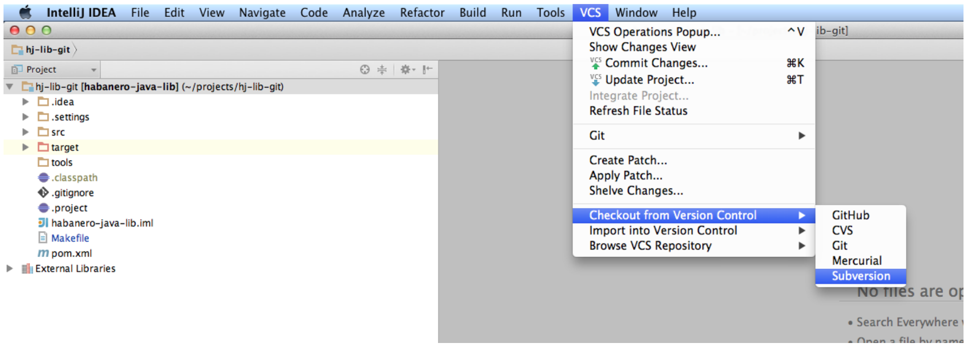
Using IntelliJ to Download and Run lab_1

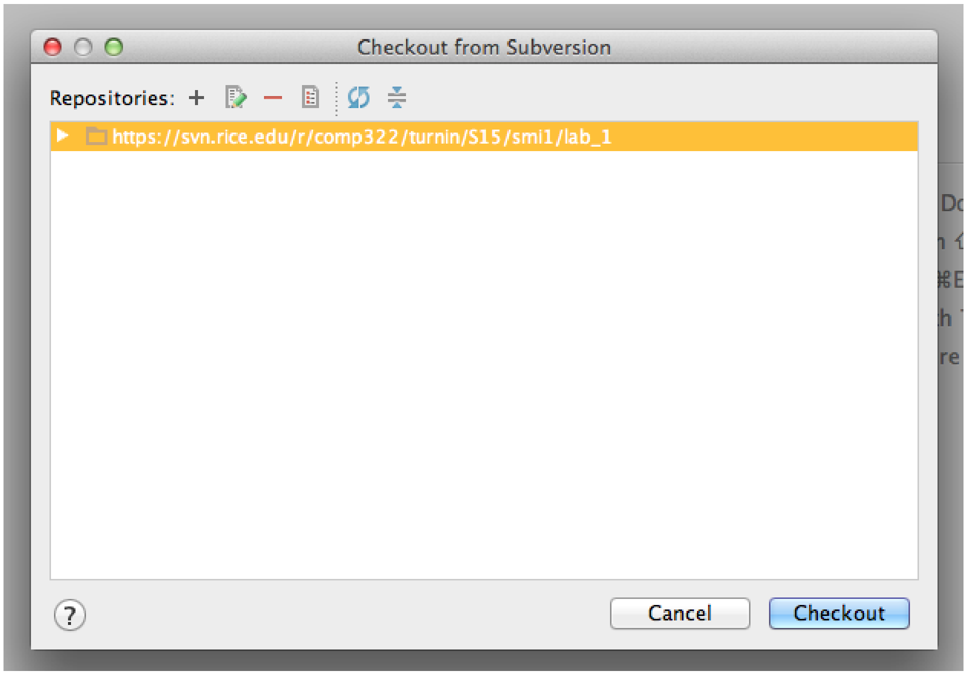
These instructions are for IntelliJ. If you are using Eclipse, similar instructions on running HJlib programs can be found at http://www.cse.wustl.edu/~cosgroved/courses/cse231/f16/javaagent/.

Instructions

1. Prepare to checkout the lab_1 project from subversion

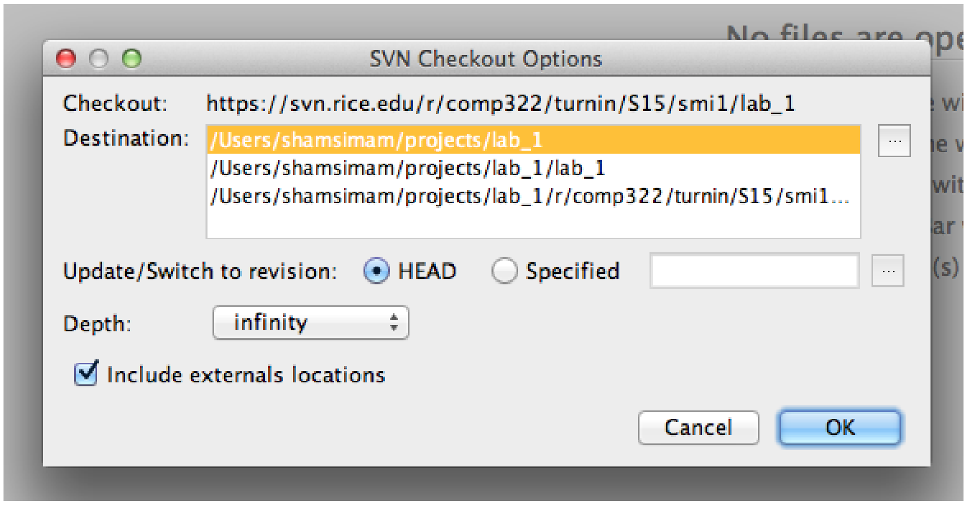


2. Use the "+" button to add the repository: https://svn.rice.edu/r/comp322/turnin/S15/**NETID**/lab_1/



https://svn.rice.edu/r/comp322/turnin/S15/smi1/lab_1

3. Confirm the directory where the directory where the project will be downloaded to.



SVN Checkout Options

Checkout: https://svn.rice.edu/r/comp322/turnin/S15/smi1/lab_1

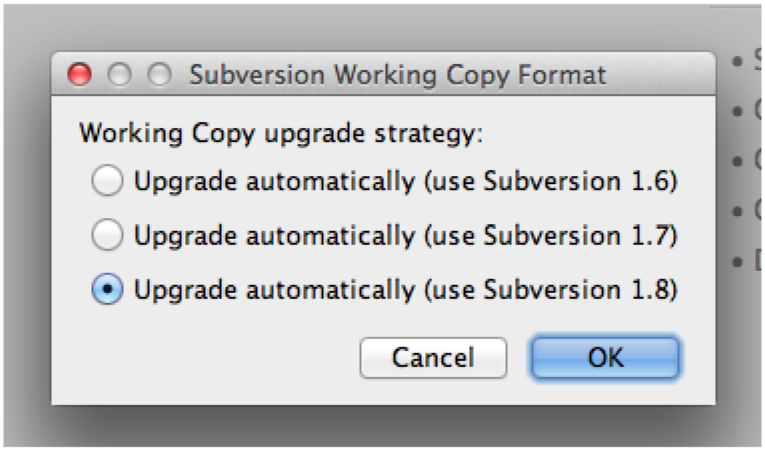
Destination: /Users/shamsimam/projects/lab_1
/Users/shamsimam/projects/lab_1/lab_1
/Users/shamsimam/projects/lab_1/r/comp322/turnin/S15/smi1...

