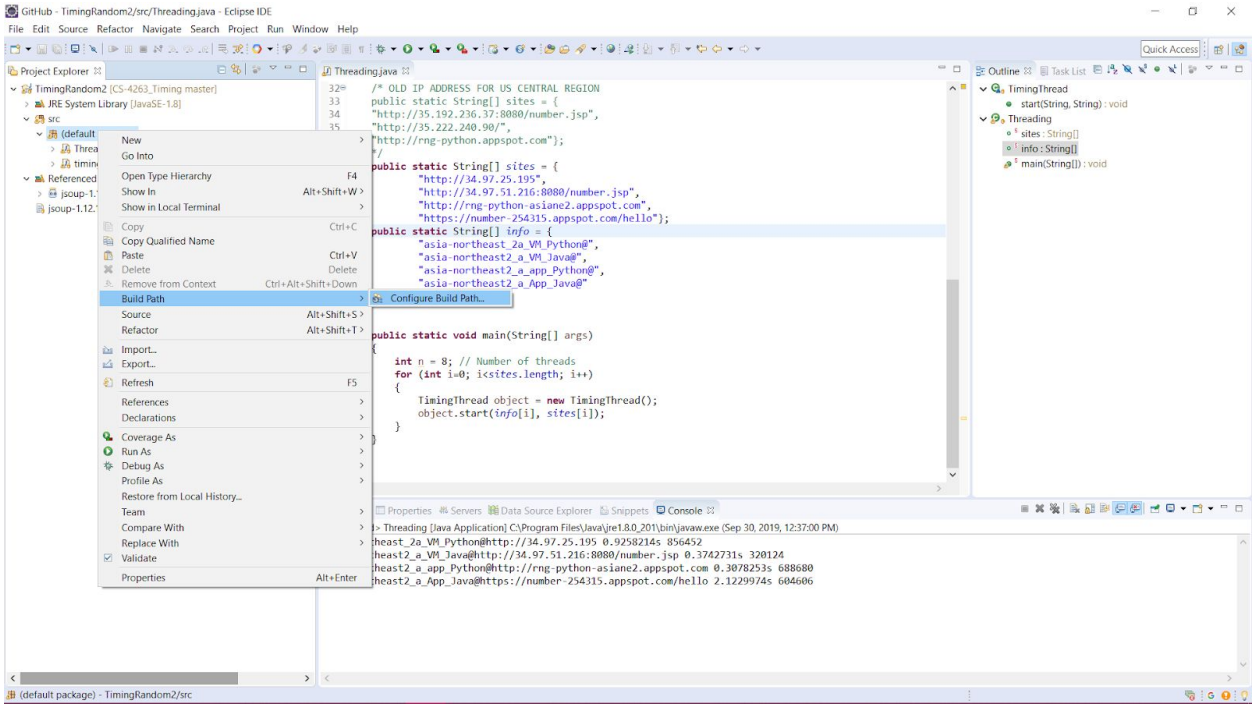
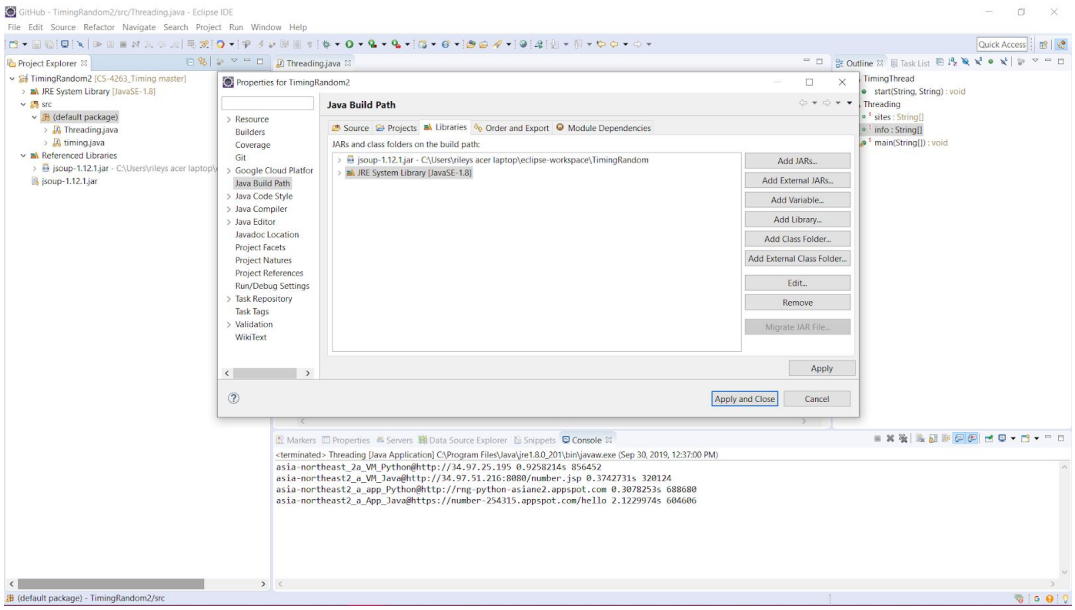
Github Link: https://github.com/coolstones/CS-4263_Timing

