

Howdy!

- [1. Download github and install.](#)
- [2. If you do not have an account please register for a github account.](#)
- [3. Clone](#)
- [4 Download intellij \(latest version\):](#)
- [5. IMPORT PROJECT](#)
- [6. Download google app engine \(python\)](#)
- [7. IMPORT APPLICATION into Google App Engine](#)
- [8. RUNNING google app engine](#)
- [9. Download and run mongodb](#)
- [10. Download maven](#)
 - [\[WINDOWS ONLY\]](#)
 - [\[NON WINDOWS\]](#)
- [11. Download protobuf](#)
- [12. Compiling servers](#)
- [13: RUNNING COURSE SKETCH](#)
 - [App Engine](#)
 - [Mongodb](#)
 - [Servers](#)
 - [\(Automatically\)](#)
 - [\(Manually\)](#)
 - [Proxy](#)
 - [Submission](#)
 - [Answer checker](#)
 - [Login Server](#)
 - [Database Server](#)
- [14: FINAL SETUP](#)
- [15. Additional Setup \(Optional but recommended\)](#)
 - [Import settings](#)
 - [How to Install checkstyle for IntelliJ](#)
 - [Install Multirun](#)
 - [Chrome workspace](#)
 - [Fixing protobuf errors](#)
 - [Viewing Mongo Database](#)
 - [How to set up fake data](#)

1. Download github and install.

go to github.com

2. If you do not have an account please register for a github account.

Send your github account name to hammond@tamu.edu to get access to the repository.

3. Clone

Once you have access to the repository, open the github application on your computer, click the coursesketch repository and click “clone to computer”. **Make sure to note the location that it is installed.**

DO NOT FORK

if you have a mac then you will have an easy life using github

if you have windows and you are struggling using github then you can download other git clients or if you are advance you just the command line. (something that will probably happen at least once because the windows interface sometimes breaks)

4 Download intillij (latest version):

and install it (this is done by moving it the location you want to put it and open it)

If it does not open chances are that the sdk is not installed.

google “java sdk 7” and follow the instructions to download it and install it.

When you open intillij for the first time it should look like this

If it says that you need legacy java 6 se runtime on a mac use this

<http://thegothicparty.com/dev/article/how-to-install-legacy-java-se-6-runtime-for-mac-10-10/>



IntelliJ IDEA

Version 14.0.2

Create New Project

Import Project

Open

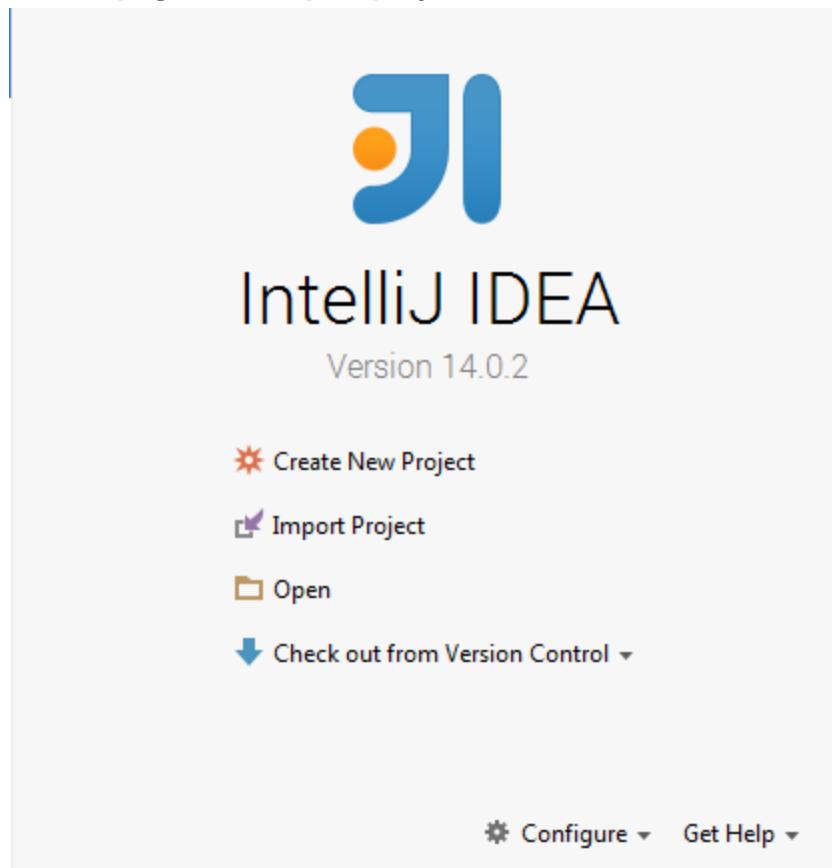
Check out from Version Control ▾

Configure ▾ Get Help ▾

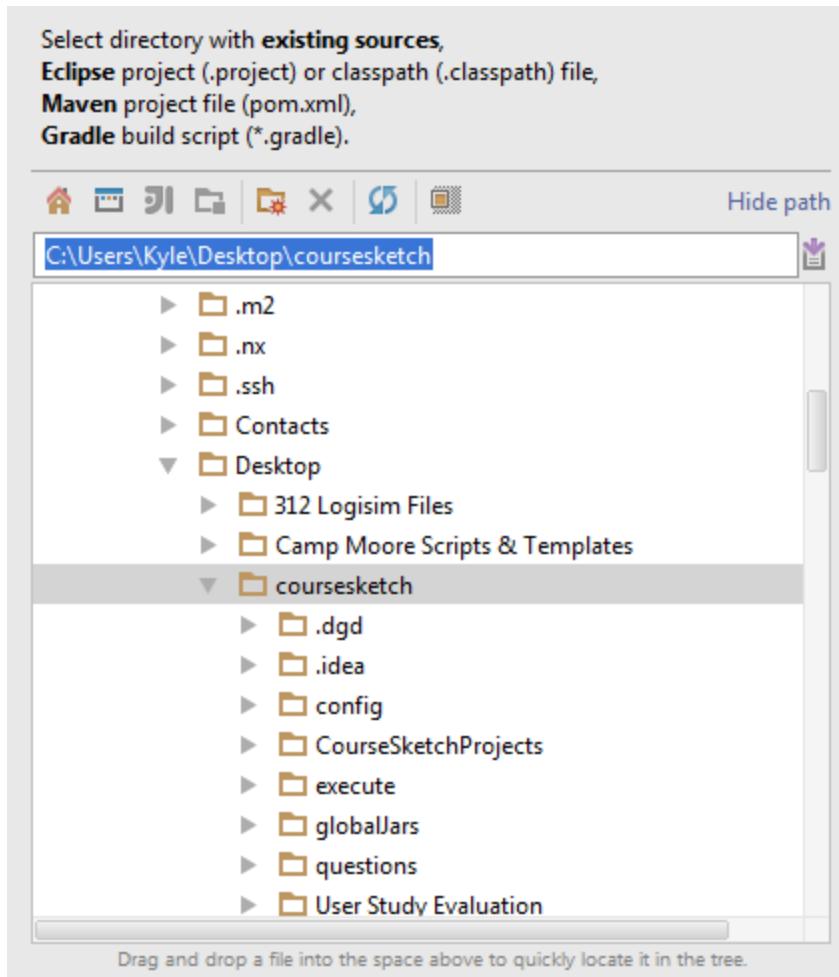
From this page we import the project

5. IMPORT PROJECT

On this page click import project

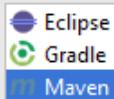


It should bring you to a file viewer. Find the coursesketch folder select it and press next

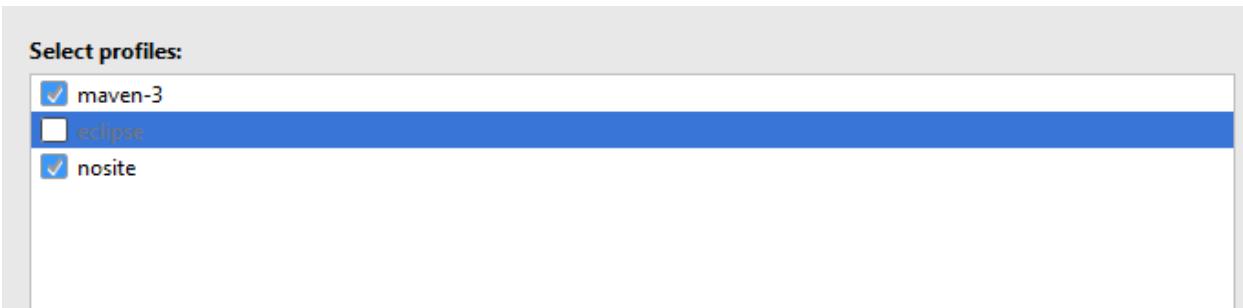


Here you select import project from external model and select maven.

- Create project from existing sources
- Import project from external model

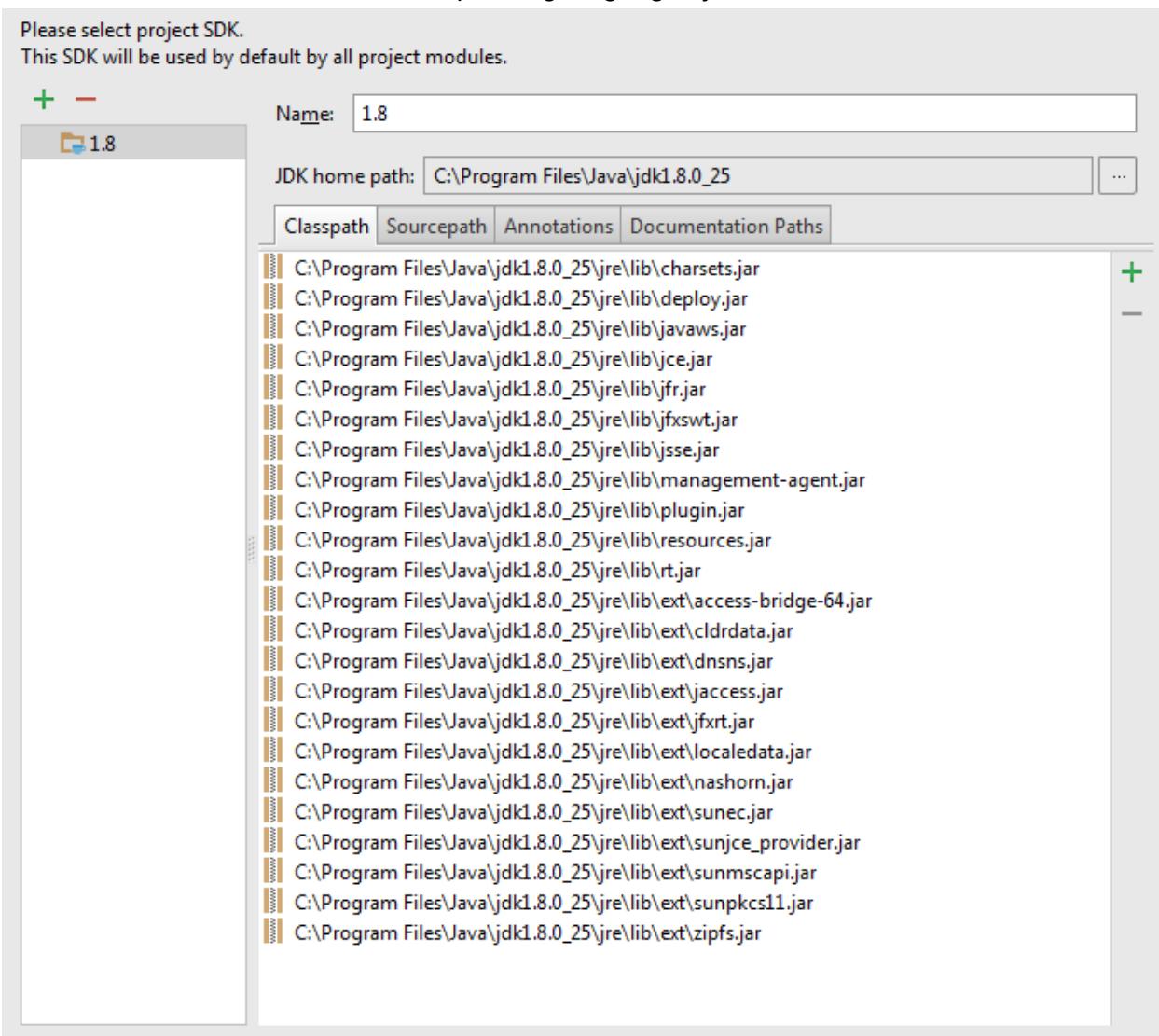


Click next till you find this screen

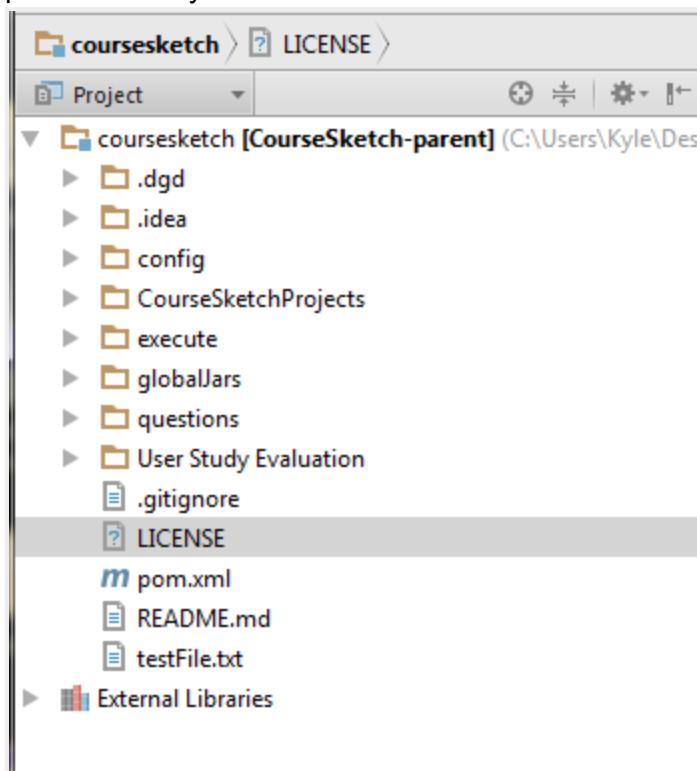


Click next till you find this screen: NOTE it may be blank

If it is blank press the + button and select the jdk you installed earlier (or install it now) The location of it is different for different operating so google “jdk location <OS>”



press next till you are done. It should show this on the left hand side.