Update/Switch to revision: ● HEAD ○ Specified

Depth: infinity

☑ Include externals locations

4. Confirm the subversion version to use for the directory format.

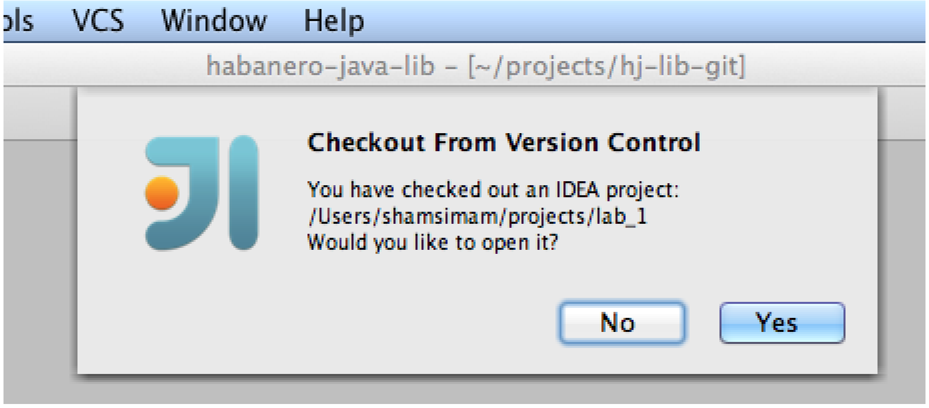


Subversion Working Copy Format

Working Copy upgrade strategy:

○ Upgrade automatically (use Subversion 1.6)
○ Upgrade automatically (use Subversion 1.7)
● Upgrade automatically (use Subversion 1.8)

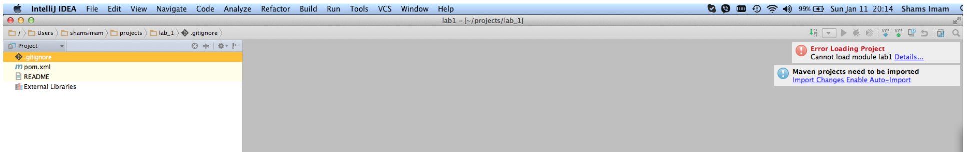
5. After the project is downloaded successfully, confirm to open the project.



Checkout From Version Control

You have checked out an IDEA project:
/Users/shamsimam/projects/lab_1
Would you like to open it?

6. Import as a Maven project. Right-click pom.xml, and select "Add as Maven Project". This option is towards the bottom of the options list. Once you have done that, again right-click on the same pom.xml le and you should now see a "Maven" option towards the bottom of the list. Click on Maven > Reimport to be sure IntelliJ has pulled in the JARs for this project. Finally, go to File > Project Structure and in the pop up select "Modules" from the left menu. You should see three JARs listed in the central pane: hjlib-cooperative:0.1.8, asm-all:5.0.3, and junit:3.8.2. Check the checkboxes next to each, hit "Apply" at the bottom right of the window, and close the window by clicking "OK".

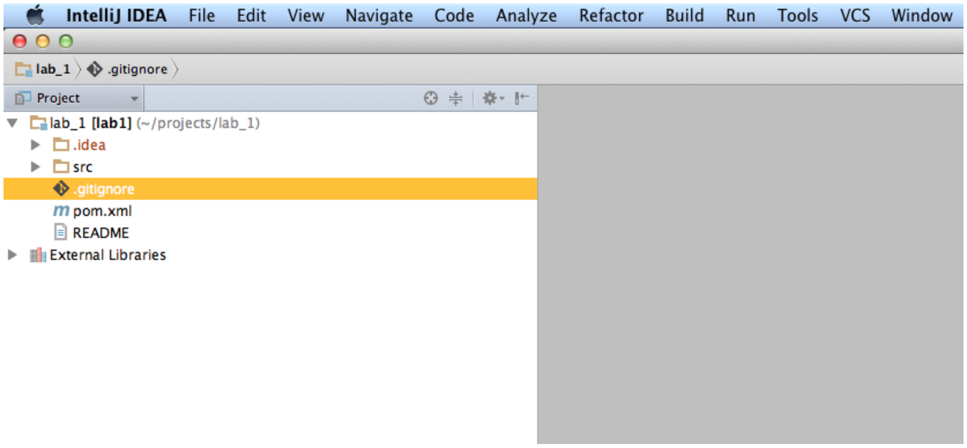


IntelliJ IDEA File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
99% Sun Jan 11 20:14 Shams Imam
lab1 – [~/projects/lab_1]
/ Users shamsimam projects lab_1 .gitignore
Project
.gitignore
pom.xml
README
External Libraries

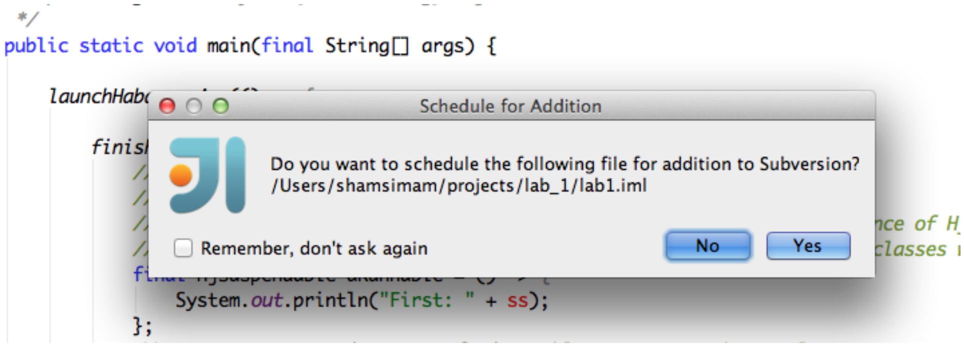
Error Loading Project
Cannot load module lab1 Details...

Maven projects need to be imported
Import Changes Enable Auto-Import

7. Check the appropriate JDK is selected. Navigate to File > Project Structure and verify that in the "Project" pane you have \1.8" selected as the \Project SDK.
8. Initial directory layout of the project.


IntelliJ IDEA File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window
lab_1 .gitignore
Project
lab_1 [lab1] (~/projects/lab_1)
.idea
src
.gitignore
pom.xml
README
External Libraries