Installation Instructions:

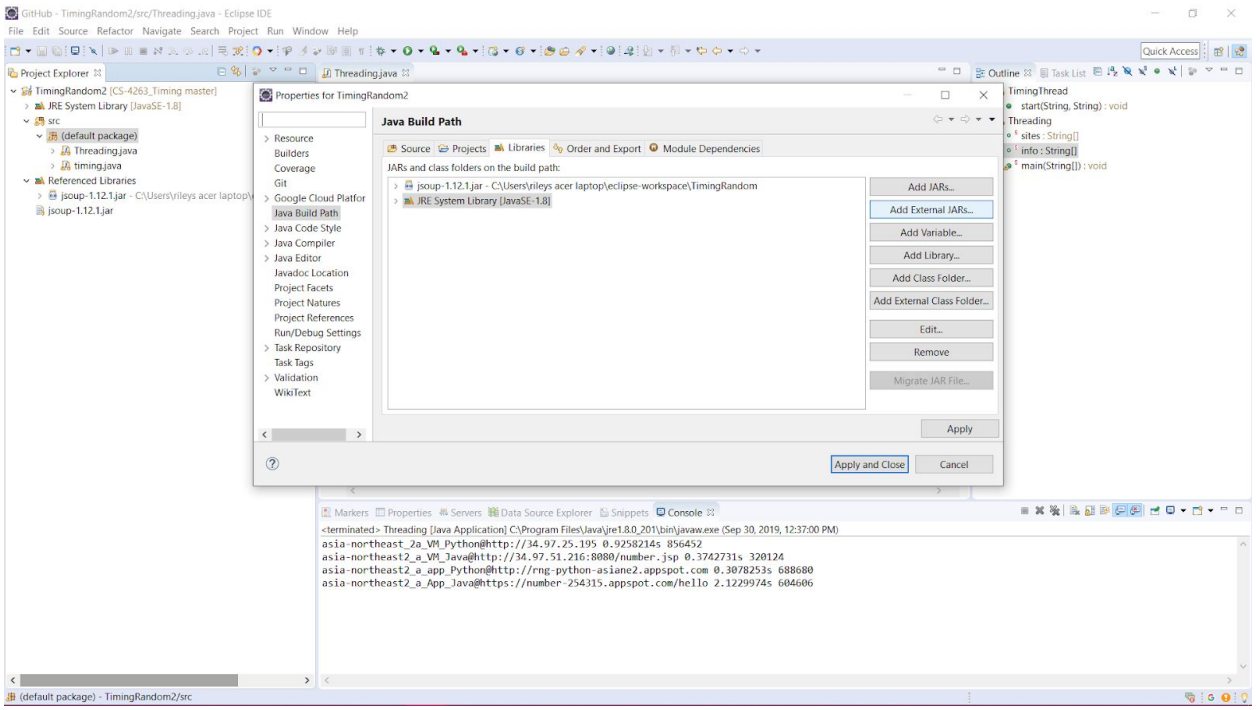
When your IDE first starts, right click the Default Package within the project and src folder. Navigate to Build Path and then click Configure Build Path.



