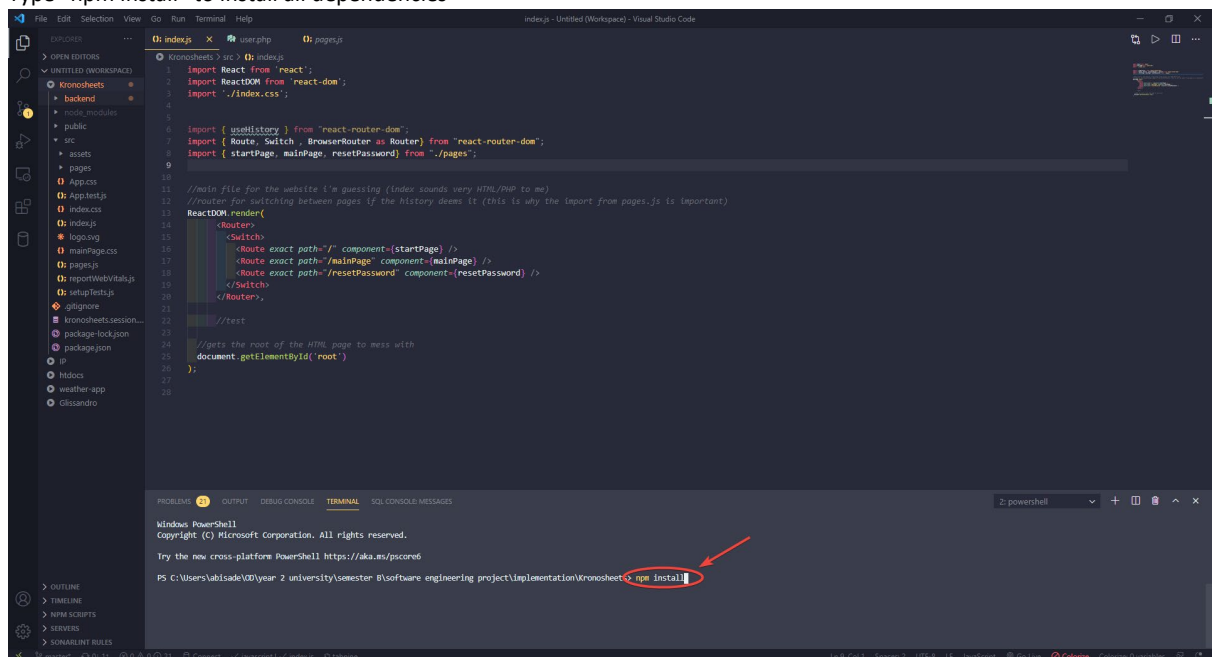
3. (optional? I think????) build the application with "npm run build" in the same terminal

Repeated steps:

1. Start the XAMPP server (for the backend PHP)

Before start:
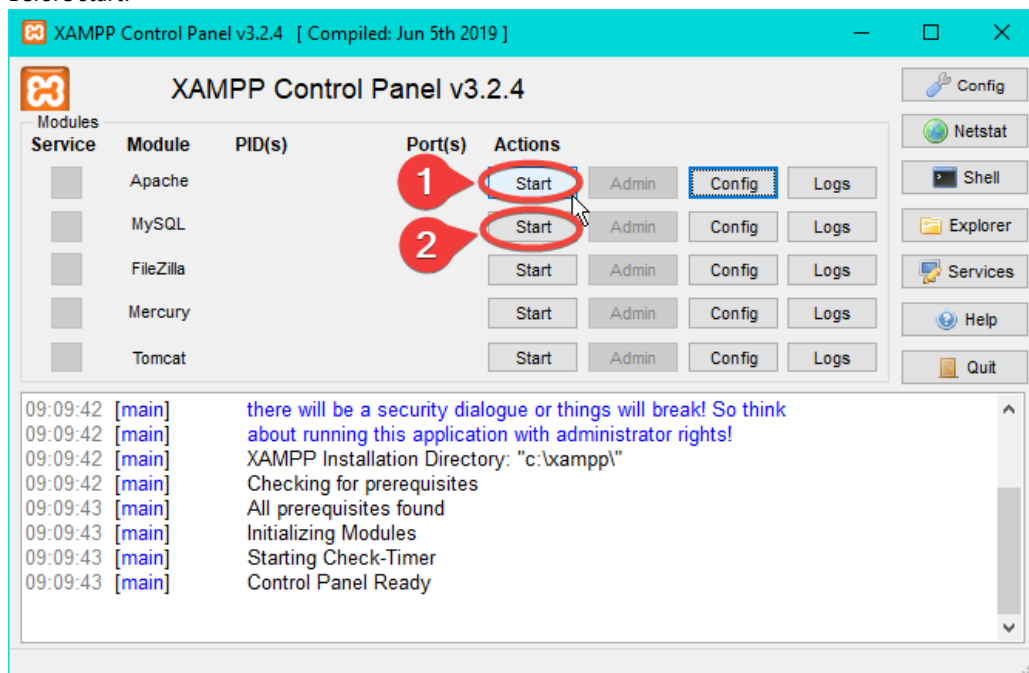
After start:
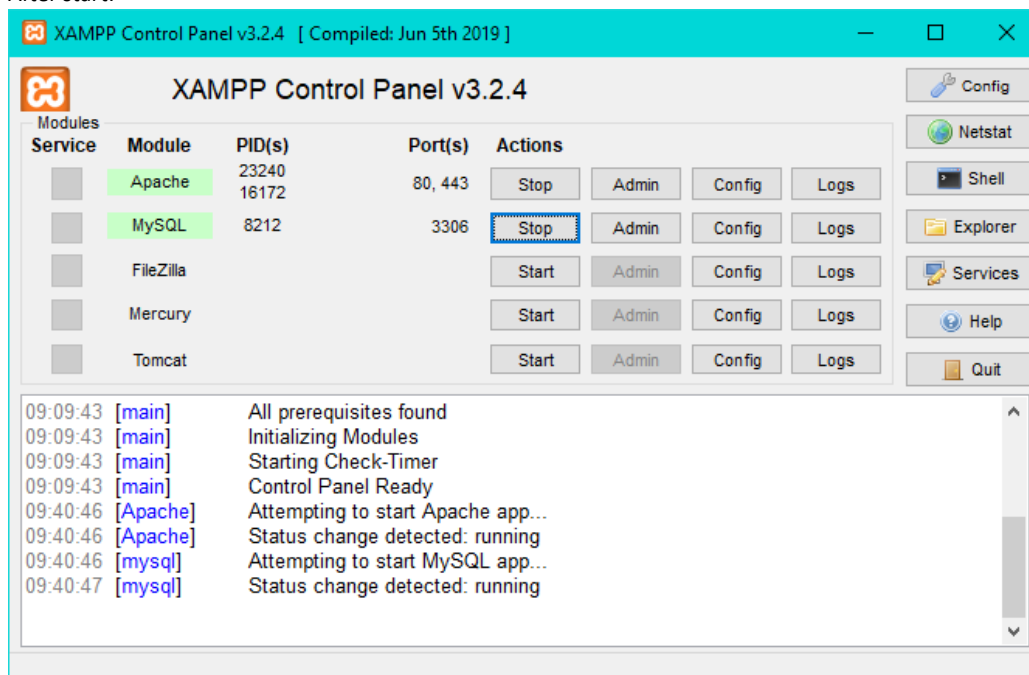
4.

Switching branches

I have create branches for front-end and backend specific work (that may not be final), if you want to switch to these branches instead of working directly on master (which I don't think is great), you can do the following:

1. Go down to the github branch tab
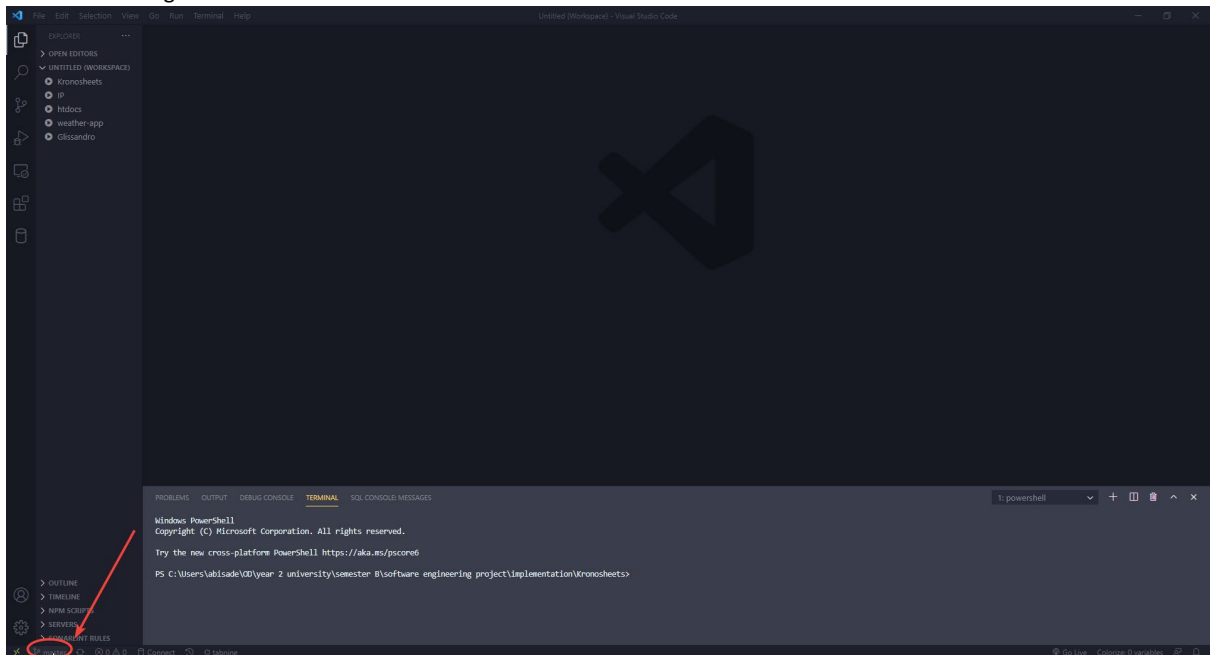
2. Select the branch you want to work on
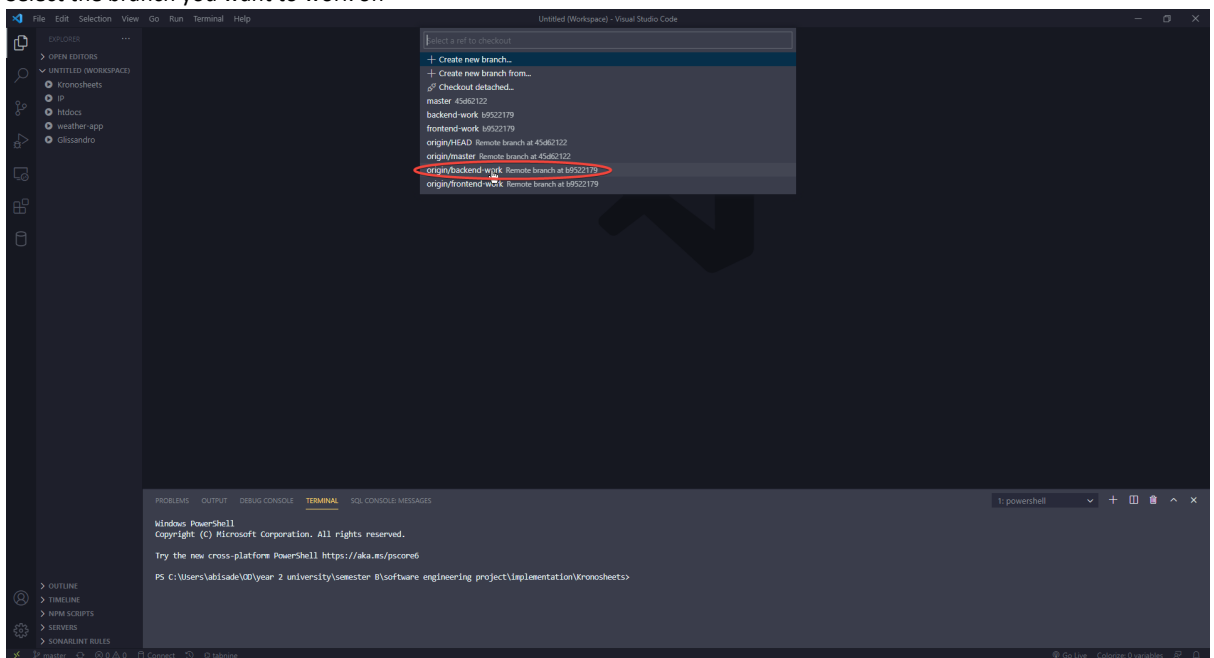
VSCode extensions

The following is the list of VSCode extensions I use (some of them are kind of redundant tho)

```
code --install-extension aaron-bond.better-comments
code --install-extension abusaidm.html-snippets
code --install-extension bmewburn.vscode-intelephense-client
code --install-extension christian-kohler.path-intellisense
code --install-extension CoenraadS.bracket-pair-colorizer-2
code --install-extension dongli.python-preview
code --install-extension eamodio.gitlens
code --install-extension felixfbecker.php-intellisense
code --install-extension formulahendry.auto-close-tag
code --install-extension formulahendry.auto-rename-tag
code --install-extension formulahendry.code-runner
code --install-extension k--kato.intellij-idea-keybindings
code --install-extension kamikillerto.vscode-colorize
code --install-extension monokai.theme-monokai-pro-vscode
code --install-extension ms-python.python
code --install-extension ms-python.vscode-pylance
code --install-extension ms-toolsai.jupyter
code --install-extension ms-vscode-remote.remote-wsl
code --install-extension msjsdiag.vscode-react-native
code --install-extension mtxr.sqltools
code --install-extension mtxr.sqltools-driver-mysql
code --install-extension oderwat.indent-rainbow
code --install-extension PKief.material-icon-theme
code --install-extension pranaygp.vscode-css-peek
code --install-extension redhat.java
code --install-extension redhat.vscode-commons
code --install-extension redhat.vscode-community-server-connector
code --install-extension redhat.vscode-rsp-ui
code --install-extension ritwickdey.LiveServer
code --install-extension shalldie.background
code --install-extension SonarSource.sonarlint-vscode
code --install-extension streetsidesoftware.code-spell-checker
code --install-extension TabNine.tabnine-vscode
code --install-extension VisualStudioExptTeam.vscodeintellicode
code --install-extension vscjava.vscode-java-debug
code --install-extension vscjava.vscode-java-dependency
code --install-extension vscjava.vscode-java-pack
code --install-extension vscjava.vscode-java-test
code --install-extension vscjava.vscode-maven
code --install-extension vscodevim.vim
code --install-extension wayou.vscode-todo-highlight
code --install-extension zhuangtongfa.material-theme
```

to install, just paste into the integrated terminal of VSCode and press enter.