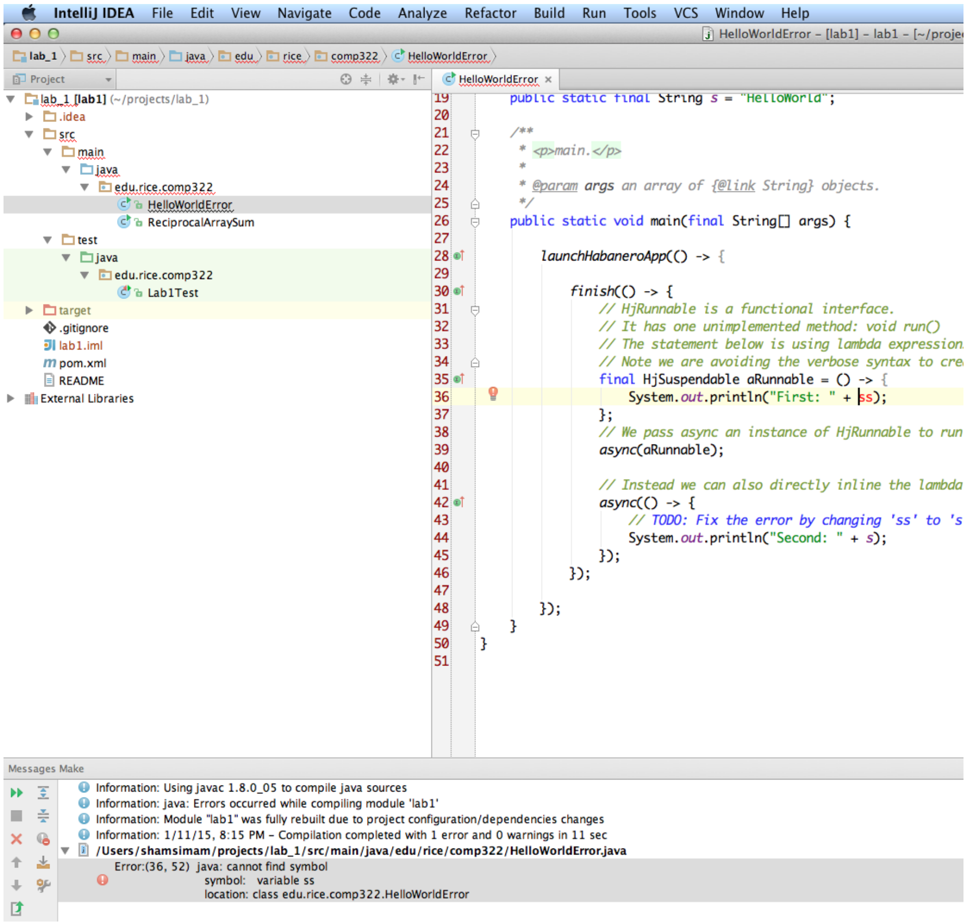
9. Click on "NO" when a popup shows up asking you whether you want to add an *.iml file to subversion.



Do you want to schedule the following file for addition to Subversion?
/Users/shamsimam/projects/lab_1/lab1.iml

```
 */
public static void main(final String[] args) {

    launchHabc

        finis
                                                                nce of H
                                                                classes
        fi
            System.out.println("First: " + ss);
        };
```

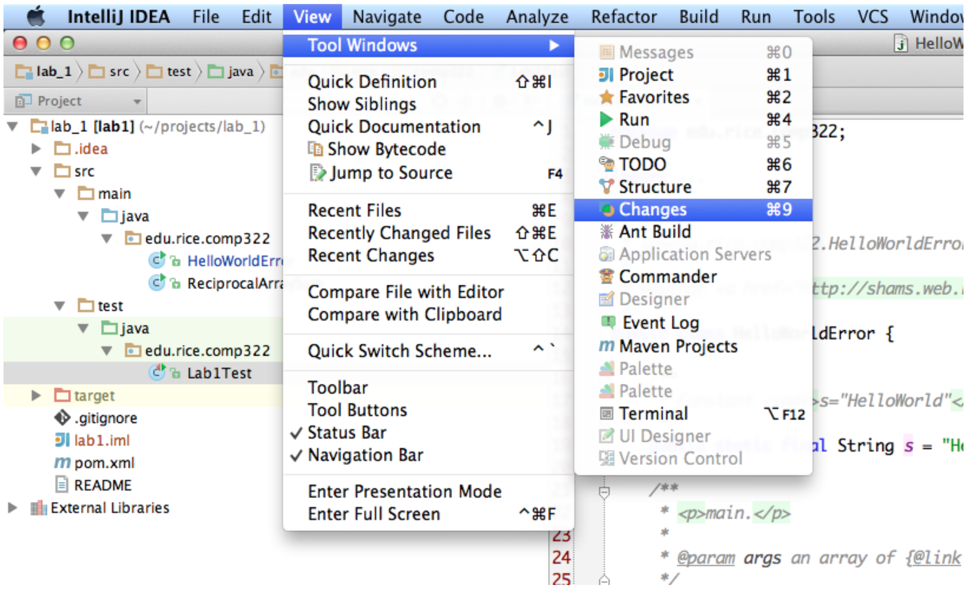
10. Open the HelloWorldError.java file by double clicking on it. Fix the error by editing the "ss" variable.


IntelliJ IDEA File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
HelloWorldError – [lab1] – lab1 – [~/proje
lab_1 src main java edu rice comp322 HelloWorldError
Project
HelloWorldError
lab_1 [lab1] (~/projects/lab_1)
.idea
src
main
java
edu.rice.comp322
HelloWorldError
ReciprocalArraySum
test
java
edu.rice.comp322
Lab1Test
target
.gitignore
lab1.iml
pom.xml
README
External Libraries
Messages Make

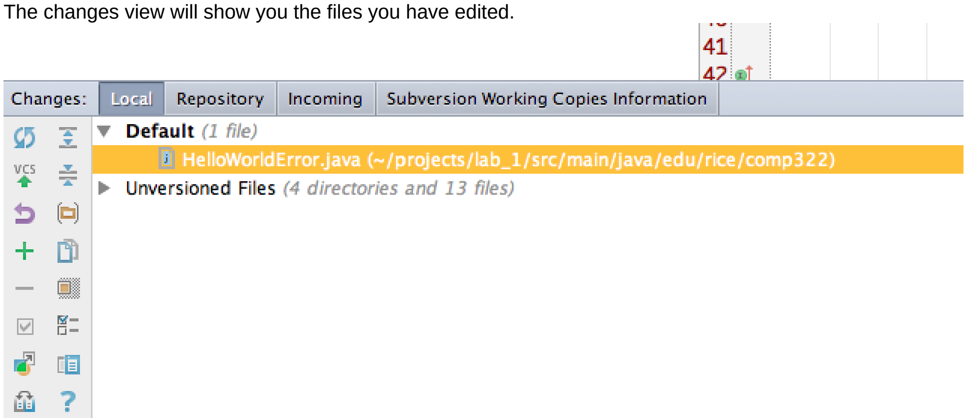
```
    public static final String s = "HelloWorld";

    /**
     * <p>main.</p>
     *
     * @param args an array of {@link String} objects.
     */
    public static void main(final String[] args) {

        launchHabaneroApp(() -> {

            finish(() -> {
                // HjRunnable is a functional interface.
                // It has one unimplemented method: void run()
                // The statement below is using lambda expression
                // Note we are avoiding the verbose syntax to cre
                final HjSuspendable aRunnable = () -> {
                    System.out.println("First: " + ss);
                };
                // We pass async an instance of HjRunnable to run
                async(aRunnable);

                // Instead we can also directly inline the lambda
                async(() -> {
                    // TODO: Fix the error by changing 'ss' to 's
                    System.out.println("Second: " + s);
                });
            });

        });
    }
}
```

```
Information: Using javac 1.8.0_05 to compile java sources
Information: java: Errors occurred while compiling module 'lab1'
Information: Module "lab1" was fully rebuilt due to project configuration/dependencies changes
Information: 1/11/15, 8:15 PM – Compilation completed with 1 error and 0 warnings in 11 sec
/Users/shamsimam/projects/lab_1/src/main/java/edu/rice/comp322/HelloWorldError.java
    Error:(36, 52)  java: cannot find symbol
                    symbol:   variable ss
                    location: class edu.rice.comp322.HelloWorldError
```

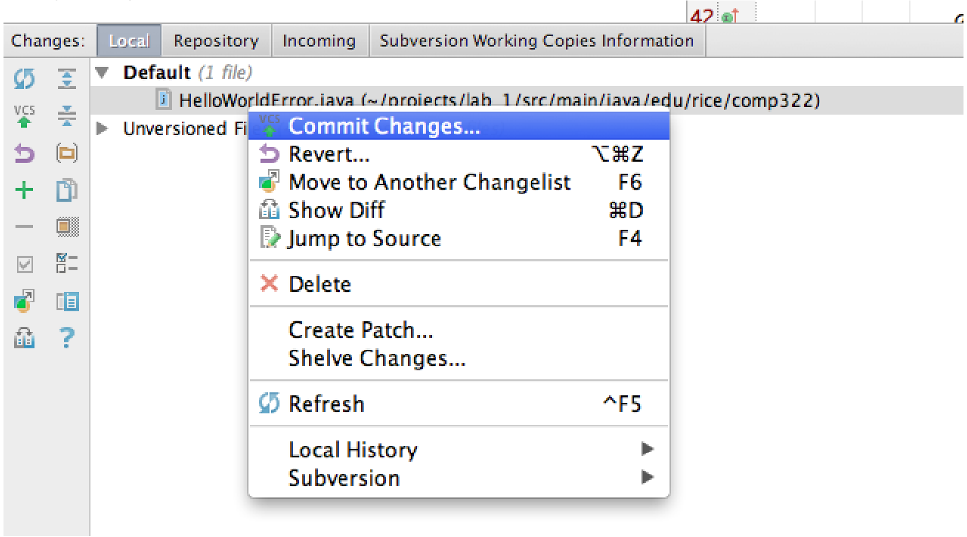
11. Open the "Changes View" so that you can see your edits.



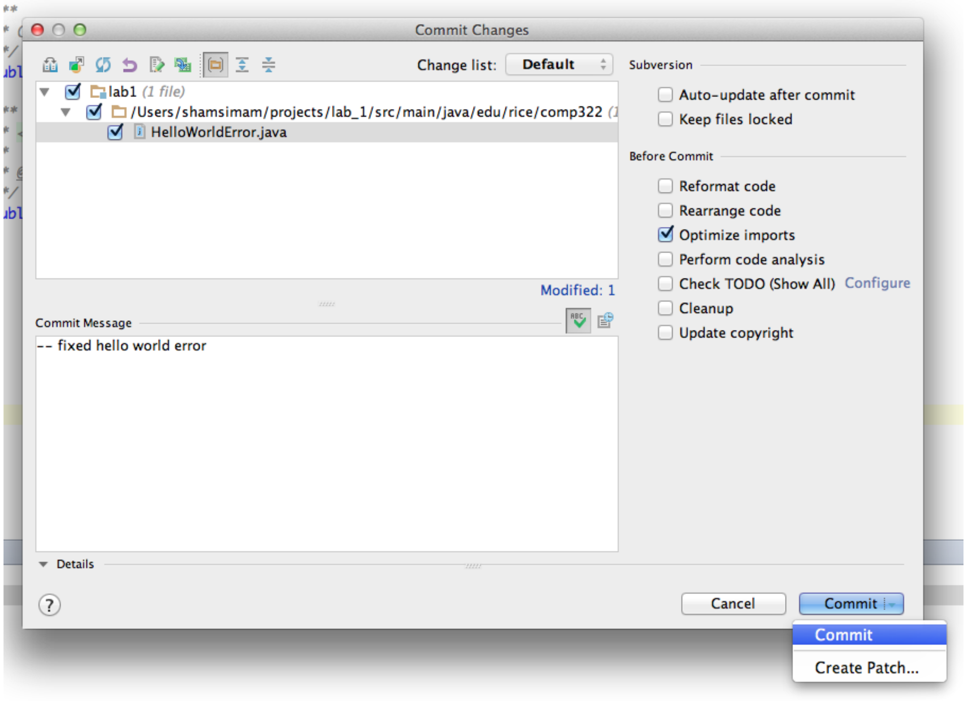
12. The changes view will show you the files you have edited.


Changes:
Local
Repository
Incoming
Subversion Working Copies Information
Default (1 file)
HelloWorldError.java (~/projects/lab_1/src/main/java/edu/rice/comp322)
Unversioned Files (4 directories and 13 files)

13. Submit your changes to the subversion repository by committing the changed files (right-clicking on the Default or HelloWorldError.java file in the Changes view).



14. Add a commit message and commit the file.

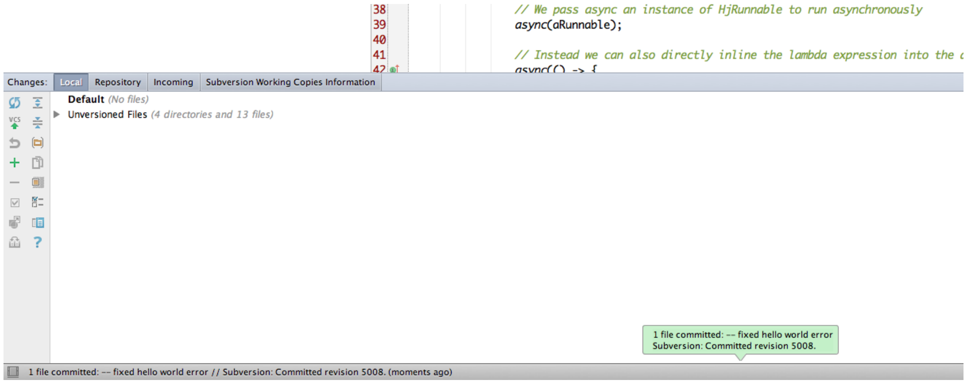


Commit Message: -- fixed hello world error

Subversion
- ☐ Auto-update after commit
- ☐ Keep files locked

Before Commit
- ☐ Reformat code
- ☐ Rearrange code
- ☑ Optimize imports
- ☐ Perform code analysis
- ☐ Check TODO (Show All)
- ☐ Cleanup
- ☐ Update copyright

15. Confirm that you see the successfully committed message to ensure that your changes have been submitted to the repository.



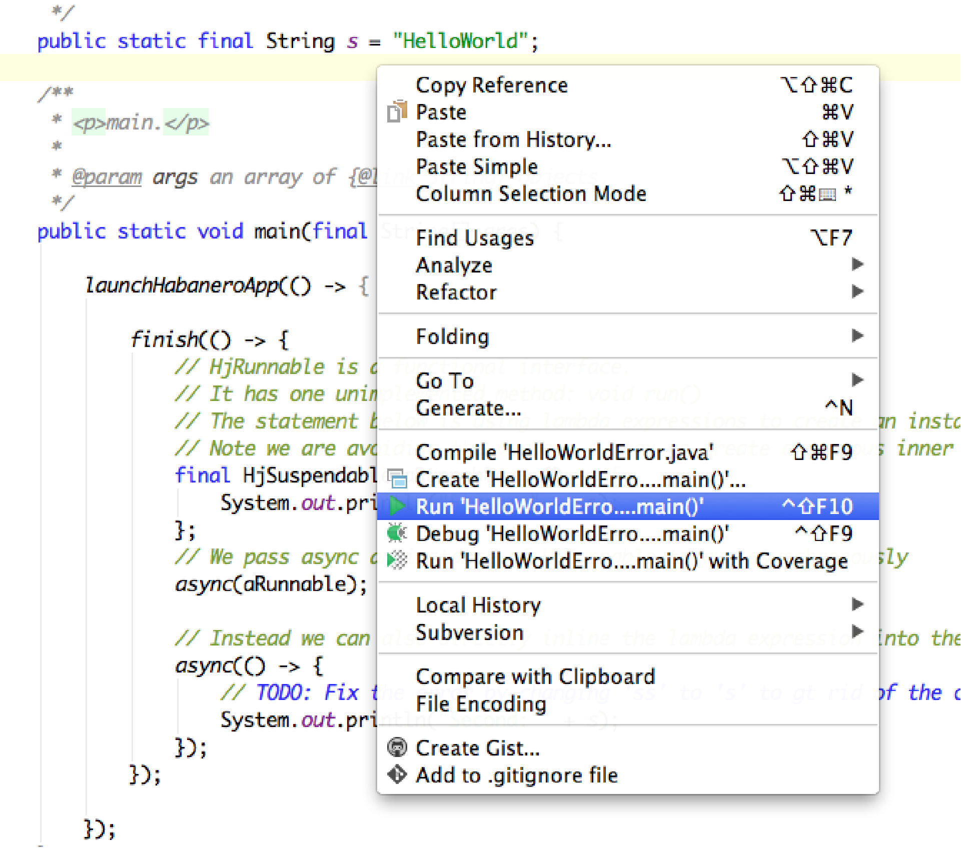
```
// We pass async an instance of HjRunnable to run asynchronously
async(aRunnable);

// Instead we can also directly inline the lambda expression into the c
async(() -> {
```

Changes:
Local
Repository
Incoming
Subversion Working Copies Information
Default (No files)
Unversioned Files (4 directories and 13 files)

1 file committed: -- fixed hello world error
Subversion: Committed revision 5008.

1 file committed: -- fixed hello world error // Subversion: Committed revision 5008. (moments ago)

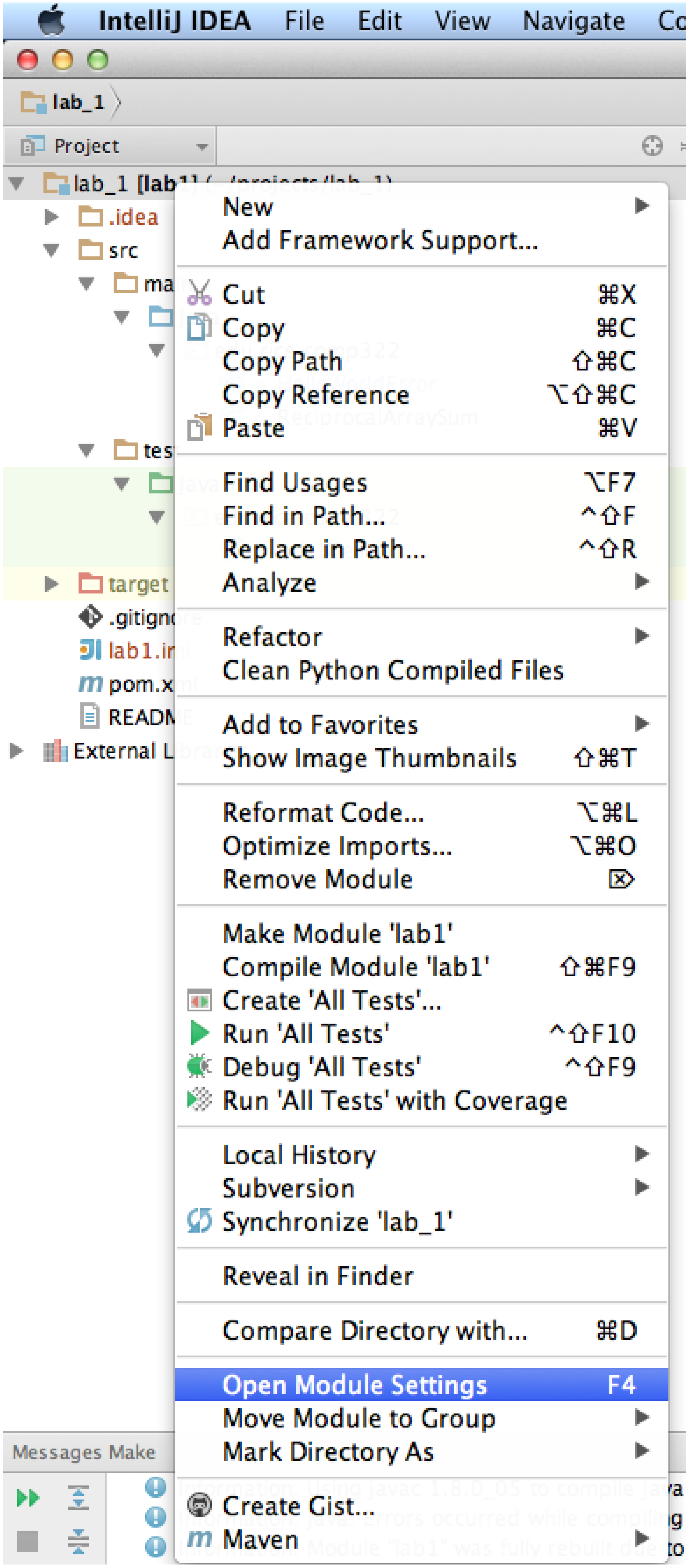
16. Attempt to run the HelloWorldError program by right-clicking on it and selecting the run option. This should fail as the javaagent option has not been set up. To configure the javaagent option we need to know the path to the HJlib jar file. Maven has already automatically downloaded this jar file for you, we can look up the path in the project.



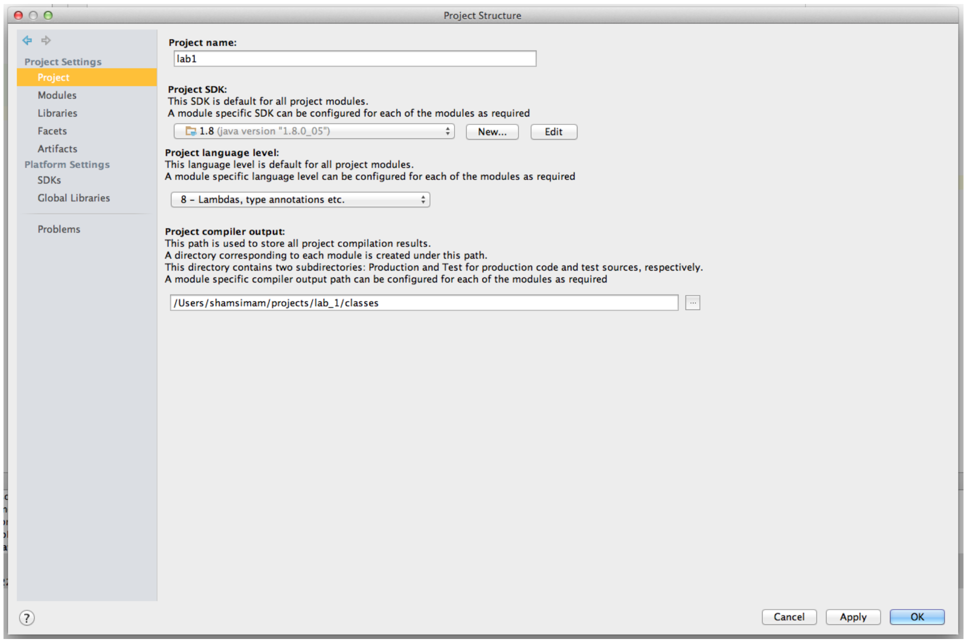
```
 */
public static final String s = "HelloWorld";

/**
 * <p>main.</p>
 *
 * @param args an array of {@l
 */
public static void main(final

    launchHabaneroApp(() -> {

        finish(() -> {
            // HjRunnable is a
            // It has one unim
            // The statement b                                           an instc
            // Note we are avo                                           s inner
            final HjSuspendabl
                System.out.pri
            };
            // We pass async a                                           sly
            async(aRunnable);

            // Instead we can                                            into the
            async(() -> {
                // TODO: Fix t                                           of the c
                System.out.pri
            });
        });

    });
```