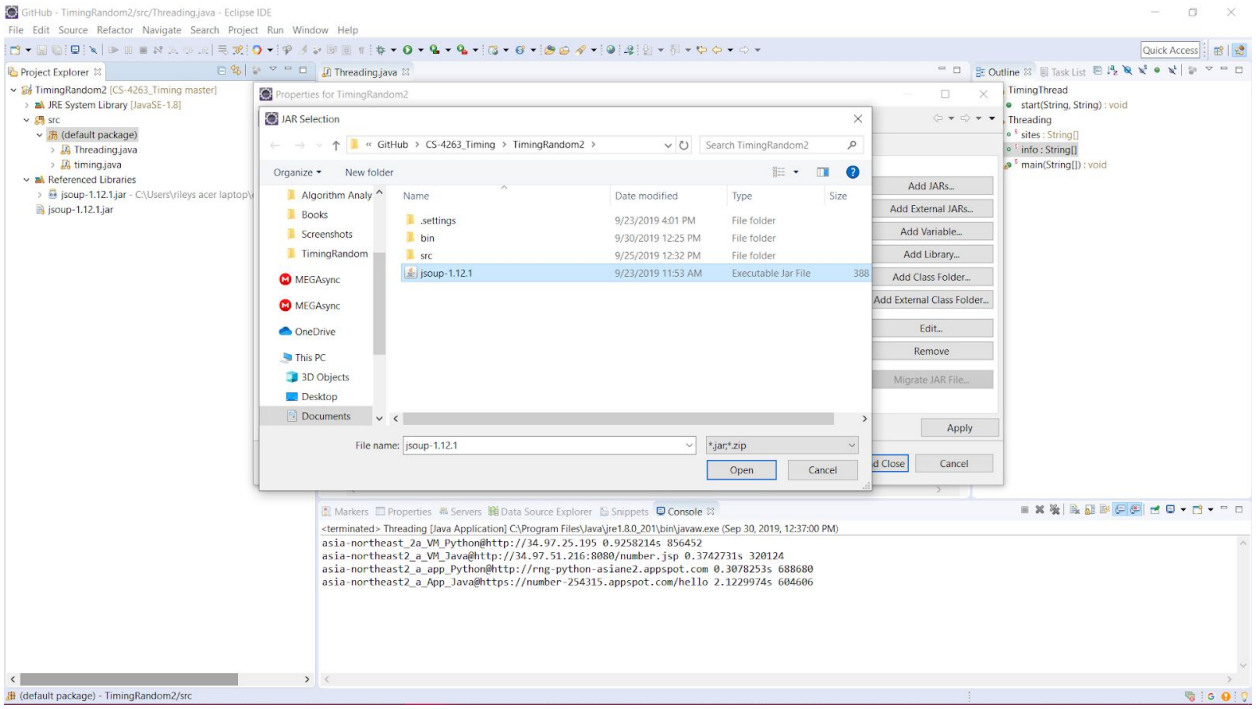
When the new window opens, click on the tab marked Libraries.



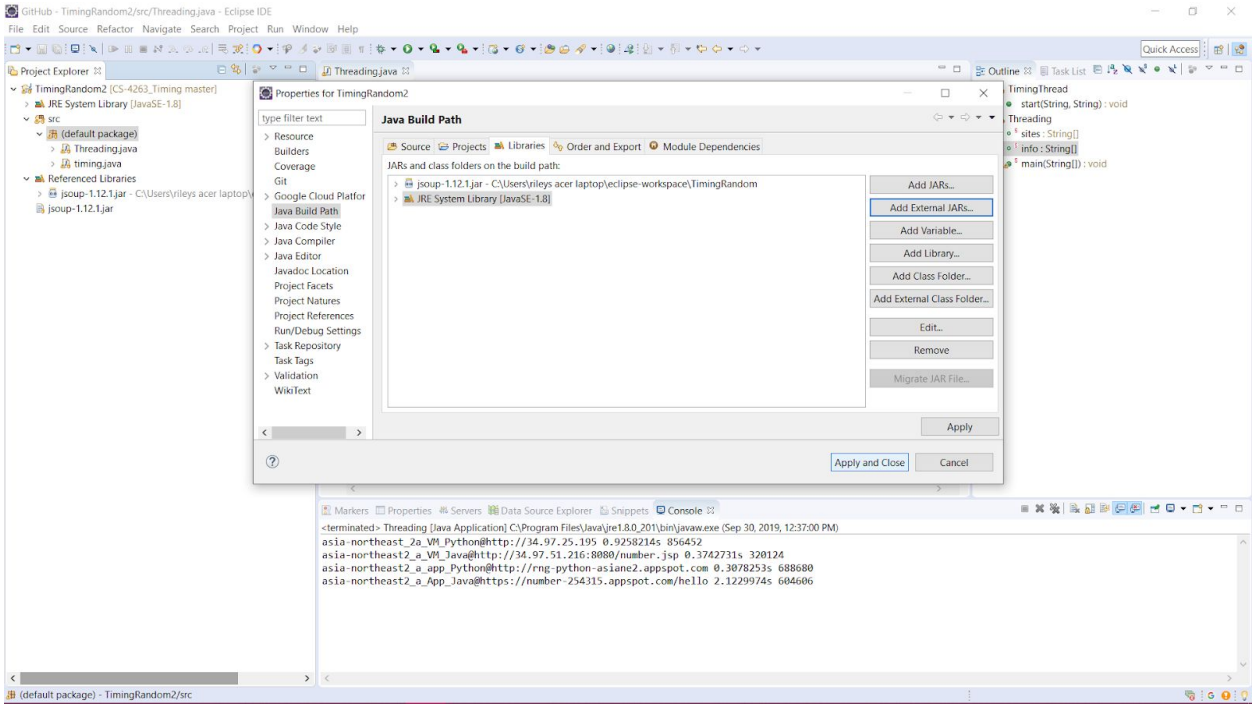
Then click Add External JARs.