6. Download google app engine (python)

<https://developers.google.com/appengine/downloads>

[LINUX ONLY]

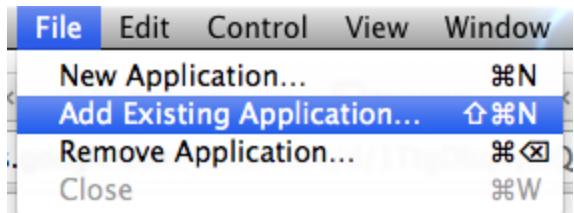
You should probably use chromium instead of chrome for all CourseSketch related work

if you do not have python installed on your computer you should install the latest version of 2.7

7. IMPORT APPLICATION into Google App Engine

after downloading open up google app engine launcher (it may be various places depending on the operating system)

After opening up we go to file and add an existing application



Navigate to course sketch -> CourseSketchProjects and select “coursesketchwebclient”
click ok
after adding it should show a line in the launcher

8. RUNNING google app engine

select the new application and click the run button (should be an arrow)
after the browse button is available click browse.
(on a mac you may have to click on the launcher to have it redraw and enable the button)

[LINUX/UNIX ONLY]

To start the google appengine on linux, please type this command
`$ /opt/google-appengine-python/dev_appserver.py
~/coursesketch/CourseSketchProjects/coursesketchwebclient/`

where `~/coursesketch` is my directory of coursesketch

9. Download and run mongodb

(<http://docs.mongodb.org/manual/installation/>)

After you have installed it you only have to run mongodb (stop after step #2) and the second half of the guide focuses on testing so you do not need to worry about.

10. Download maven

[WINDOWS ONLY]

if you are using windows then we have a script that *should* set it up for you. This can be found in the config section of course sketch. after that please skip to step 12.

[NON WINDOWS]

if you have a package manager use that to install maven.

If not windows (and no package manager):

google “install maven <OS>”

after it is installed you should be able to run:

mvn -v

and it should not throw any errors

the next thing you have to do is set the JAVA_HOME variable.

Mac

<http://stackoverflow.com/a/26456579/2187510>

Then

```
/usr/libexec/java_home -v '1.7*' Or 1.8* if you installed Java 8 instead
```

Windows

Run the batch file (RunMe.bat) in the coursesketch config\windows folder. This will setup maven, protobuf, and protoc in the directory of your choice. You may skip to step 12.

11. Download protobuf

We are currently using version 2.6
google “protobuf download 2.6”
then follow instructions

Linux:

if you run into protoc errors it could be in the wrong place.

It needs to be in:

/usr/local/bin/protoc

12. Compiling servers

On a mac open up terminal
change directory till you are inside course sketch:

On windows go to github and select the course sketch repo. Then open a terminal (it should open in the directory of CourseSketch)

Both:

```
$ mvn clean install -Dsmart -Dnosite
```

Different possible errors:

can not compile to target version 1.7: It means your JAVA_HOME is not set to the correct version (probably running 1.6) point it to 1.7 or 1.8 (which ever one you have)

JAVA_HOME is not defined: You never set the path for java home (needs to be found)

you have X number of checkstyle errors: Everything was installed correctly. But you either have an old version of master or you should fix them.

Can not find binary for protoc: it means protobuf is either in the wrong spot or was never installed correctly.

```
[ERROR] Failed to execute goal com.google.protobuf.tools:maven-protoc-plugin:0.3
.2:compile (default) on project ProtoFiles: protoc did not exit cleanly. Review
output for more information. -> [Help 1] - it means protobuf is either in the wrong spot or was
never installed correctly.
```

13: RUNNING COURSE SKETCH

App Engine

make sure google app engine is running (see #8)

Mongodb

Open up the terminal (or command prompt)

type:

mongod

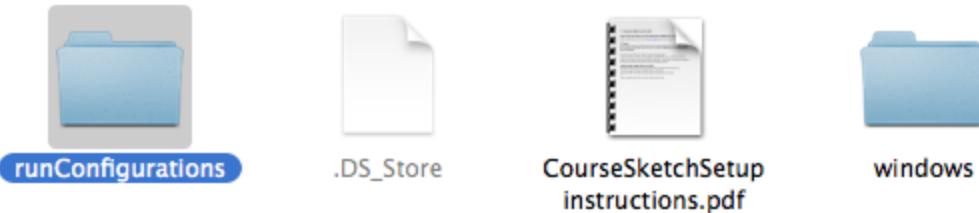
mongodb should now be running

Servers

(Automatically)

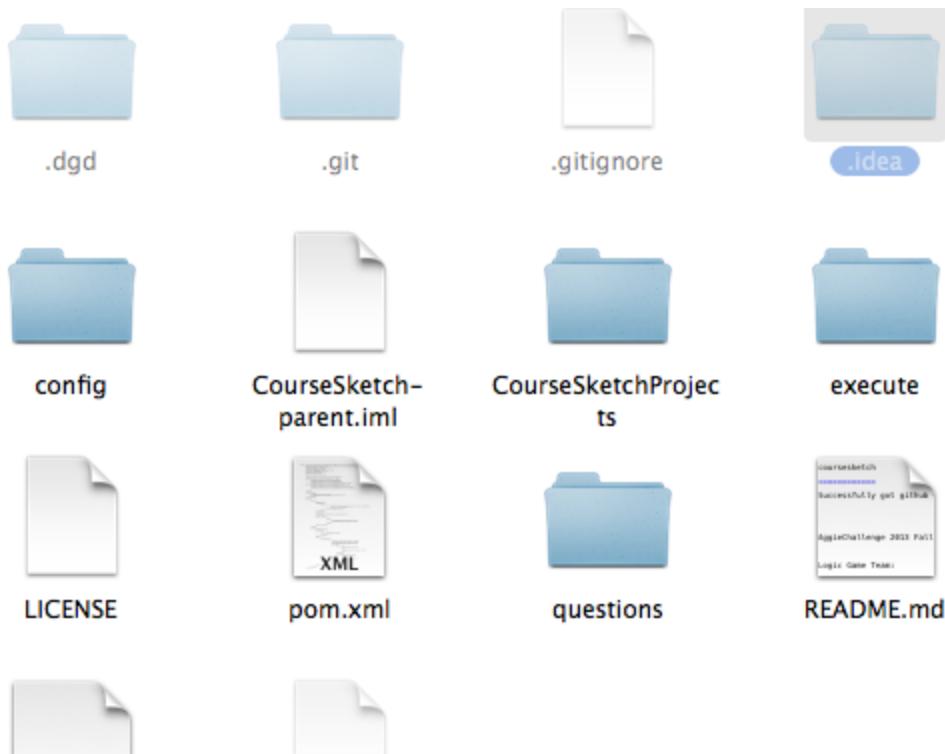
If you complete this option you do not need to go through the manual method as you will already be set up.

Inside the instructions folder there should be a folder called runConfigurations



Please copy that folder.

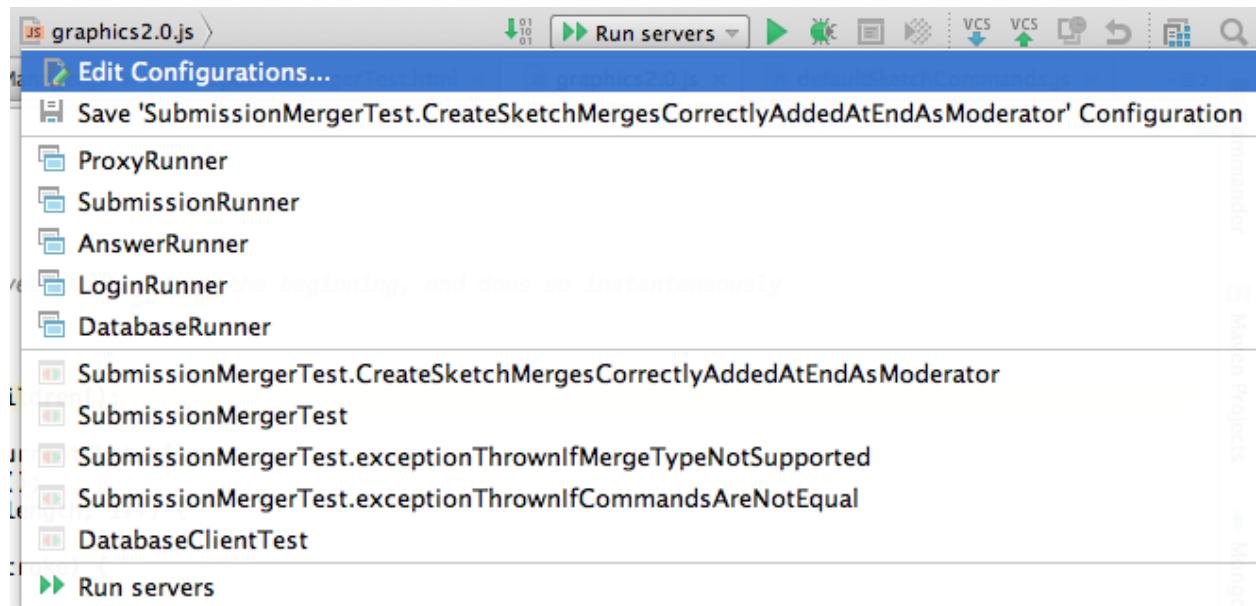
And put it inside a folder called .idea



The .idea folder should be in the main directory of CourseSketch

(Manually)