Copy Reference ⌥⇧⌘C
Paste ⌘V
Paste from History... ⇧⌘V
Paste Simple ⌥⇧⌘V
Column Selection Mode ⇧⌘⌨ *
Find Usages ⌥F7
Analyze
Refactor
Folding
Go To
Generate... ^N
Compile 'HelloWorldError.java' ⇧⌘F9
Create 'HelloWorldErro....main()'...
Run 'HelloWorldErro....main()' ^⇧F10
Debug 'HelloWorldErro....main()' ^⇧F9
Run 'HelloWorldErro....main()' with Coverage
Local History
Subversion
Compare with Clipboard
File Encoding
Create Gist...
Add to .gitignore file

17. Open the Module Settings



18. Next ensure that the Project Language Level is set to 8.



Project name: lab1

Project SDK:
This SDK is default for all project modules.
A module specific SDK can be configured for each of the modules as required

1.8 (java version "1.8.0_05")

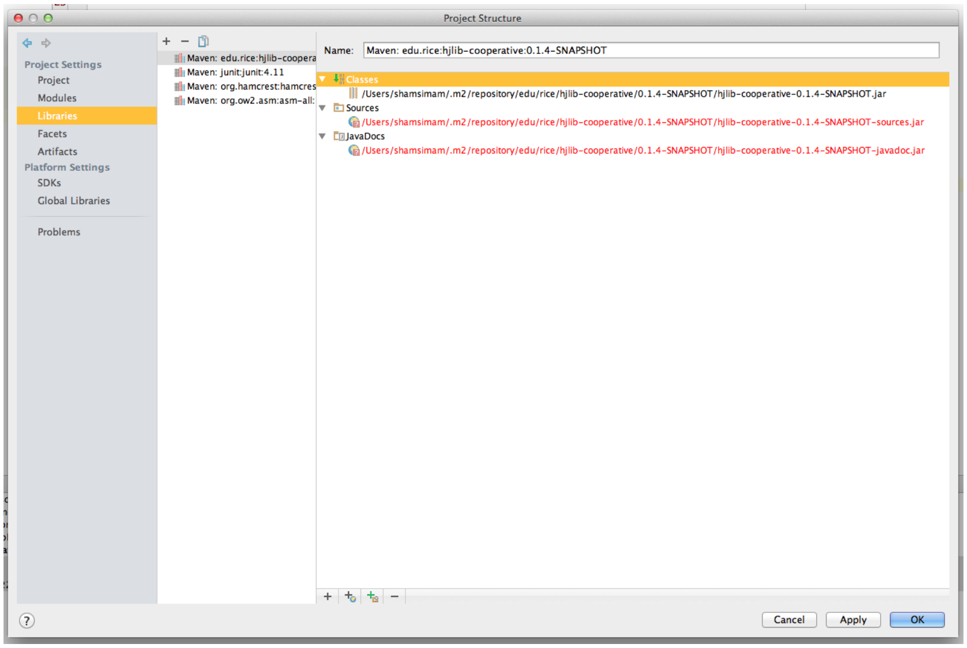
Project language level:
This language level is default for all project modules.
A module specific language level can be configured for each of the modules as required

8 - Lambdas, type annotations etc.

Project compiler output:
This path is used to store all project compilation results.
A directory corresponding to each module is created under this path.
This directory contains two subdirectories: Production and Test for production code and test sources, respectively.
A module specific compiler output path can be configured for each of the modules as required

/Users/shamsimam/projects/lab_1/classes

19. Click on the Libraries option to reveal the path for the hjlib jar file. On my machine it is located at `/Users/shamsimam/.m2/repository/edu/rice/hjlib-cooperative/0.1.4-SNAPSHOT/hjlib-cooperative-0.1.4-SNAPSHOT.jar`

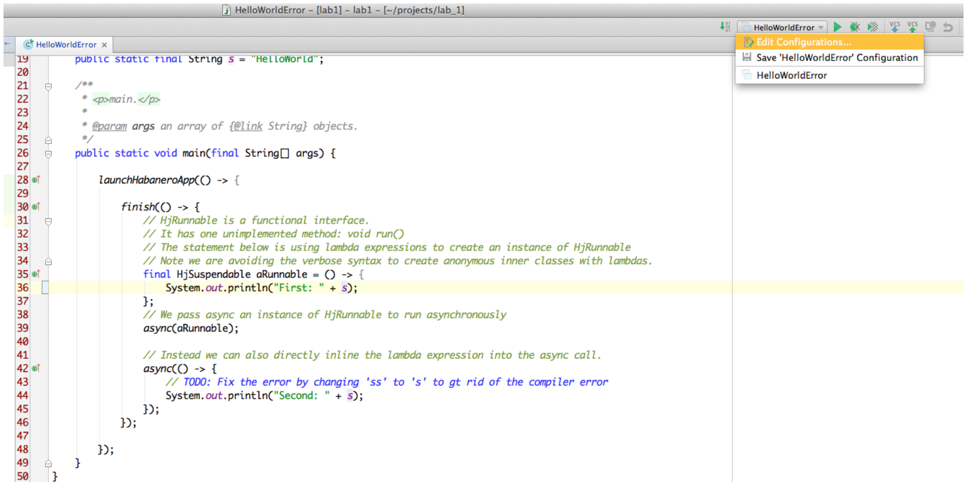


Maven: edu.rice:hjlib-coopera
Maven: junit:junit:4.11
Maven: org.hamcrest:hamcres
Maven: org.ow2.asm:asm-all:

Name: Maven: edu.rice:hjlib-cooperative:0.1.4-SNAPSHOT

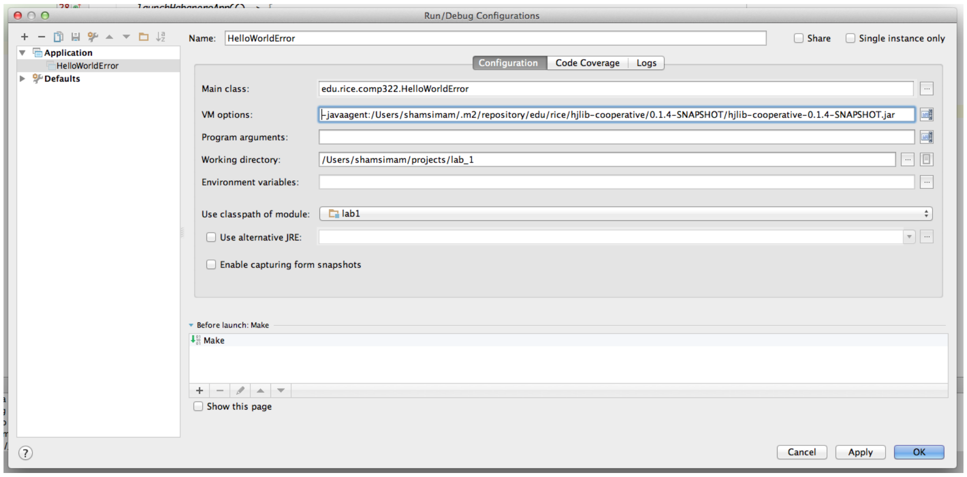
Classes
/Users/shamsimam/.m2/repository/edu/rice/hjlib-cooperative/0.1.4-SNAPSHOT/hjlib-cooperative-0.1.4-SNAPSHOT.jar
Sources
/Users/shamsimam/.m2/repository/edu/rice/hjlib-cooperative/0.1.4-SNAPSHOT/hjlib-cooperative-0.1.4-SNAPSHOT-sources.jar
JavaDocs
/Users/shamsimam/.m2/repository/edu/rice/hjlib-cooperative/0.1.4-SNAPSHOT/hjlib-cooperative-0.1.4-SNAPSHOT-javadoc.jar

20. Now that we know the path, we are ready to edit the run configuration and insert the javaagent option.


HelloWorldError – [lab1] – lab1 – [~/projects/lab_1]
HelloWorldError
Edit Configurations...
Save 'HelloWorldError' Configuration
HelloWorldError

```
    public static final String s = "HelloWorld";

    /**
     * <p>main.</p>
     *
     * @param args an array of {@link String} objects.
     */
    public static void main(final String[] args) {

        launchHabaneroApp(() -> {

            finish(() -> {
                // HjRunnable is a functional interface.
                // It has one unimplemented method: void run()
                // The statement below is using lambda expressions to create an instance of HjRunnable
                // Note we are avoiding the verbose syntax to create anonymous inner classes with lambdas.
                final HjSuspendable aRunnable = () -> {
                    System.out.println("First: " + s);
                };
                // We pass async an instance of HjRunnable to run asynchronously
                async(aRunnable);

                // Instead we can also directly inline the lambda expression into the async call.
                async(() -> {
                    // TODO: Fix the error by changing 'ss' to 's' to gt rid of the compiler error
                    System.out.println("Second: " + s);
                });
            });

        });
    }
}
```

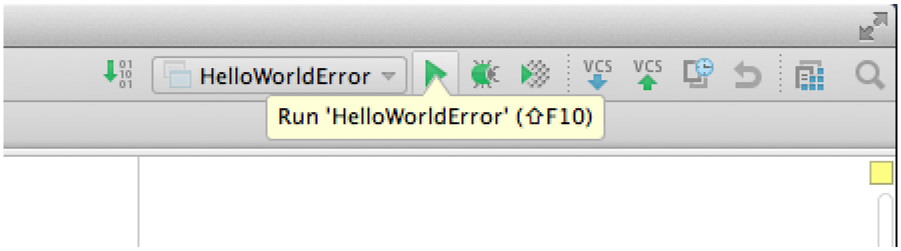
21. Under the VM Options textfield insert `-javaagent:/Users/shamsimam/.m2/repository/edu/rice/hjlib-cooperative/0.1.4-SNAPSHOT/hjlib-cooperative-0.1.4-SNAPSHOT.jar` text and click OK to save the configuration.
22.

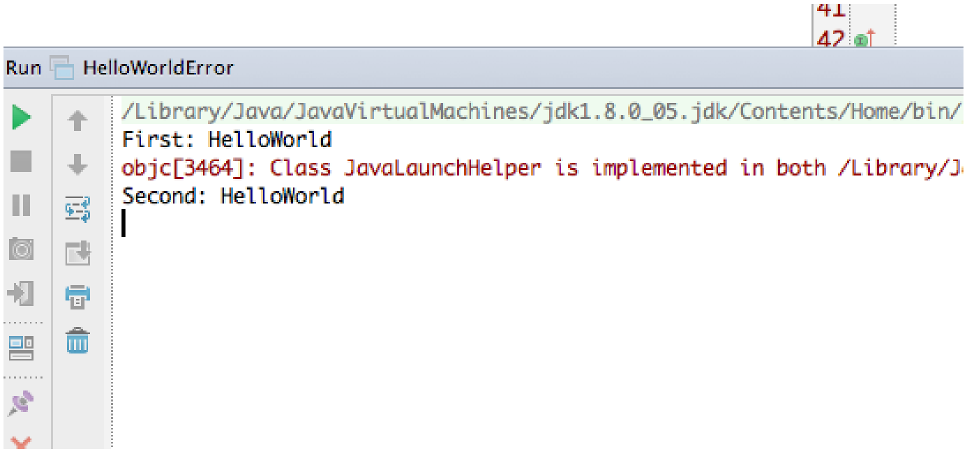


Name: HelloWorldError
□ Share □ Single instance only

Main class: edu.rice.comp322.HelloWorldError
VM options: -javaagent:/Users/shamsimam/.m2/repository/edu/rice/hjlib-cooperative/0.1.4-SNAPSHOT/hjlib-cooperative-0.1.4-SNAPSHOT.jar
Program arguments:
Working directory: /Users/shamsimam/projects/lab_1
Environment variables:
Use classpath of module: lab1
□ Use alternative JRE:
□ Enable capturing form snapshots
□ Show this page

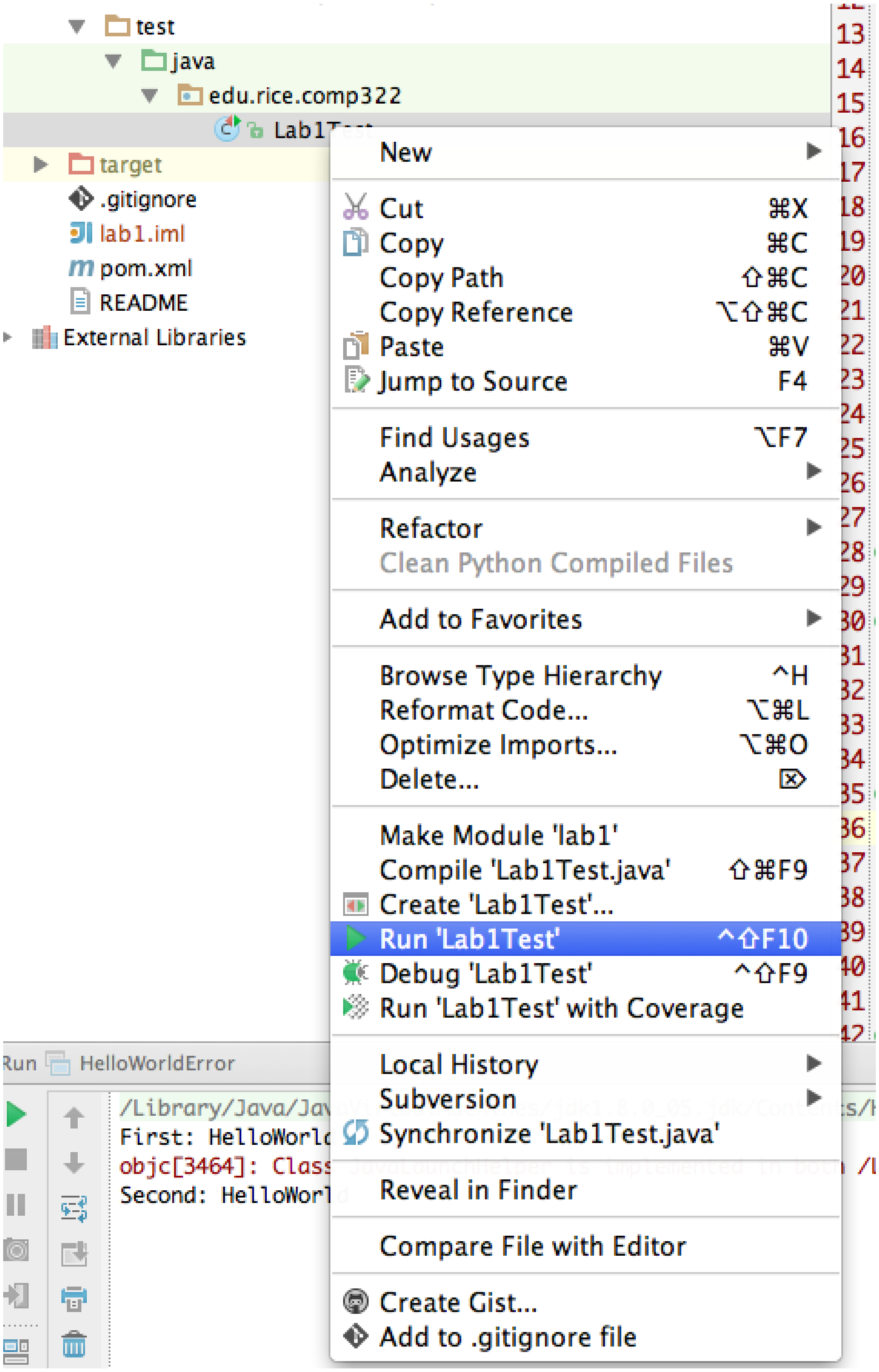
23. Now, click on the green play button to run the HelloWorldError.java file wiht the correct VM configurations. The class should be run successfully and you should see the output as below.





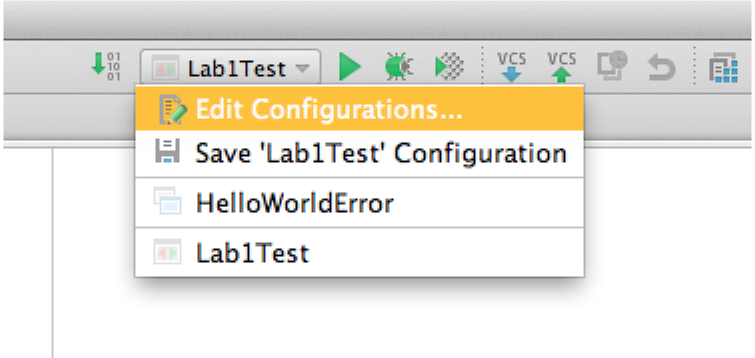
```
/Library/Java/JavaVirtualMachines/jdk1.8.0_05.jdk/Contents/Home/bin/
First: HelloWorld
objc[3464]: Class JavaLaunchHelper is implemented in both /Library/J
Second: HelloWorld
```

24. First attempt to run the unit test by right-clicking on the file and selecting Run. This will fail as the javaagent flag has not been set up.

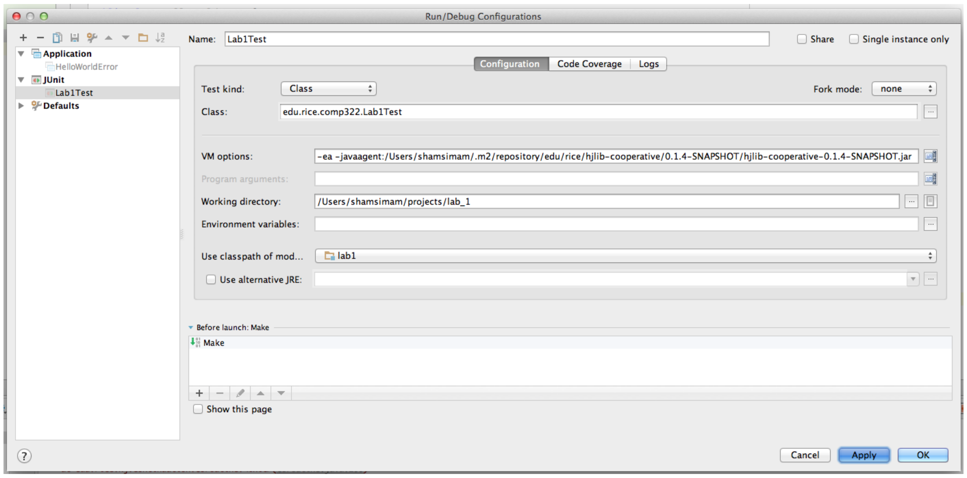


```
/Library/Java/Ja
First: HelloWorld
objc[3464]: Class
Second: HelloWorld
```

25. Prepare to edit the configurations for the unit test.



26. Like in Step 20, ensure that the javaagent flag has been correct passed as a VM option to run the Unit test.

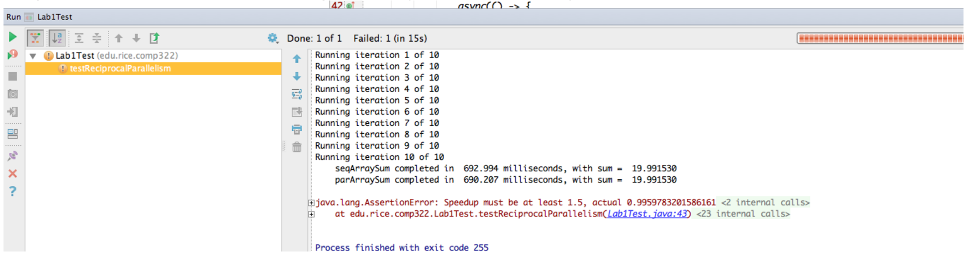


Name: Lab1Test
□ Share
□ Single instance only

Test kind: Class
Fork mode: none
Class: edu.rice.comp322.Lab1Test
VM options: -ea -javaagent:/Users/shamsimam/.m2/repository/edu/rice/hjlib-cooperative/0.1.4-SNAPSHOT/hjlib-cooperative-0.1.4-SNAPSHOT.jar
Program arguments:
Working directory: /Users/shamsimam/projects/lab_1
Environment variables:
Use classpath of mod... lab1
□ Use alternative JRE:

□ Show this page

27. Note the red bar displaying that the Unit test has failed. After you successfully complete the second part of the lab by correctly parallelizing the reciprocal sum, this unit test should pass and you should see a green bar.



Done: 1 of 1   Failed: 1 (in 15s)

```
Running iteration 1 of 10
Running iteration 2 of 10
Running iteration 3 of 10
Running iteration 4 of 10
Running iteration 5 of 10
Running iteration 6 of 10
Running iteration 7 of 10
Running iteration 8 of 10
Running iteration 9 of 10
Running iteration 10 of 10
    seqArraySum completed in  692.994 milliseconds, with sum =  19.991530
    parArraySum completed in  690.207 milliseconds, with sum =  19.991530

java.lang.AssertionError: Speedup must be at least 1.5, actual 0.9959783201586161 <2 internal calls>
    at edu.rice.comp322.Lab1Test.testReciprocalParallelism(Lab1Test.java:43) <23 internal calls>


Process finished with exit code 255
```

Ask your instructor or TA if you run into any issues.