When the file explorer opens click on jsoup-1.12.1. This file should be in the main project folder.

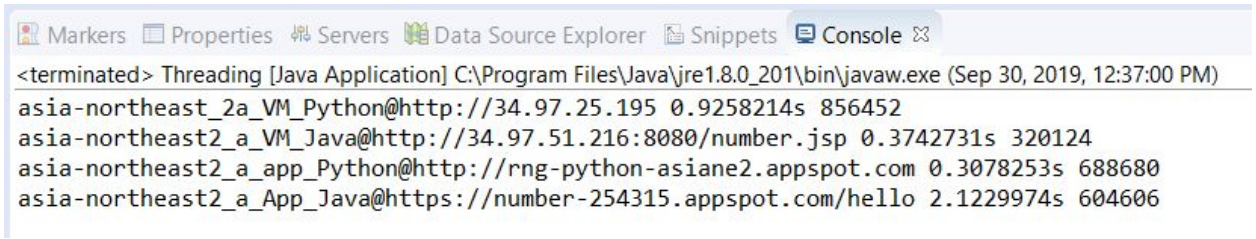


Once it is added to the list of libraries, then click Apply and Close. This should resolve the dependency-based errors with the program.



You can then change the IP Addresses and Site info by modifying the strings within the data arrays of “sites” and “info”

Example output:

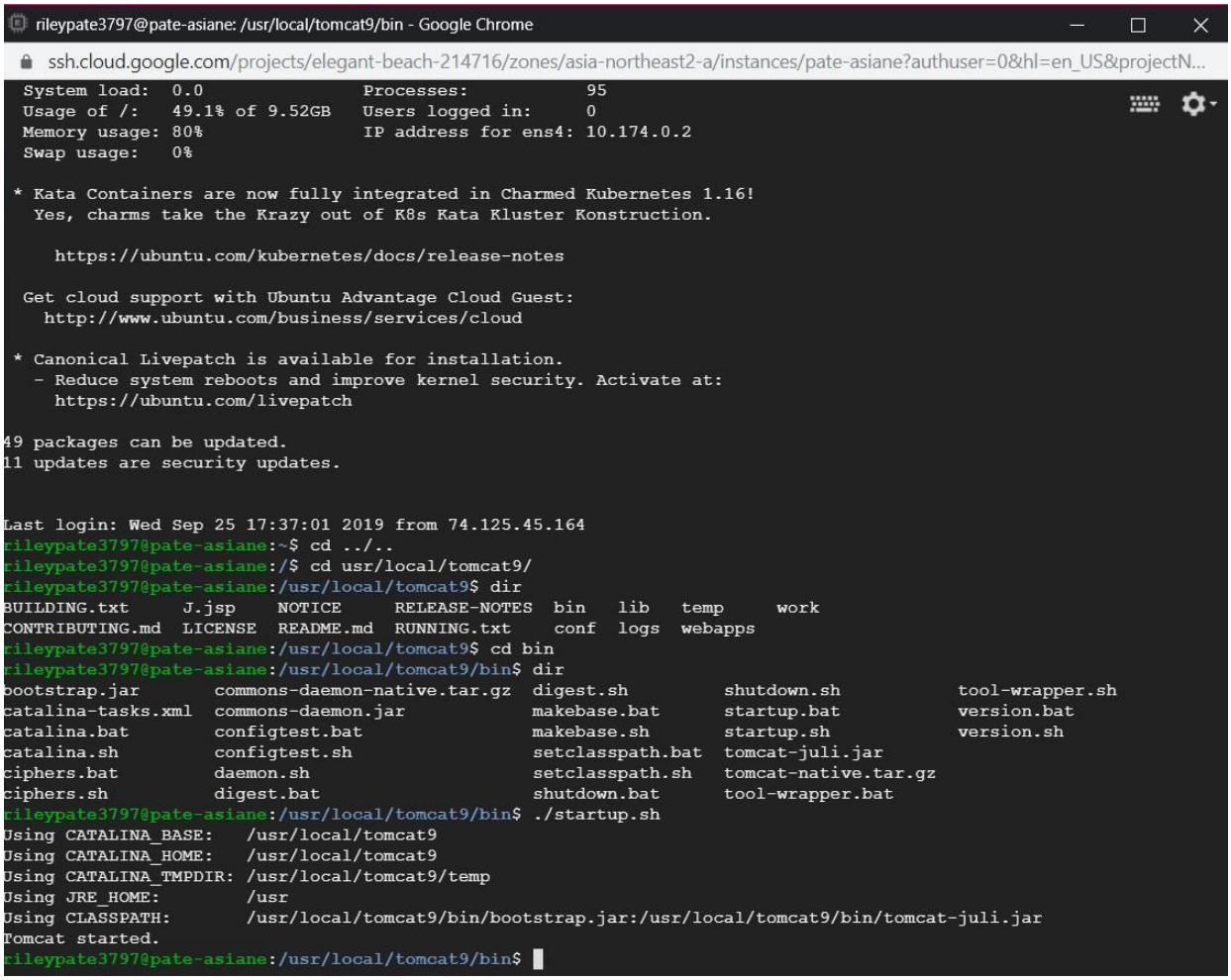


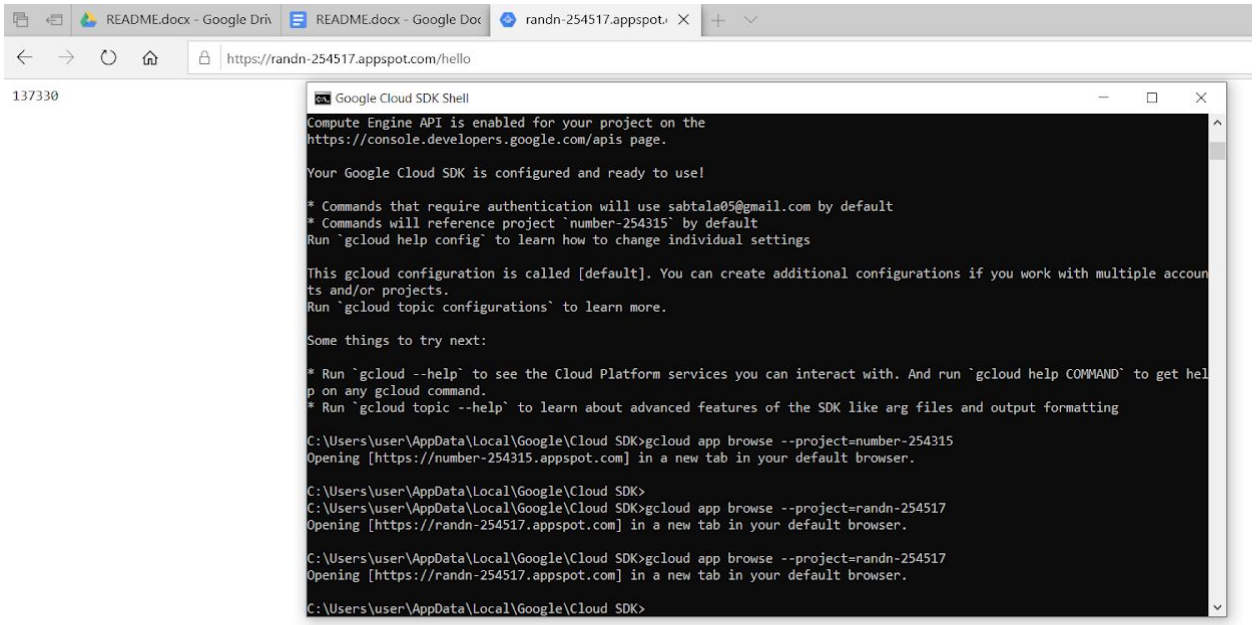
This project required us to move our VM instances to a different region, Asia in this case.

A VM can be cloned by creating an image of the drive and following the normal creation page, changing the region, and the hard disk image.