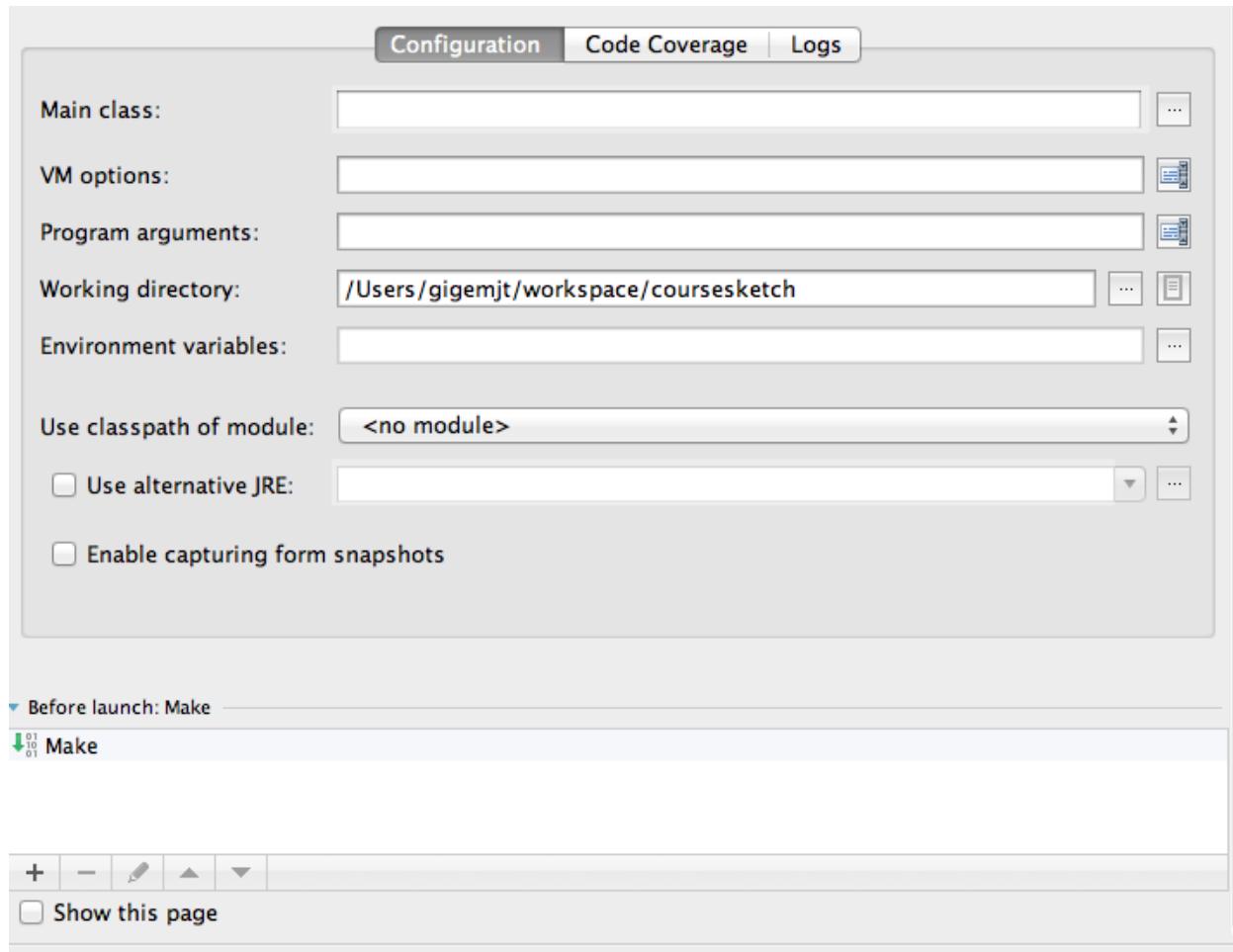
you have to create a launch configuration (select the down arrow next to the play arrow)



Click edit configurations

After the window pops up click the + button and choose "Application"

It should pop up with a blank configuration



Fill it out and a couple of additional configurations for each server:

Proxy

Name: Share Single instance only

Main class:

VM options:

Program arguments:

Working directory:

Environment variables:

Use classpath of module:

Use alternative JRE:

Enable capturing form snapshots

Submission

Name: Share Single instance only

Main class:

VM options:

Program arguments:

Working directory:

Environment variables:

Use classpath of module:

Use alternative JRE:

Enable capturing form snapshots

Answer checker

Name: Share Single instance only

Main class:

VM options:

Program arguments:

Working directory:

Environment variables:

Use classpath of module:

Use alternative JRE:

Enable capturing form snapshots

Login Server

Name: Share Single instance only

Main class:

VM options:

Program arguments:

Working directory:

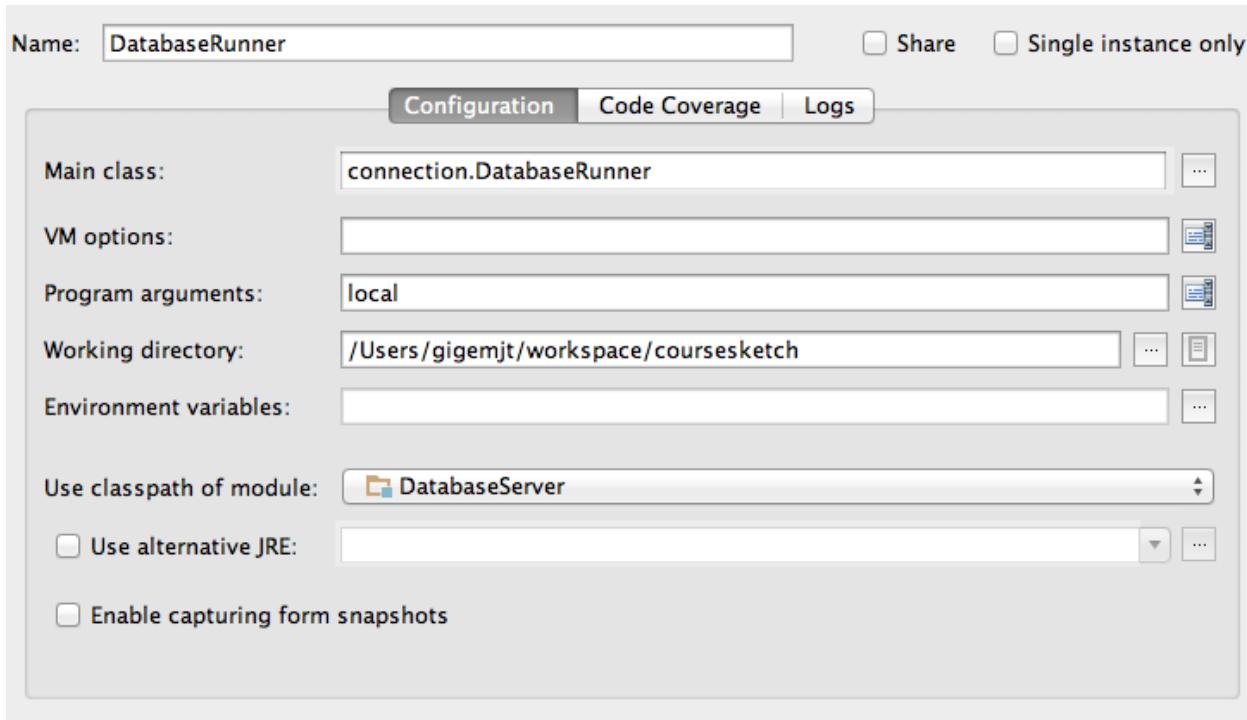
Environment variables:

Use classpath of module:

Use alternative JRE:

Enable capturing form snapshots

Database Server



Run order:

LOGIN
DATABASE
SUBMISSION
ANSWER CHECKER
PROXY

Or install multirun (below in additional setup)

14: FINAL SETUP

good news if you made it this far you are 99%. please feel free to look at the code, follow additional optional setup below or follow any additional instructions you are given in class

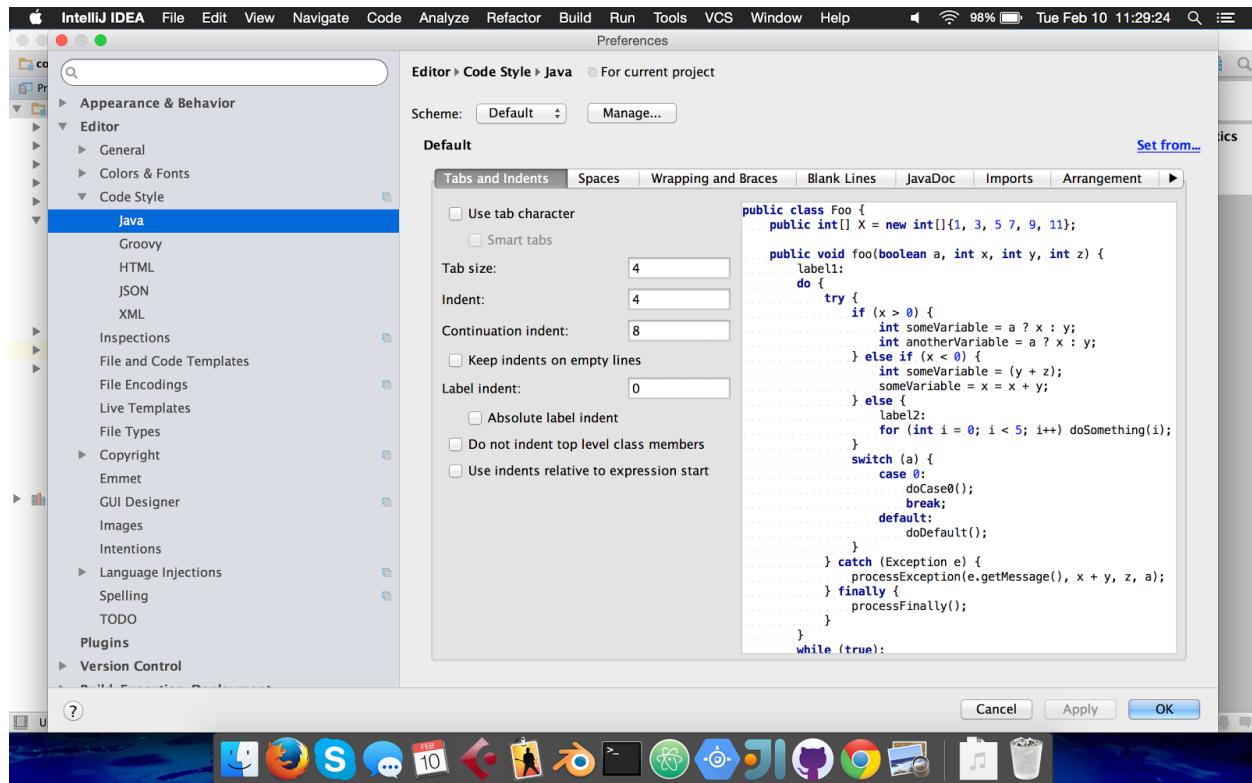
15. Additional Setup (Optional but recommended)

Import settings

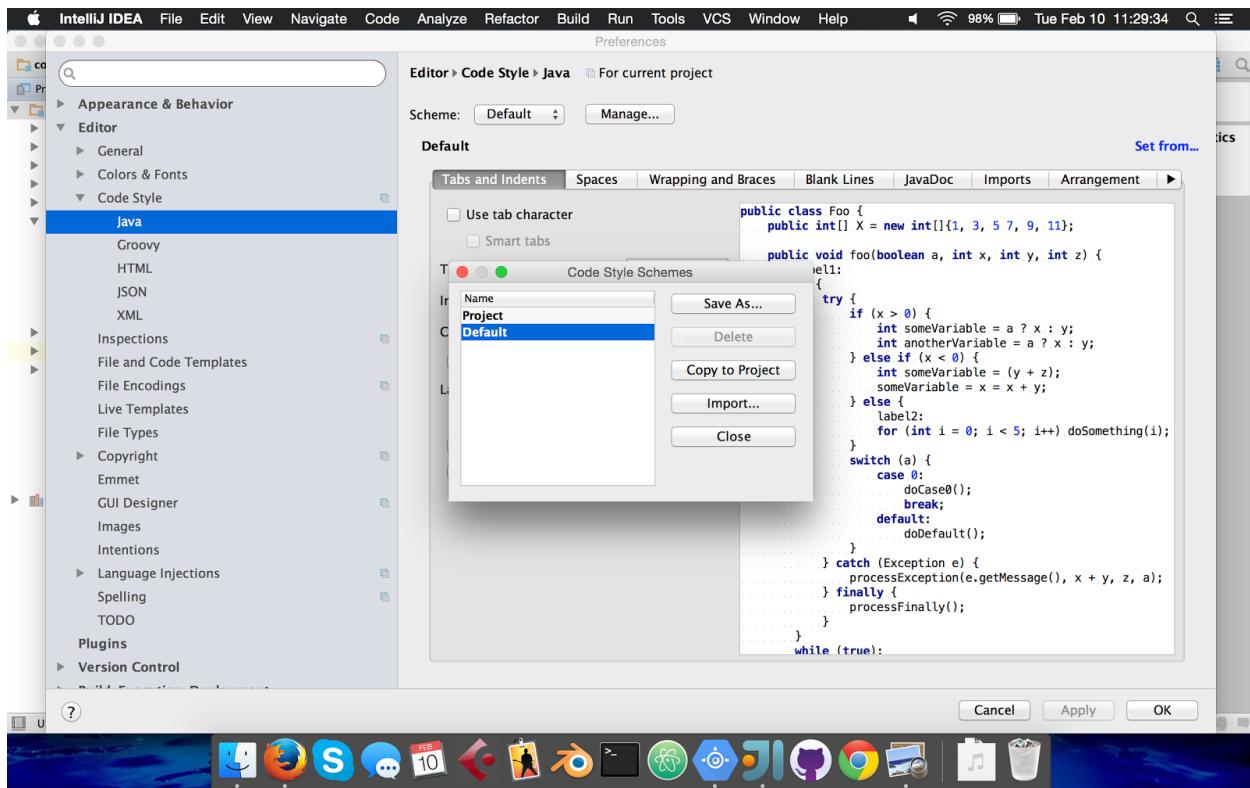
Import settings for intellij (this helps with auto formatting)

Go to settings (under preferences for mac) OR got to help -> find action and type “settings”

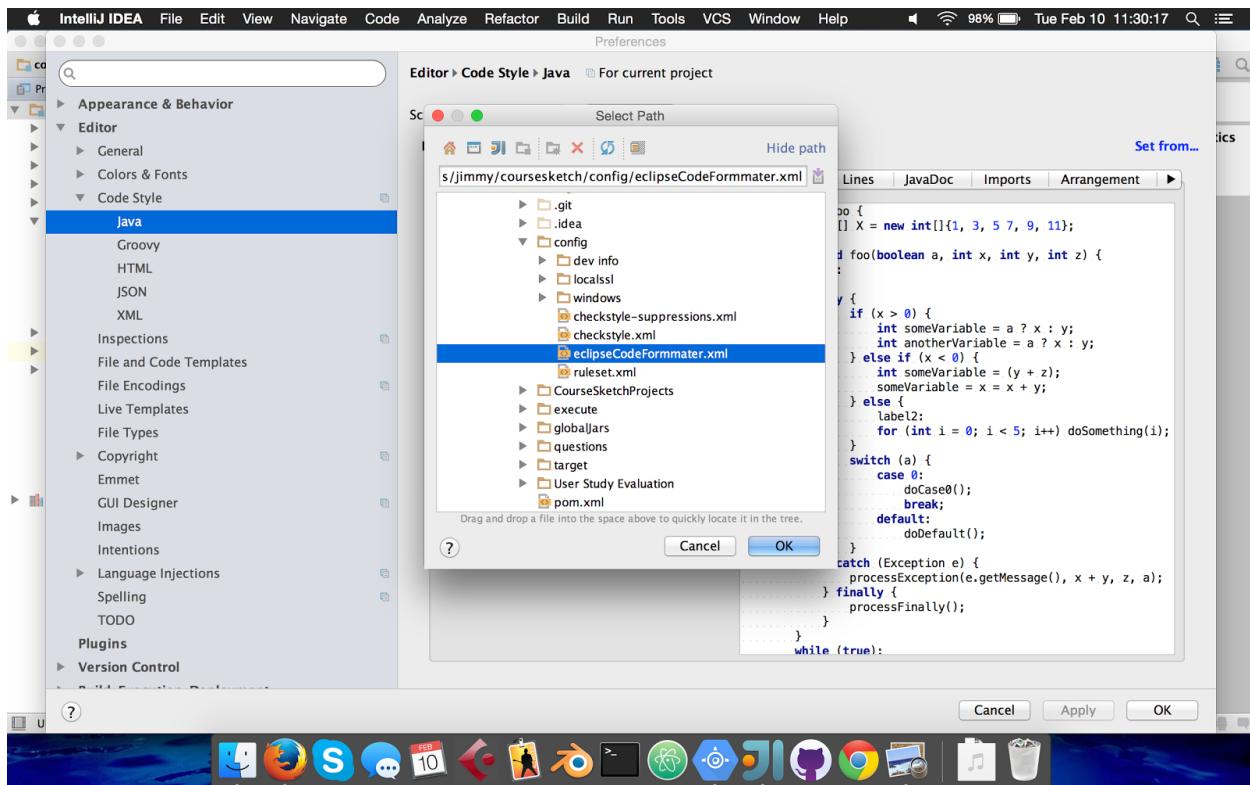
It should pull up this screen:

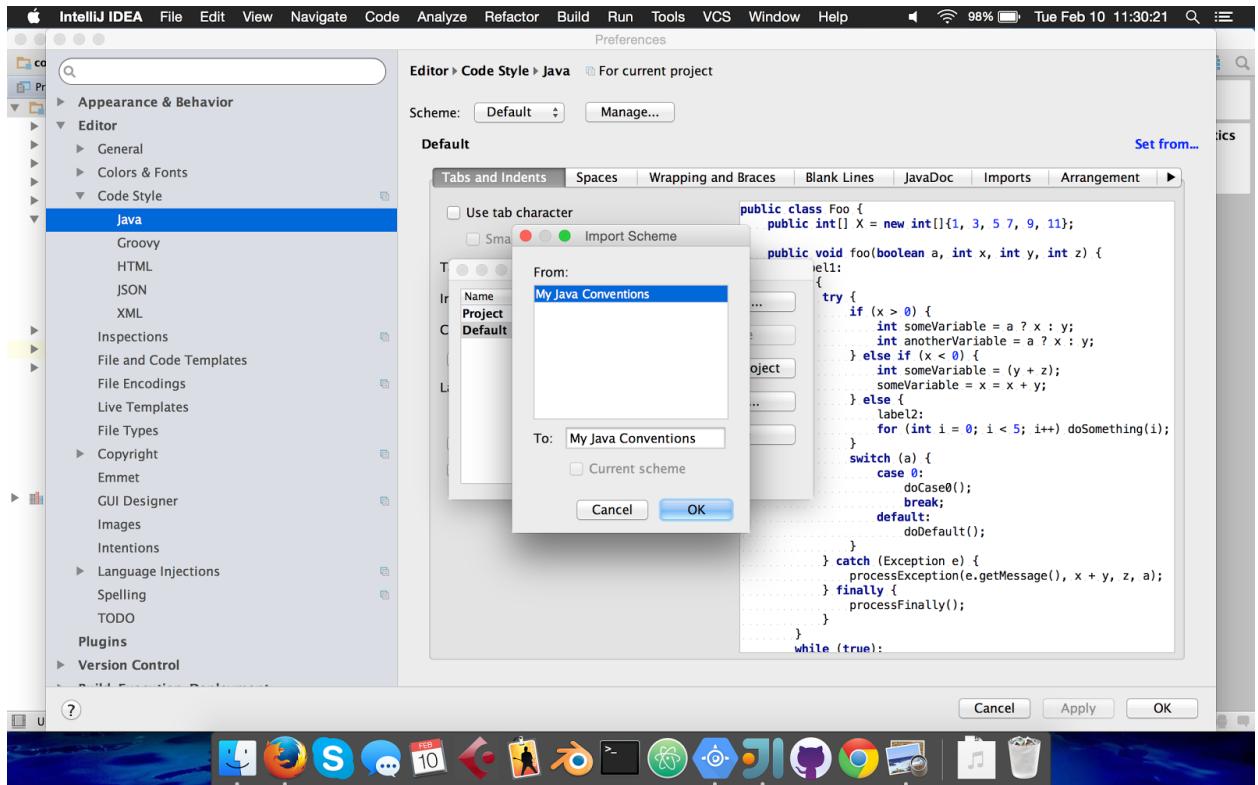


Click on Manage



Choose import





How to Install checkstyle for IntelliJ

go to help -> find actions

then type plugins

then click browse repositories

BROWSE REPOSITORIES

checkstyle Category: All ▾

Sort by: name ▾

	INSPECTION	TOOLS INTEGRATION	SEARCH AND REPLACE	CODE TOOLS	TOOLS INTEGRATION	TOOLS INTEGRATION	TOOLS INTEGRATION
	CheckStyle-IDEA						
	469,940	469,940	469,940	469,940	469,940	469,940	469,940
	★★★★★	★★★★★	★★★★★	★★★★★	★★★★★	★★★★★	★★★★★
	2 weeks ago	8 years ago	one month ago	one month ago	10 months ago	10 months ago	one month ago
	JetStyle	17,657	17,657	17,657	17,657	17,657	17,657
	TOOLS INTEGRATION	TOOLS INTEGRATION	TOOLS INTEGRATION	TOOLS INTEGRATION	TOOLS INTEGRATION	TOOLS INTEGRATION	TOOLS INTEGRATION
	jQAssistant Plugin	54	54	54	54	54	54
	SEARCH AND REPLACE	SEARCH AND REPLACE	SEARCH AND REPLACE	SEARCH AND REPLACE	SEARCH AND REPLACE	SEARCH AND REPLACE	SEARCH AND REPLACE
	QAPlug	86,432	86,432	86,432	86,432	86,432	86,432
	CODE TOOLS	CODE TOOLS	CODE TOOLS	CODE TOOLS	CODE TOOLS	CODE TOOLS	CODE TOOLS
	QAPlug - Checkstyle	63,966	63,966	63,966	63,966	63,966	63,966
	TOOLS INTEGRATION	TOOLS INTEGRATION	TOOLS INTEGRATION	TOOLS INTEGRATION	TOOLS INTEGRATION	TOOLS INTEGRATION	TOOLS INTEGRATION
	QAPlug - FindBugs	69,531	69,531	69,531	69,531	69,531	69,531
	TOOLS INTEGRATION	TOOLS INTEGRATION	TOOLS INTEGRATION	TOOLS INTEGRATION	TOOLS INTEGRATION	TOOLS INTEGRATION	TOOLS INTEGRATION
	QAPlug - Hammurapi	26,027	26,027	26,027	26,027	26,027	26,027
	TOOLS INTEGRATION	TOOLS INTEGRATION	TOOLS INTEGRATION	TOOLS INTEGRATION	TOOLS INTEGRATION	TOOLS INTEGRATION	TOOLS INTEGRATION
	QAPlug - PMD	70,059	70,059	70,059	70,059	70,059	70,059
	TOOLS INTEGRATION	TOOLS INTEGRATION	TOOLS INTEGRATION	TOOLS INTEGRATION	TOOLS INTEGRATION	TOOLS INTEGRATION	TOOLS INTEGRATION

INSPECTION

CheckStyle-IDEA

★★★★★ 469940 downloads
Updated 12/27/14 ver 4.11.2

This plugin provides both real-time and on-demand scanning of Java files with CheckStyle 6.1.1 from within IDEA.

Change Notes
Bug fixes.

Vendor
James Shiell
<http://infernus.org/>
james@infernus.org

Plugin homepage
<https://github.com/jshiell/checkstyle-idea>

Size
4.1 M

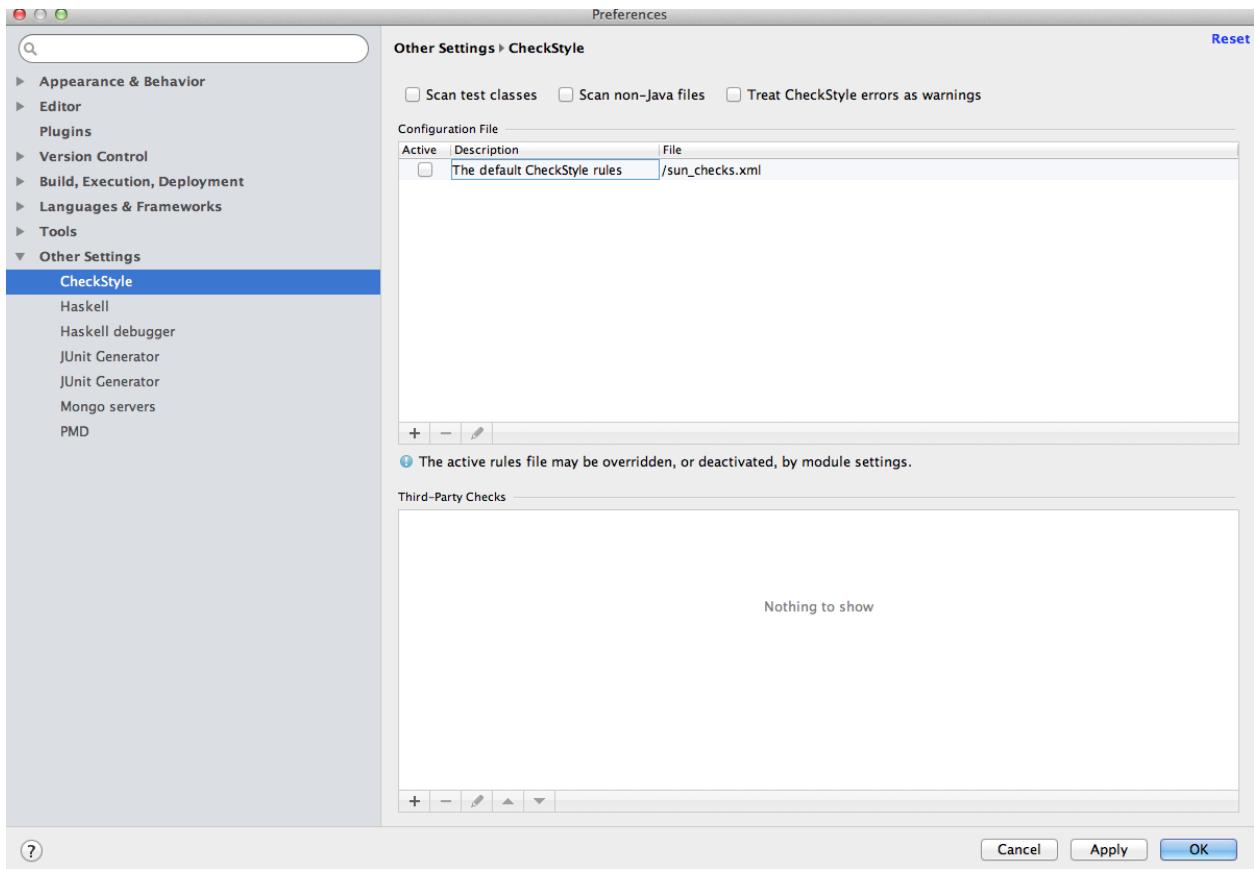
HTTP Proxy Settings... Manage repositories... Close

Now go to other settings

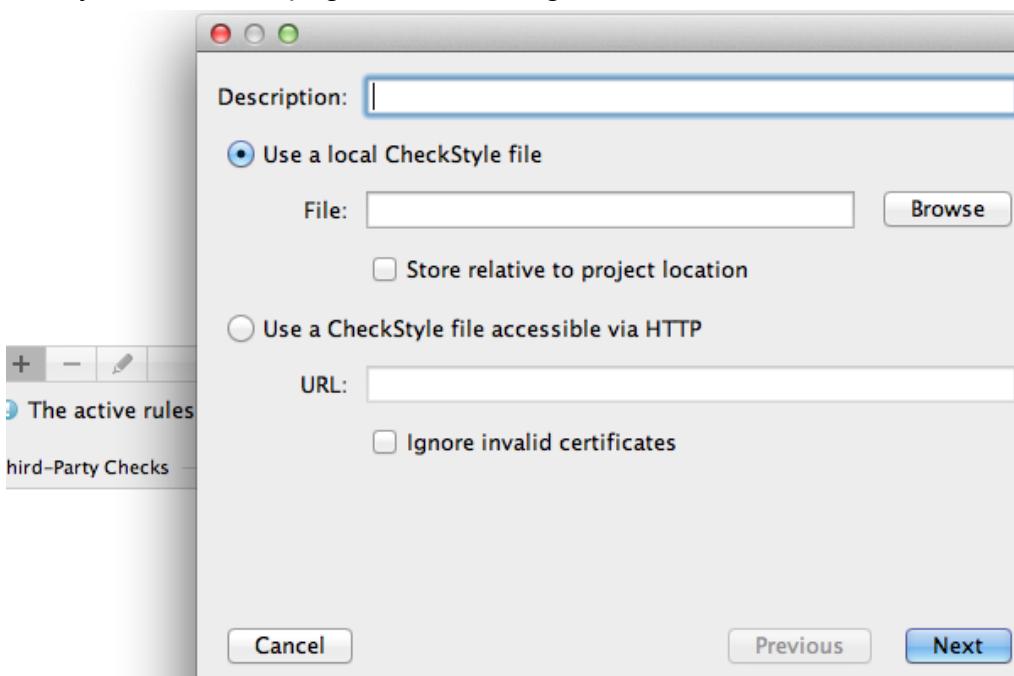
Enter action or option name:

other settings

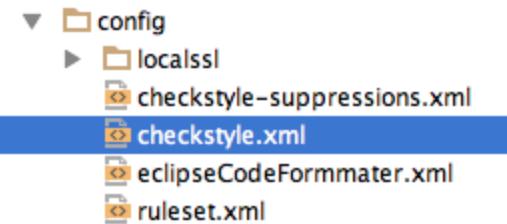
Other Settings



After you are at this page under “Configuration File” click the +

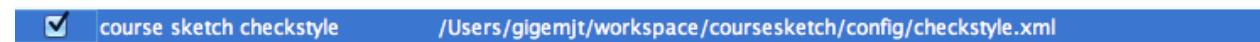


Click browse and find the file under config -> checkstyle.xml



Add a description and click next then finish

And select the newly added line. And click Apply



Install Multirun

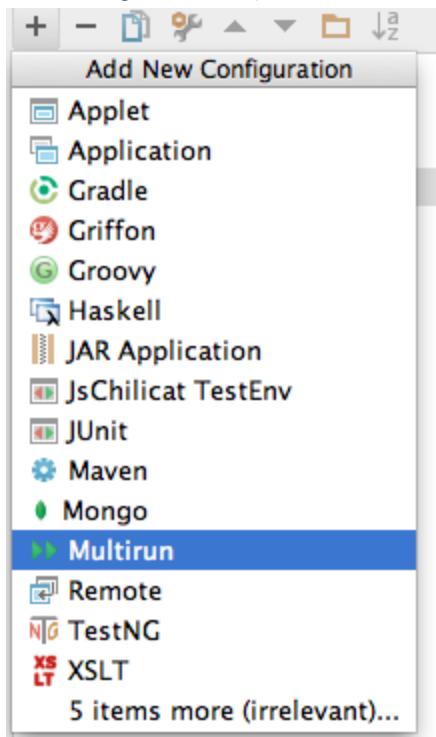
same process as above with installing checskyle

go to plugins and browse repository but type “Multirun” instead

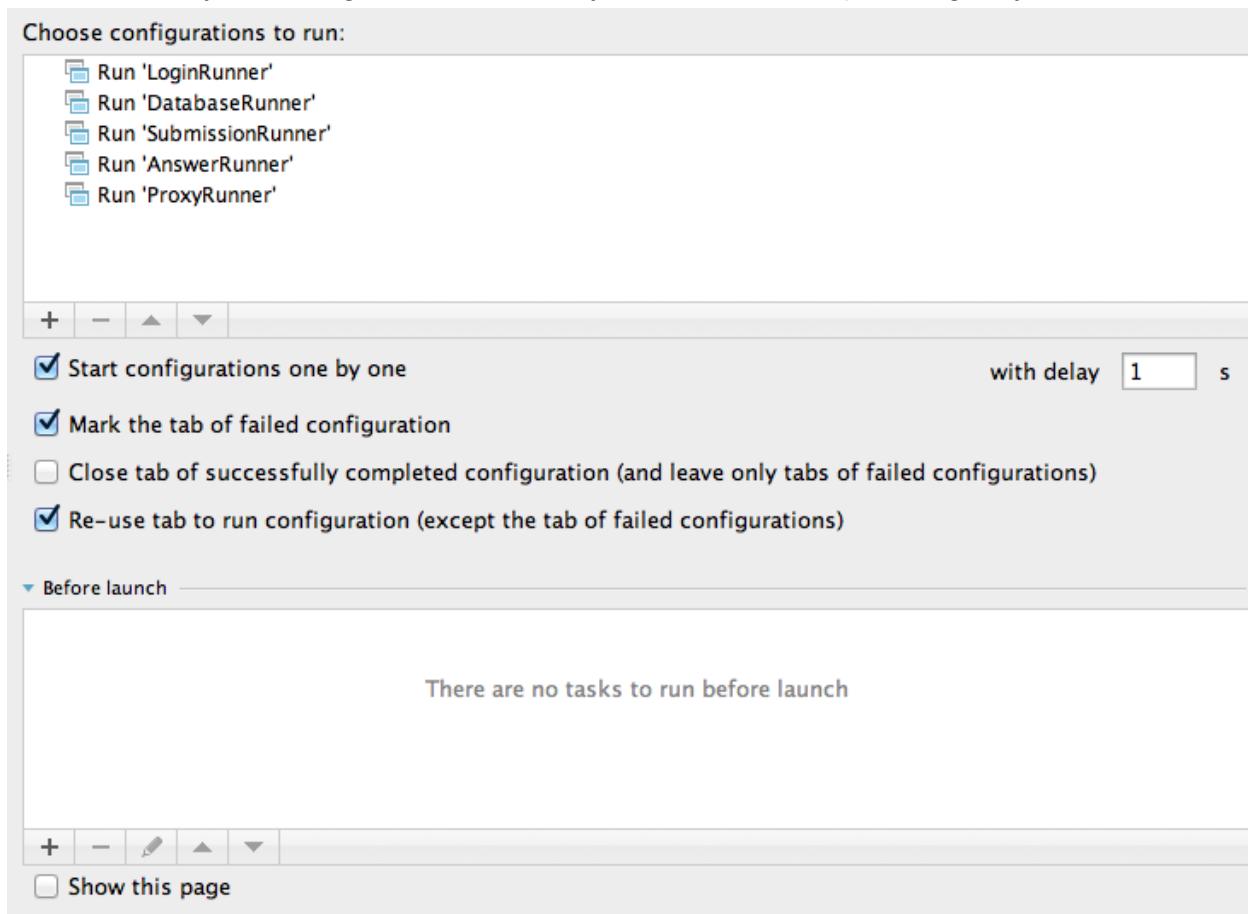
(this is the plugin webpage)

<https://plugins.jetbrains.com/plugin/7248?pr=idea>

When you add a new configuration you can create a multirun configuration that runs multiple run configurations (A lot of this is explained in running the application)



Here is what my run configuration looks like yours should end up looking very similar:



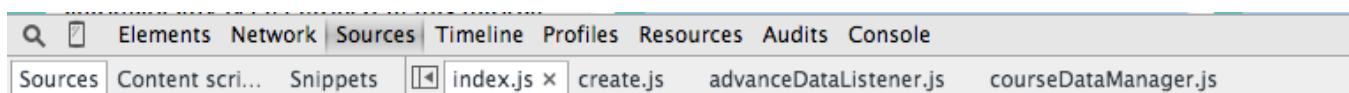
Chrome workspace

This allows you to edit files in chrome and have that reflected in your physical editor. Great for fixing tiny bugs with perfect autocomplete.

This requires that app engine is already set up (see #8)

(If you have any issues visit [here](#) but the instructions will accomplish your goal just fine)

Go into the dev console (you can right click anywhere and click inspect element). Then go to the source page view



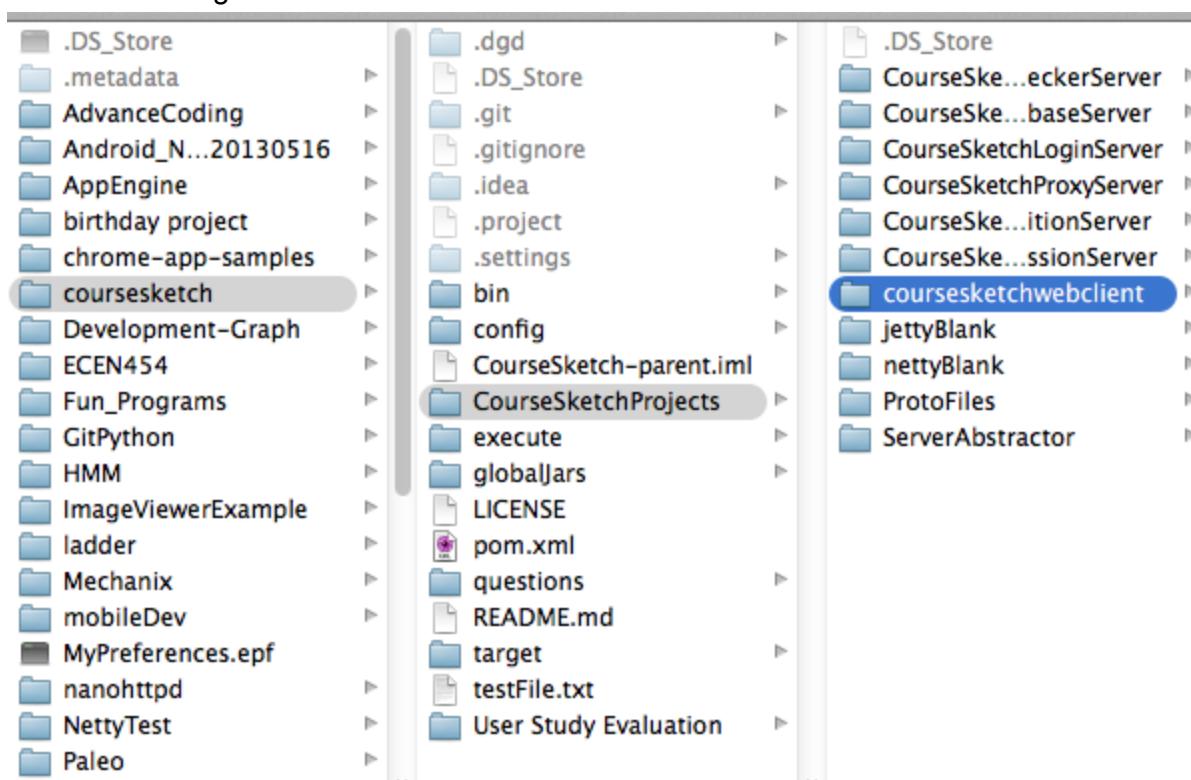
Next look on the left hand side and see a series of folders.

The screenshot shows a browser developer tools window. On the left, the 'Sources' tab is active, displaying a portion of a JavaScript file with line numbers 123 through 134. Line 130 contains the code: `ses[course.id] = course.toBase64(); // stored in memory`. A blue tooltip box labeled 'Add Folder to Workspace' appears over this line. On the right, the 'Call Stack' panel is visible, showing a stack trace with several frames, some of which have breakpoints set and are marked as 'Not Paused'. The 'Breakpoints' section also lists some checked entries.

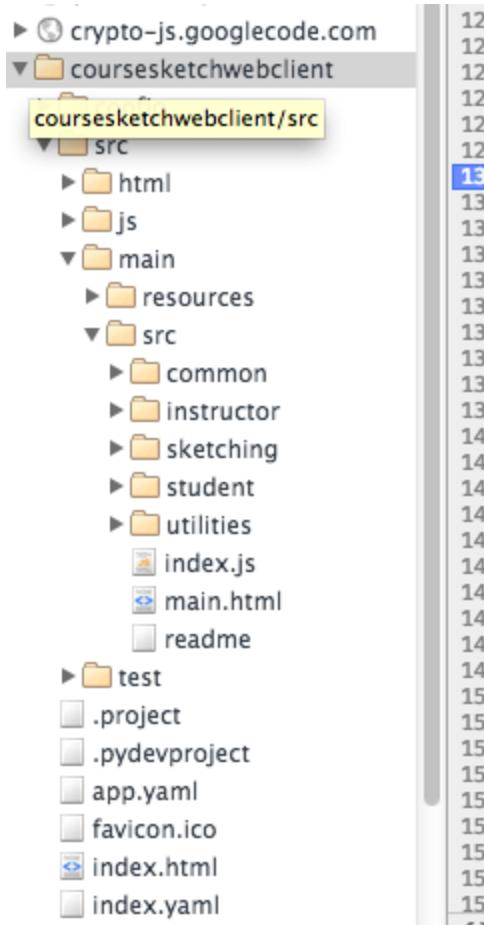
Right click and a box that says "Add Folder to Workspace" should appear.

This screenshot is similar to the previous one, but the 'Add Folder to Workspace' tooltip box is now positioned directly over the line of code where the right-click occurred (line 130). The rest of the interface, including the code editor and the right-hand panels, remains the same.

Click it and navigate to the coursesketchwebclient



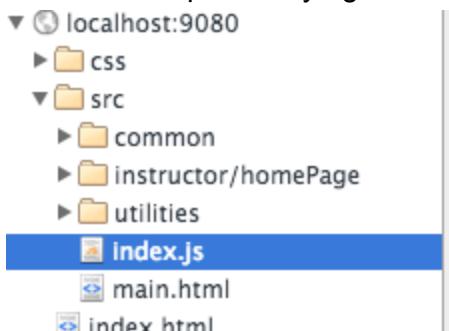
It should now have a folder that looks this when you expand it.



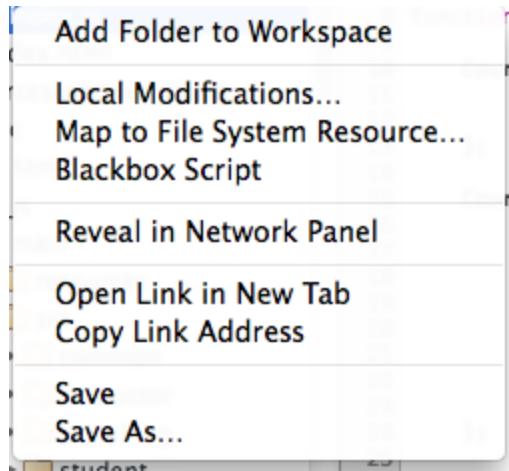
Now we need to map a file to the physical location.

So this is the actual website itself coming from the local server.

You need to specifically right click index.js (any other and a bunch of things get messed up)



Now select “Map to File System Resource”



Then choose the option of index.js that is shown below.

It was the first one for me but it may not be the first one for you.

```
defined(CourseSketch)) {  
  alue  
  tch  
  tch  
  t).  
    index.js  
  coursesketchwebclient/src/main/src/index.js  
} }
```

Now if you edit a js file in chrome it will update locally if you save it.

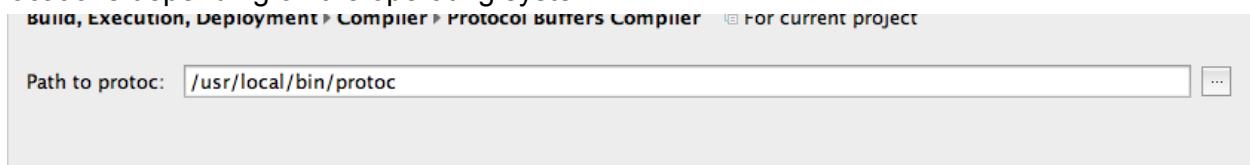
This also allows you to edit javascript without needing to reload the page and instead it will just run the new code.

Fixing protobuf errors

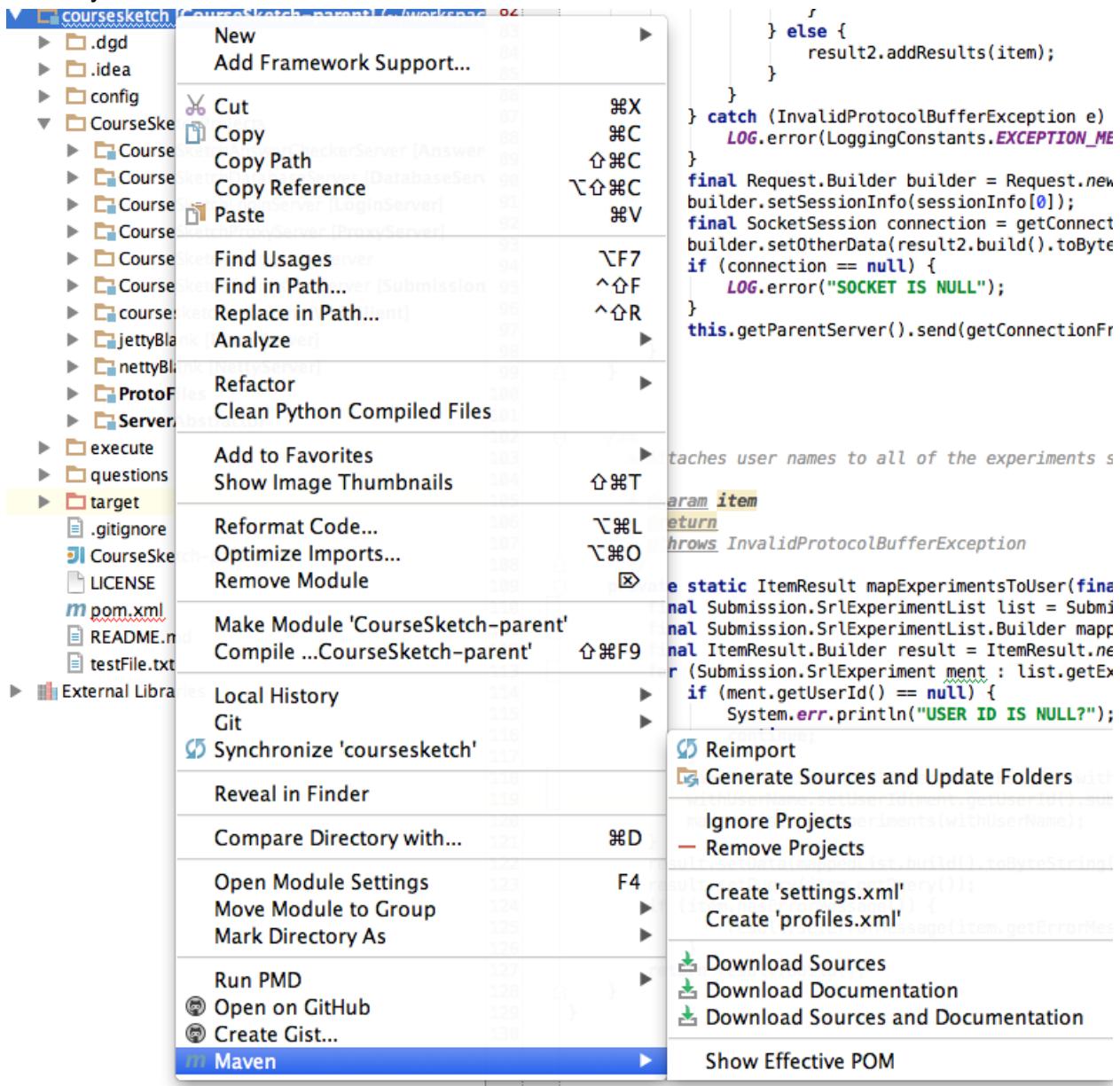
first install the protobuf plugin for intellij

this can be done by opening a protobuf file and hoping intellij recognizes it and asks to install the plugin.

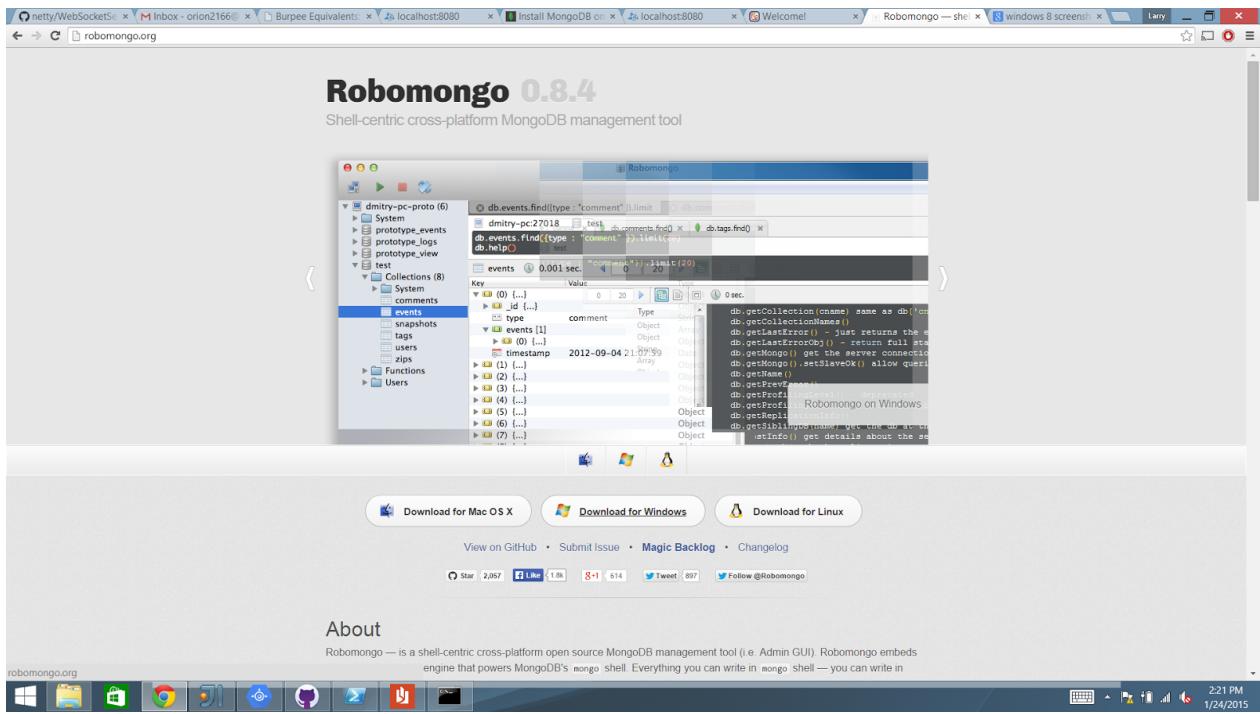
After the plugin is installed you need to provide a link to the protobuf compiler it is in different locations depending on the operating system.



Now after that select “Maven” and click “Reimport” and it should have the protobuf set up correctly.



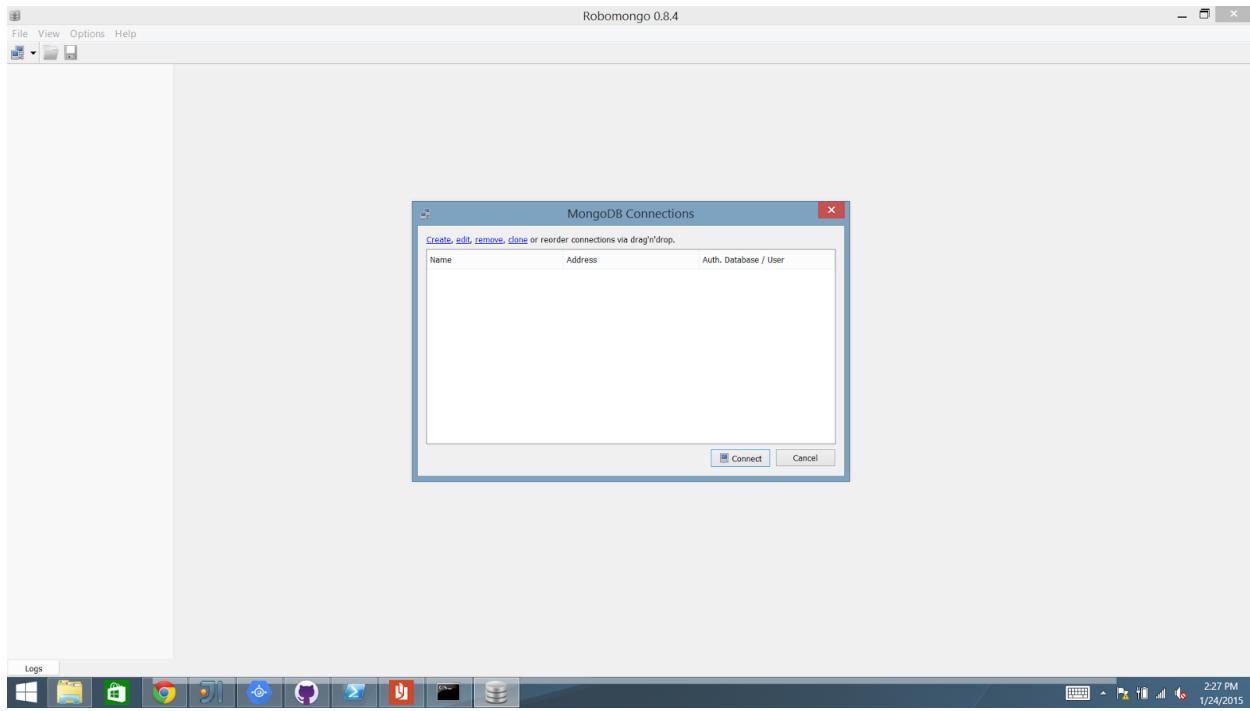
Viewing Mongo Database



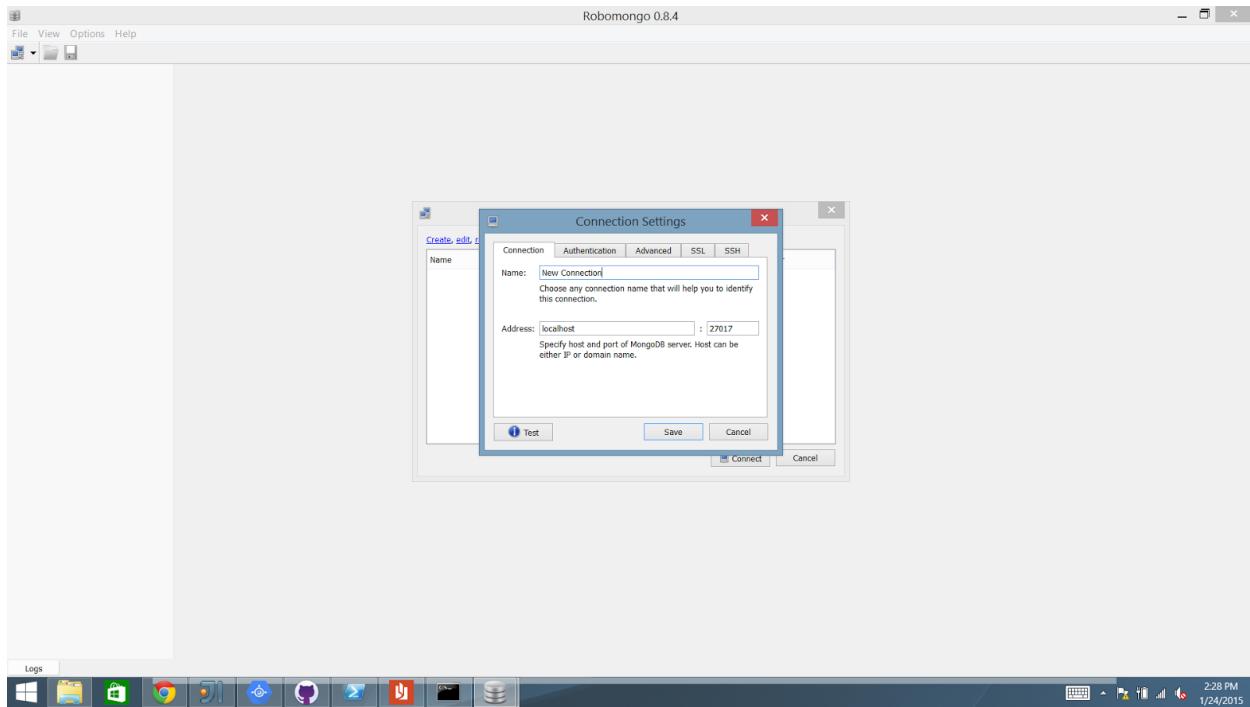
To install ROBOMONGO

Google Robomongo and click the right operating system.

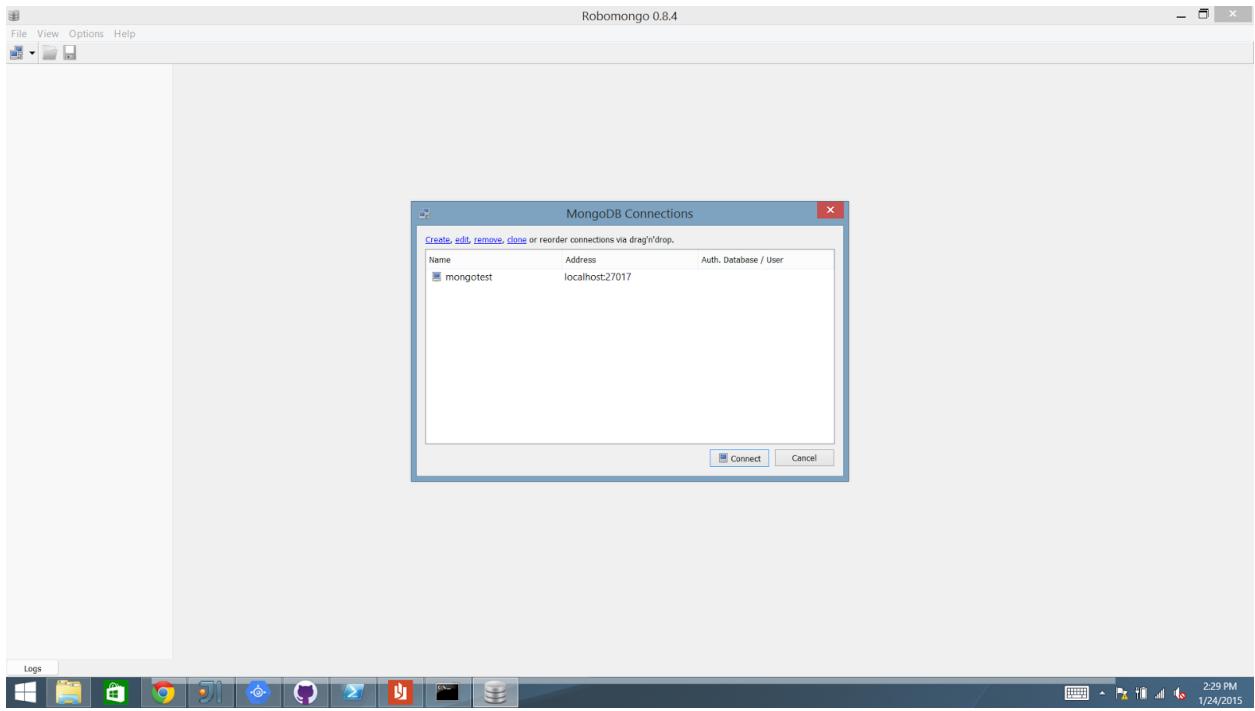
Open it up and click connect under the file bar.



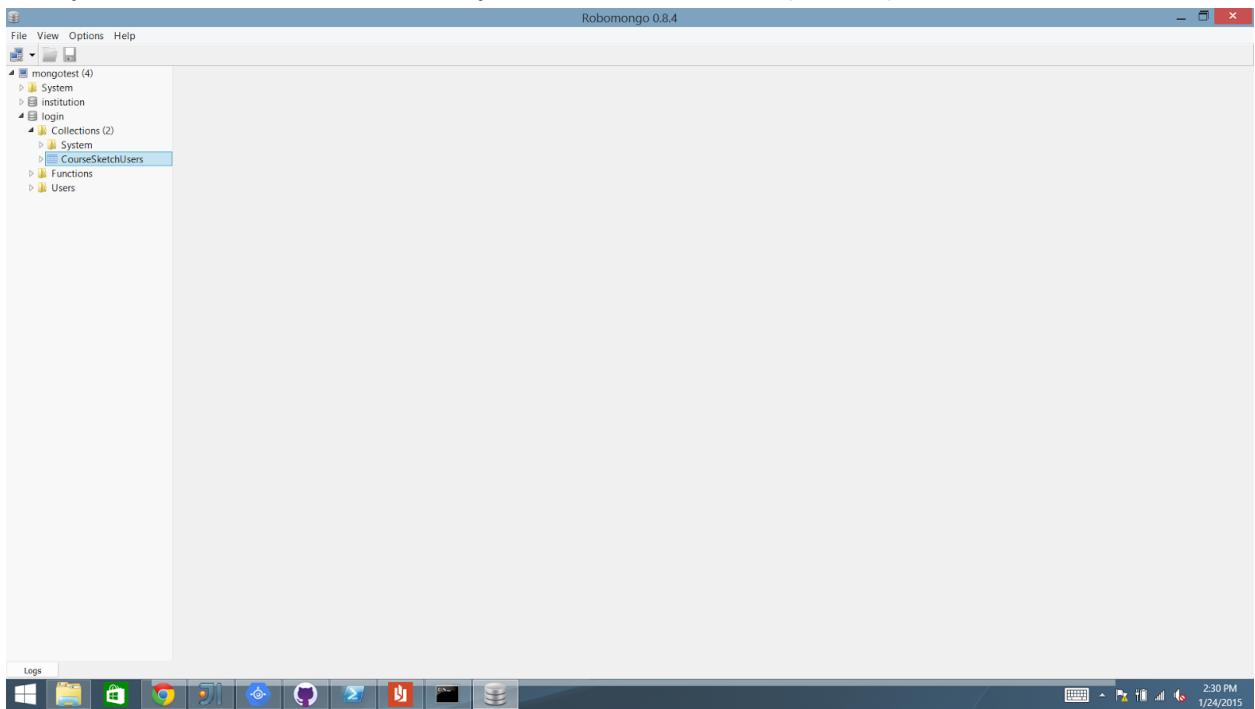
click create and then you should see this screen



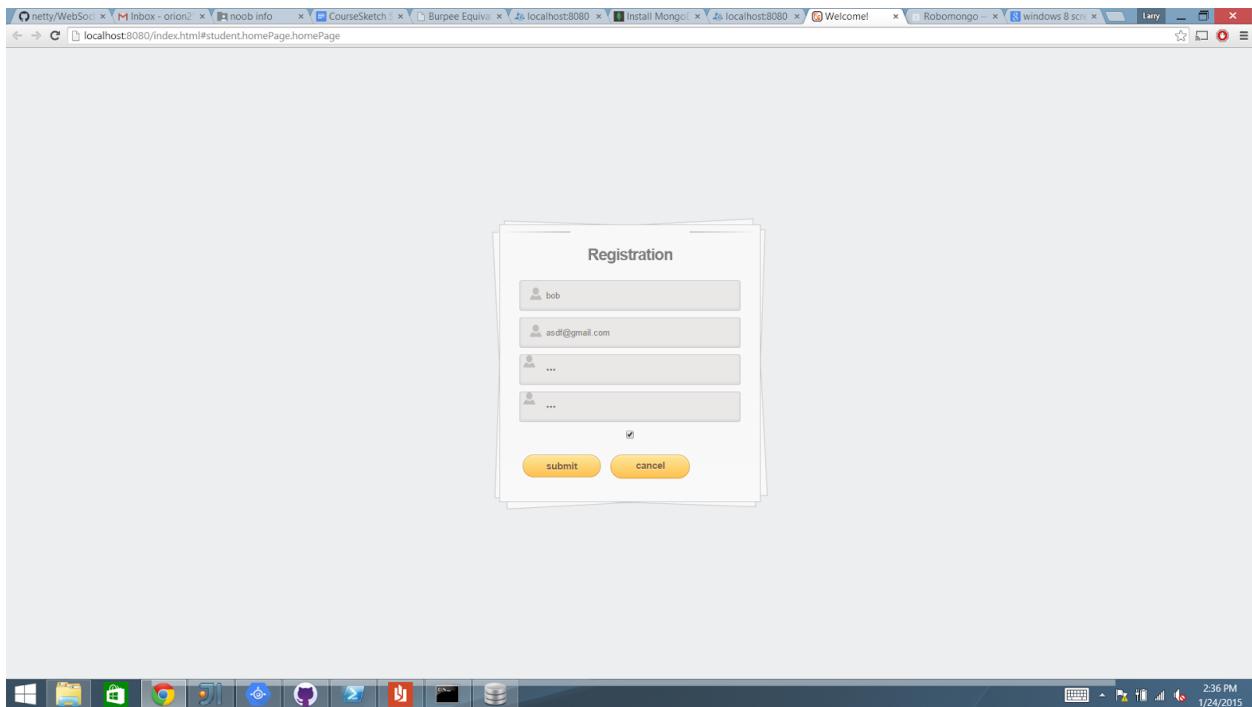
put something in the name section EX(mongotest)
then press save



after you selected and connect and you should be able to open it up and look like this



How to set up fake data make an instructor account



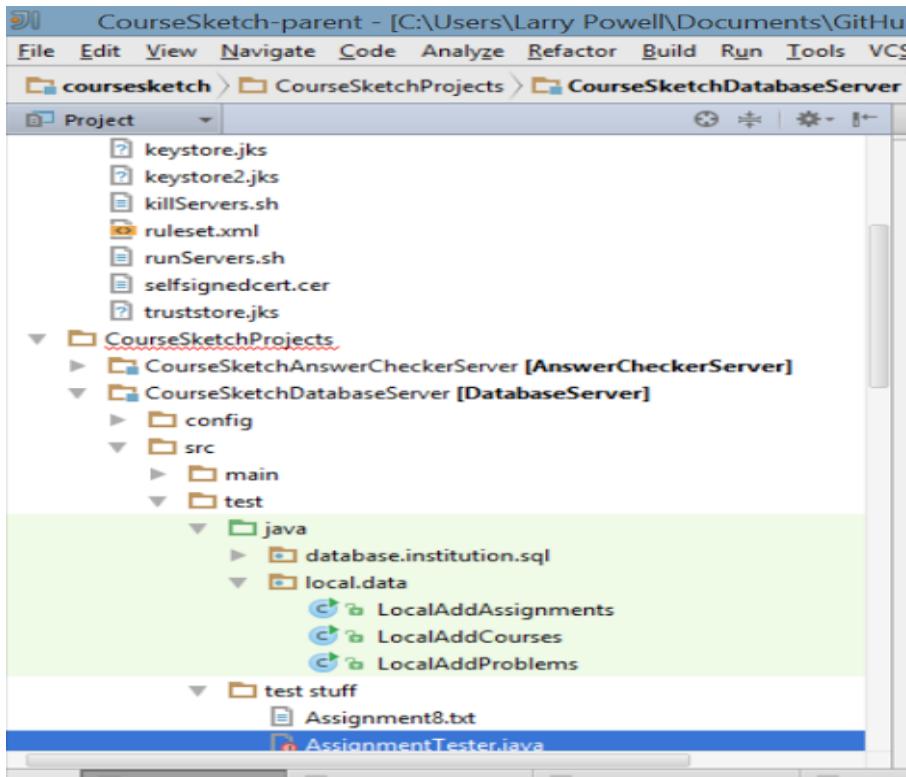
go back to Robomongo

A screenshot of the Robomongo application. On the left, the sidebar shows a database named "mongotest" with several collections: "System", "institution", "User_Updates", "Users", "login", and "CourseSketchUsers". The "CourseSketchUsers" collection is selected, and its contents are displayed in a table on the right. The table has columns for "Key", "Value", and "Type". One document is shown in the table:

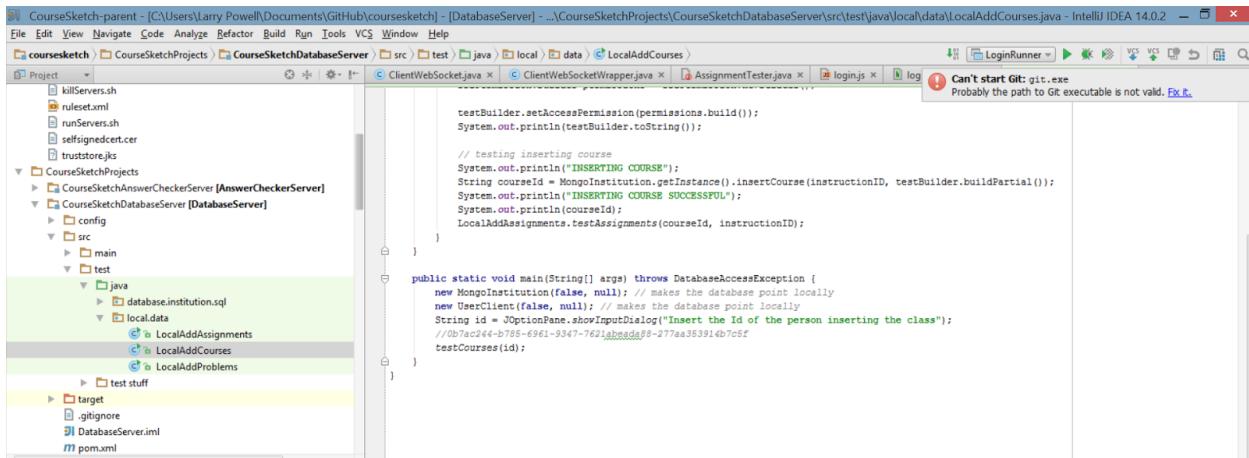
Key	Value	Type
ObjectID("54c4117027453ee355d63024")	(9 fields) _id: ObjectID("54c4117027453ee355d63024") bob: String 1000e159d1b524e2e99108e2dfea6fe0e7e8ed53164a61dbdaec515332ad7d02d...: String asdf@gmail.com: String true: Boolean 9bc36327-09d6-4d1a-b71c-22310acc204c-237a708ad3f8c2a: String 11aa16ef-8bf3-4f4e-957b-f66e301fa6d4-1fa281263941482a: String e0bc5dcf-3b3b-44fc-aa43-3ee9f2dd8cd0: String 3fd2d664-1b20-498e-b9c6-1230334db52: String	Object ObjectID String String Boolean String String String String

COPY the instructor ID

go back to intellij and open up database server



Open up LocalAddCourses



run program you should see this window

coursesketch] - [DatabaseServer] - ...\\CourseSketchProjects\\CourseSketchDatabaseServer\\src\\test\\java\\local\\data\\LocalAddCourses.java - IntelliJ IDEA 14.0.2

File Window Help

src test java local data LocalAddCourses

ClientWebSocket.java ClientWebSocketWrapper.java AssignmentTester.java login.js log

Can't start Git: git.exe
Probably the path to Git executable is not valid. Fix it.

```

import ...

public class LocalAddCourses {
    public static void testCourses(String instructionID) throws DatabaseAccessException {
        String[] name = new String[]{"CourseSketch 101"};
        String[] description = new String[]{"Hi Welcome to CourseSketch, you have automatically been enrolled in this tutorial."
            + " To expand the description of a class click the down arrow."};
        for (int k = 0; k < name.length; k++) {
            SrlCourse.Builder testBuilder = SrlCourse.newBuilder();
            testBuilder.setAccess(SrlCourse.Accessibility.SUPER_PUBLIC);
            testBuilder.setSemester("FALL");
            testBuilder.setName(name[k]);
            testBuilder.setDescription(description[k]);
            SrlPermission.Builder permissions = SrlPermission.newBuilder();

            testBuilder
                .setInsertionId(1)
                .setPersonId("54c417460d453d5147378a92")
                .setCourseId("54c417460d453d5147378a92")
                .setAssignmentId("54c417460d453d5147378a92")
                .setProblemBankId("54c417460d453d5147378a92")
                .setName("Problem41")
                .setGradeWeight("50%");

            insertCourse(instructionID, testBuilder.buildPartial());
            System.out.println("INSERTING COURSE SUCCESSFUL");
            System.out.println(courseId);
            LocalAddAssignments.testAssignments(courseId, instructionID);
        }
    }
}

```

LocalAddCourses

paste the ID you copied in Robomongo and put it in the field
you should see these values at the bottom

Run: LoginRunner DatabaseRunner SubmissionRunner AnswerRunner ProxyRunner LocalAddCourses

```

courseId: 54c417460d453d5147378a91
assignmentId: "54c417460d453d5147378a92"
problemBankId: "54c417460d453d5147378a93"
name: "Problem41"
gradeWeight: "50%"

INSERTING PROBLEM
{ "$addToSet": { "Admin": { "$each": [ "group54c417460d453d5147378a90"]}, "Mod": { "$each": [ "group54c417460d453d5147378a8f"]}, "Users": { "$each": [ "group54c417460d453d5147378a8e"]}}}
INSERTING PROBLEM SUCCESSFUL

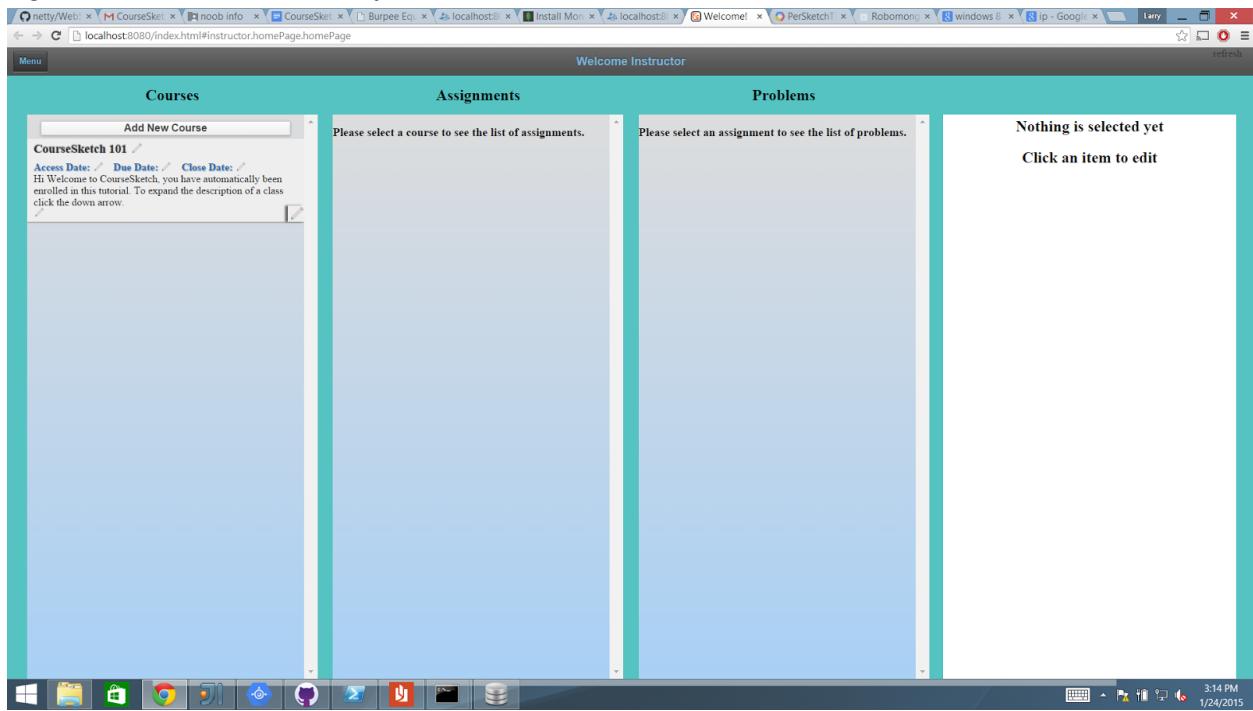
Process finished with exit code 0

```

All files are up-to-date (moments ago)

422:1 CRLF:1 UTF-8:1 Git: allnetty 3:06 PM 1/24/2015

log back into course sketch you should see this window



YOU MADE IT!!!!!!

If you made it this far you followed all instructions and you are now 100% setup to develop coursesketch.