We then had to start our webservers to be accessed externally. This is an example of us starting a tomcat server on our virtual machine. This allowed us to host file to be accessed simply by a link like this one: http://34.97.51.216:8080/number.jsp

Each server will have different output as it runs depending on the framework used, but they may look something like the following:





Step by step implementation of Java App Engine application

Before you start, you need to install Eclipse IDE for Java EE Developers

---

Step1: Go to GCP and create a new App Engine project

Step2: Configure your Eclipse IDE. What you need to do is to install Google Cloud Tools for Eclipse from the Eclipse Marketplace.

After completing the installation, go ahead and restart your Eclipse IDE.

Step3: Now go to Eclipse IDE and you should be able to create an App Engine Java Project. For this assignment, I have created a Google App Engine Standard Java Project.

Click File>New>Other>Google App Engine Standard Java Project

Enter a name for your project and leave the other fields as their defaults.

Now click >Finish

You have created your project!!!

Step4: Go back to GCP and copy the project id of the project where you want your application to be deployed into.

Step5: Go back to eclipse, go to appengine-web.xml file and paste <application>?</application>

“?” is your Project id.

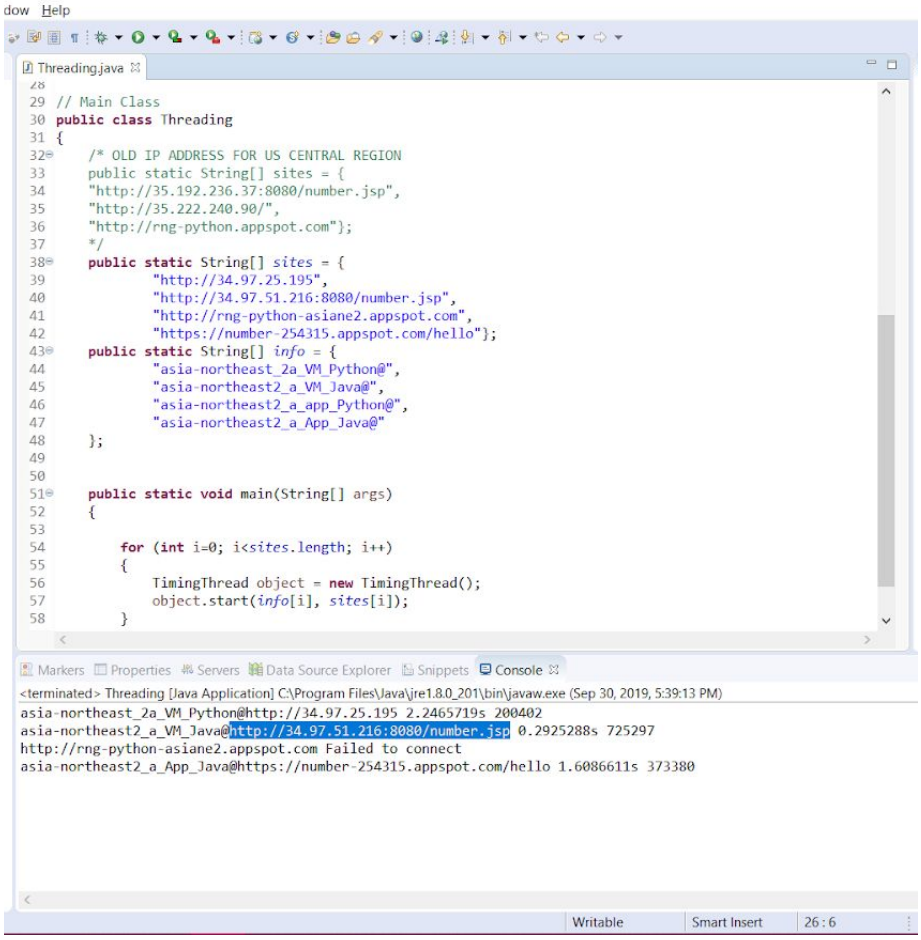
Step6: Now save your application. You are now ready to deploy your application.

Right click the project in the Package Explorer and select Deploy to App Engine Standard. In the account box, click Deploy.

The status of the deployment appears in the Eclipse Console. Once the deployment is completed, a browser window opens and displays the output of your Servlet.

Your application is now running on Google Cloud!!!!!

By modifying our previous project, we were able to add in information about each web server as well as clean up the resulting output. By performing these changes we were able to access each server while showing the number of seconds it took to access:



Another Screenshot with added data points to show scalability: