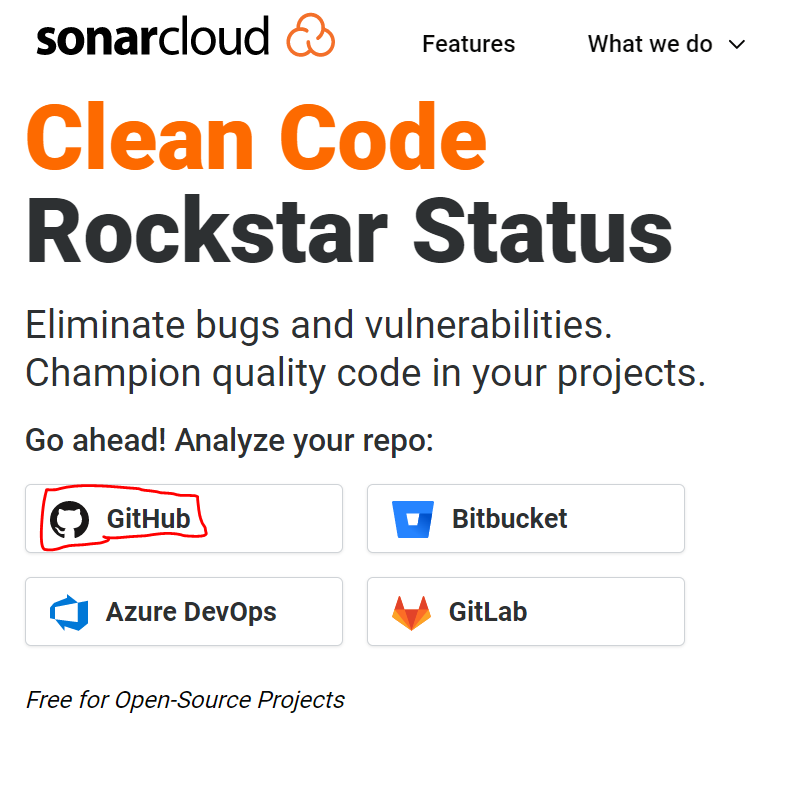
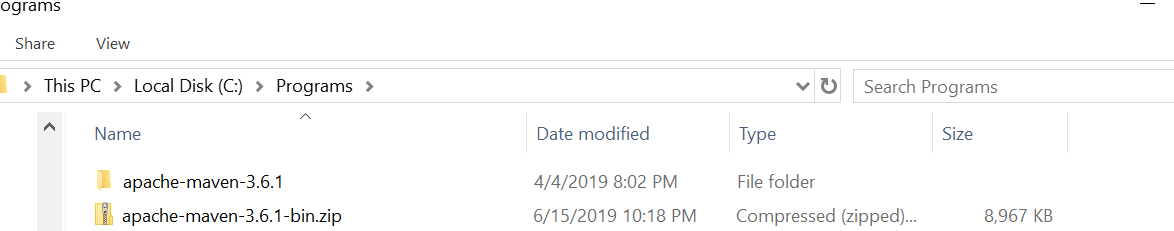
To avoid losing time in class you can ask them to download all prerequisites and initial setup programs/repo following the README.md at <https://github.com/mperales90/CQAcademy>

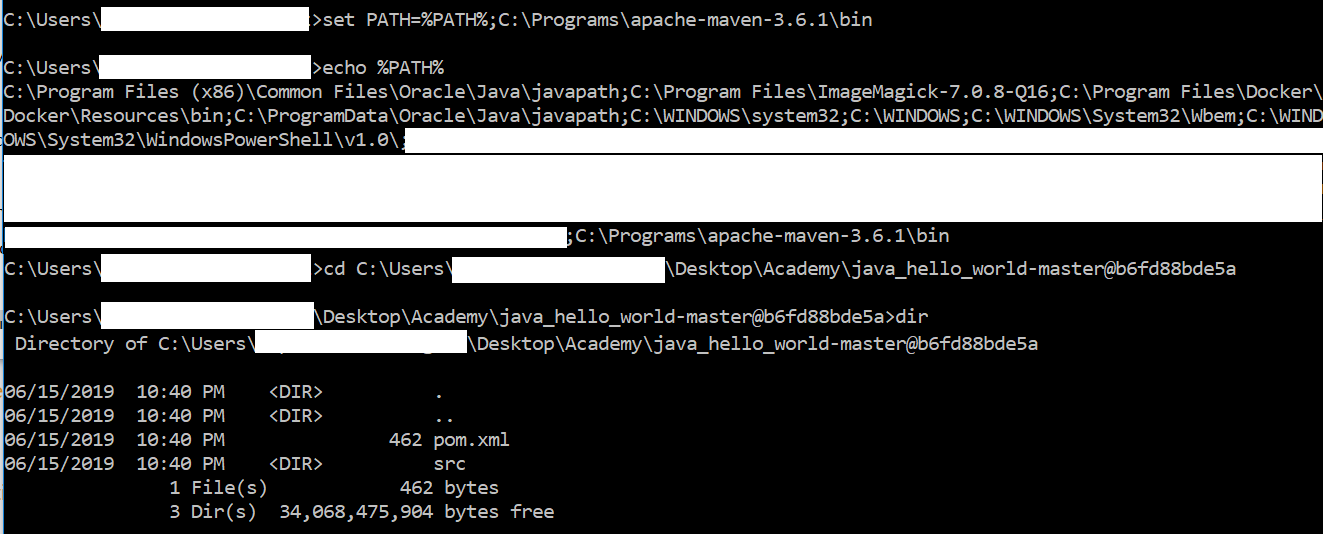
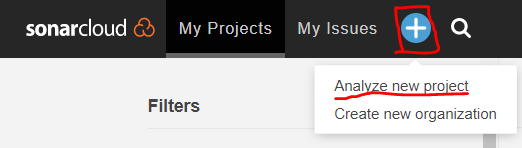
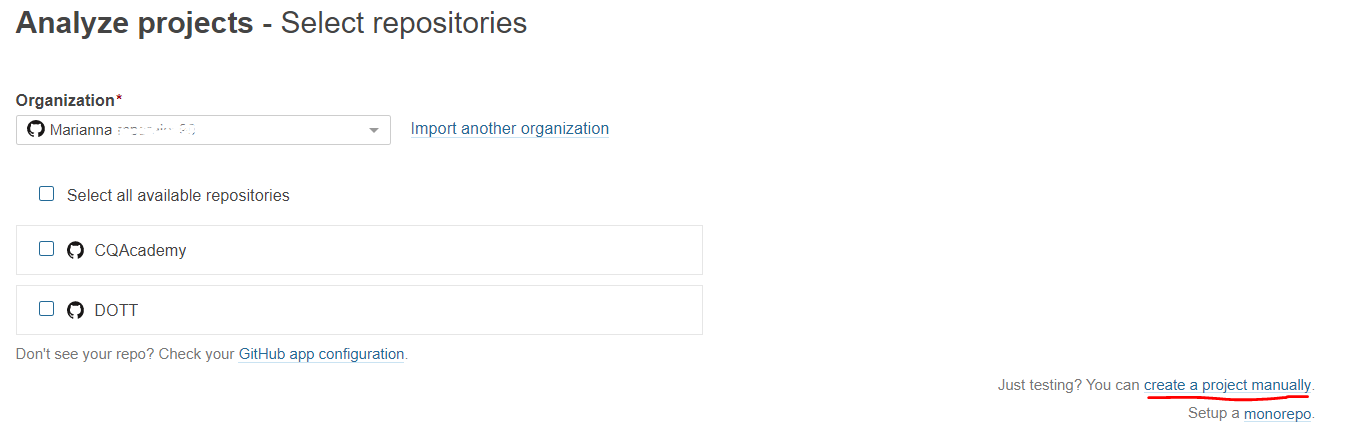
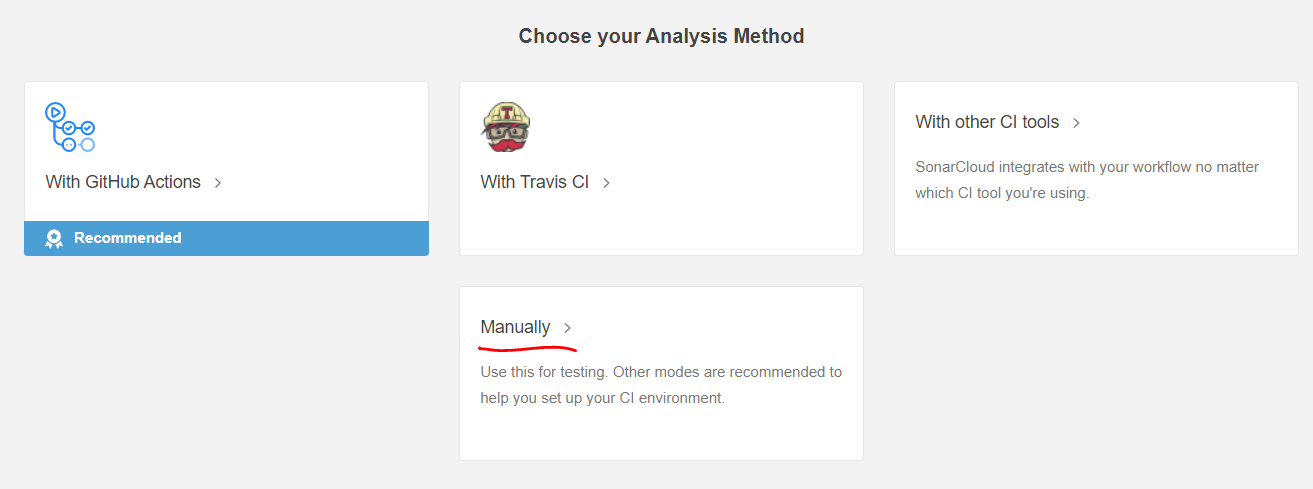
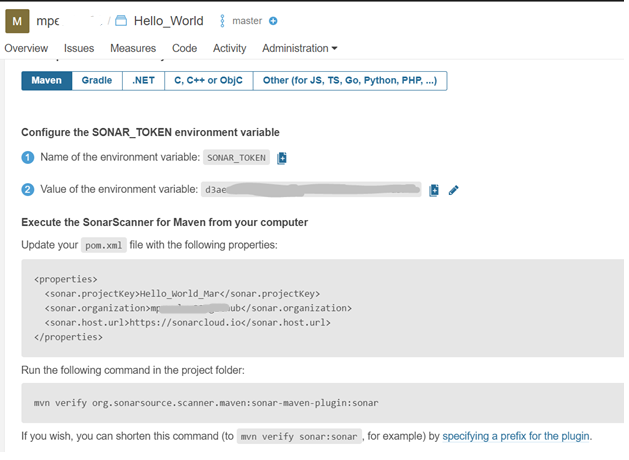
SonarQube Demo

In this demo we will show how we can manually execute a SonarQube scan. There are many ways to scan code with SonarQube depending on the environment and the programming language, in this demo we will focus on scanning Java code by using Maven.

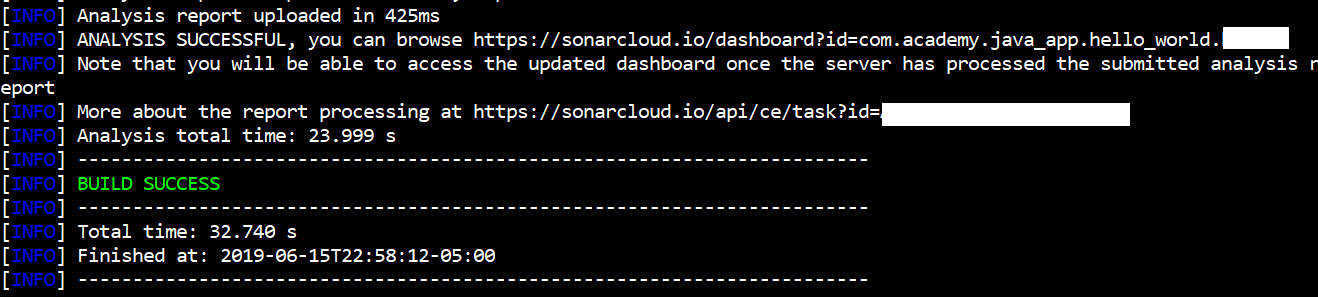
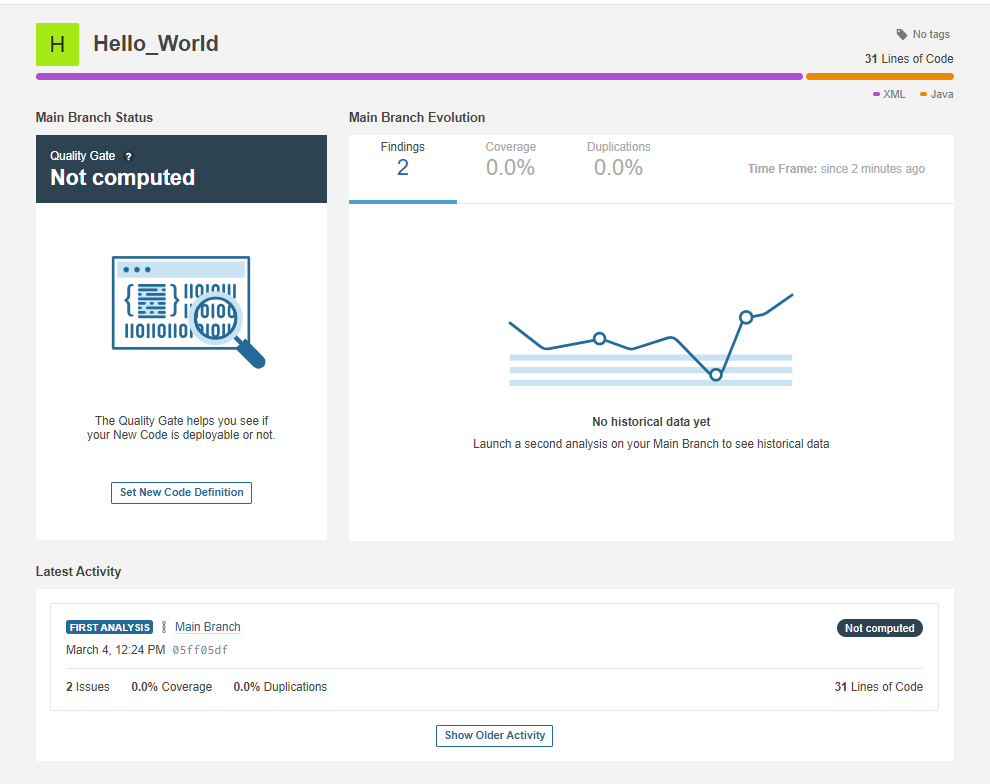
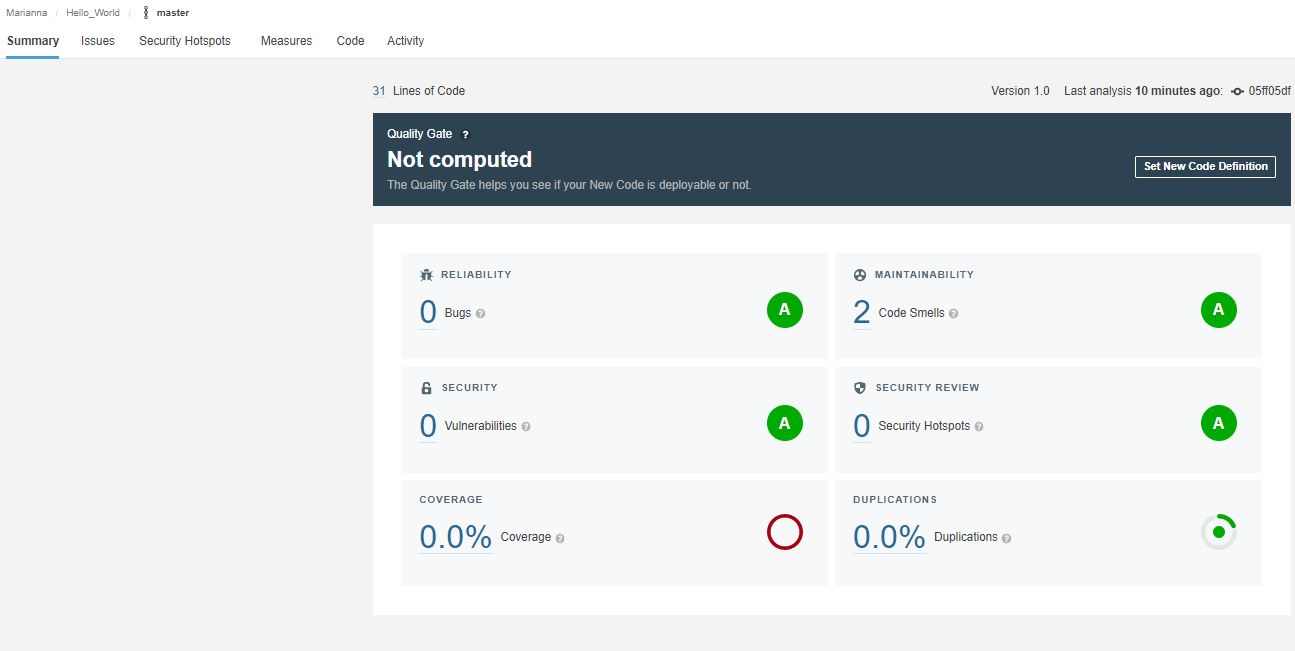
# Initial setup

1. Create a Github account (if not done already).
   1. <https://github.com/>
2. Login to SonarQube Online site using your Github account.
   1. [https://sonarcloud.io](https://sonarcloud.io/)
   2. 
3. Download Apache Maven.
   1. <https://maven.apache.org/download.cgi>
4. Unzip Maven in a directory where you want to install it. For example C:\Programs
   1. 
5. Add Maven to your environment variables.
   1. If you don’t have admin access, run the following command in the command prompt.
      1. For command prompt
         1. set PATH=%PATH%;C:\Programs\apache-maven-3.8.4\bin
      2. For bash
         1. PATH=$PATH:/c/Programs/apache-maven-3.8.4/bin
      3. Note that this command only works for the session of the command prompt in which you ran it. If you close it you will need to run it again.
      4. Type ‘mvn -version’ (cmd) or ‘mvn --version’ (bash) to confirm the environment variable was correctly set.

# Running the SonarQube scan

1. Download any code you want to scan.
   1. You can use the following code as an example: <https://github.com/mperales90/CQAcademy/tree/master/Java_Hello_World>
2. Navigate to the location where you downloaded the code in command prompt.
   1. 
3. Now go to sonarcloud.io website. Click on the + button and Analyze New Project.
   1. 
4. Click on create manually at the bottom right.
5. 
6. Give a Project Key name.
   1. For example you can use: com.academy.java\_app.hello\_world.<your\_name>
7. Give a Display name.
   1. For example you can use: Hello\_World
8. By default we need to select public, as we are using SonarCloud free tier.
9. Click on Set Up.
10. On the Analyze your project section, select ‘Manually’
    1. 
11. Select Maven as the option that best describes your build.
12. Add the SONAR\_TOKEN env variable:
    1. set SONAR\_TOKEN=<your token>
    2. 
13. Open the pom.xml file in the Java\_Hello\_World repo and add the properties tag (make sure tab spaces are respected!)

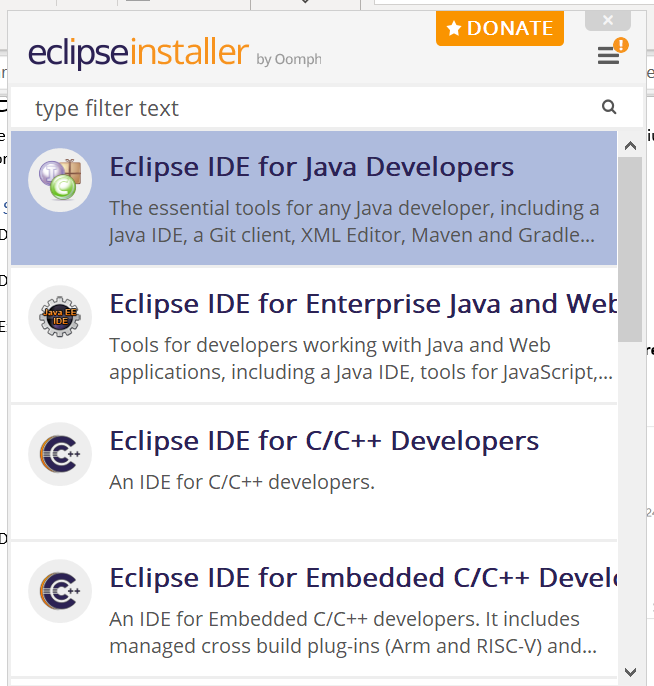
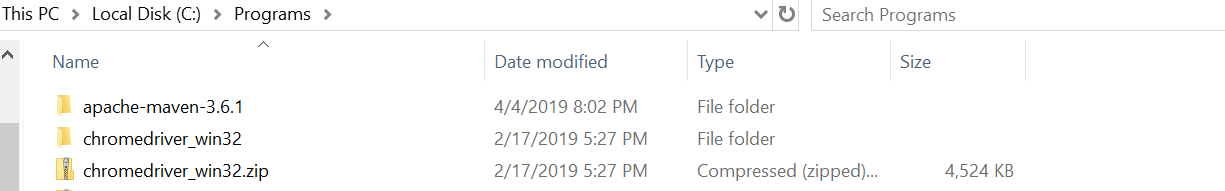
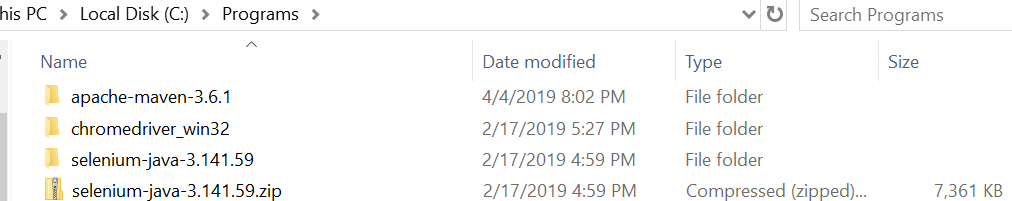


1. Save the file and on cmd/bash type ‘mvn verify org.sonarsource.scanner.maven:sonar-maven-plugin:sonar
2. Wait for the build and scan to complete. It will provide you a url at the end, navigate to it.
   1. 
3. You can see the results of your scan. You can click on the issues reported and see the details.
   1. For example you can click on the Code Smells.
   2. 
   3. 
4. Explore other types of metrics and explain to the class.

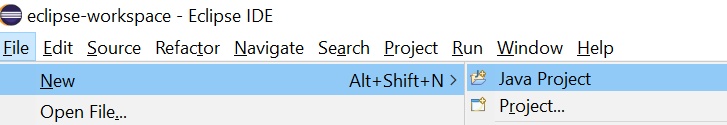
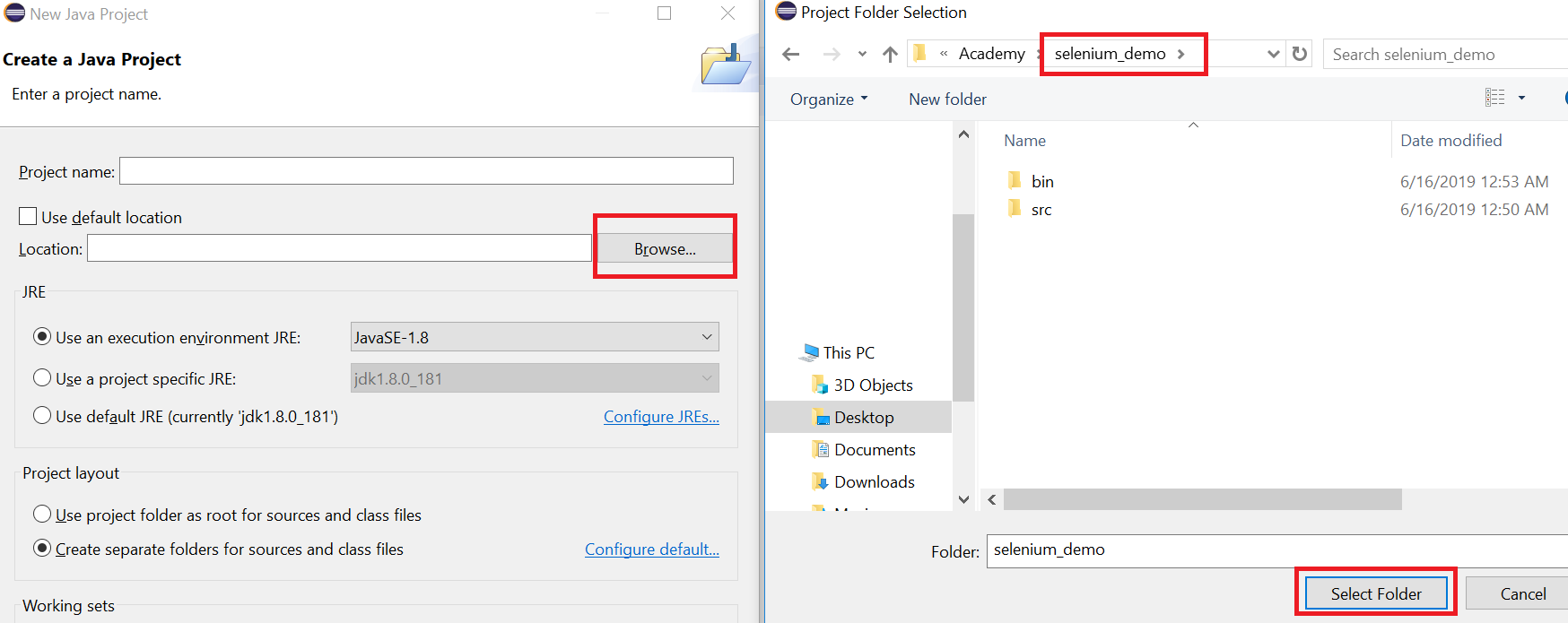
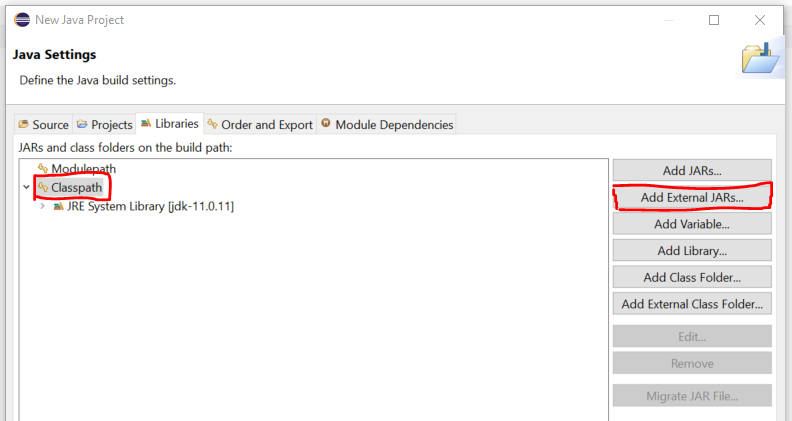
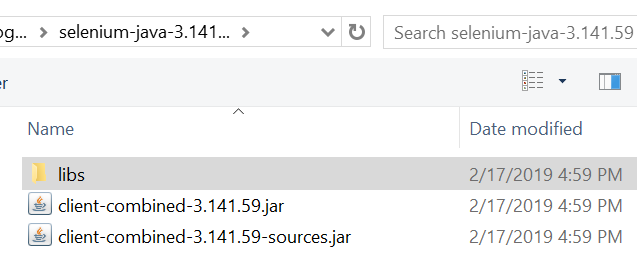
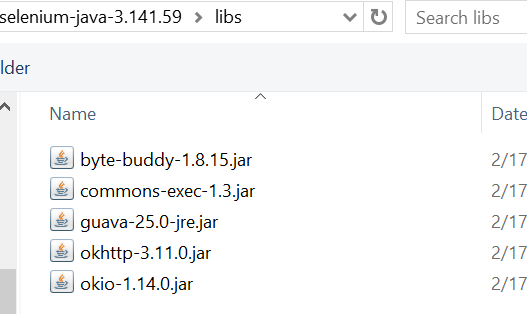
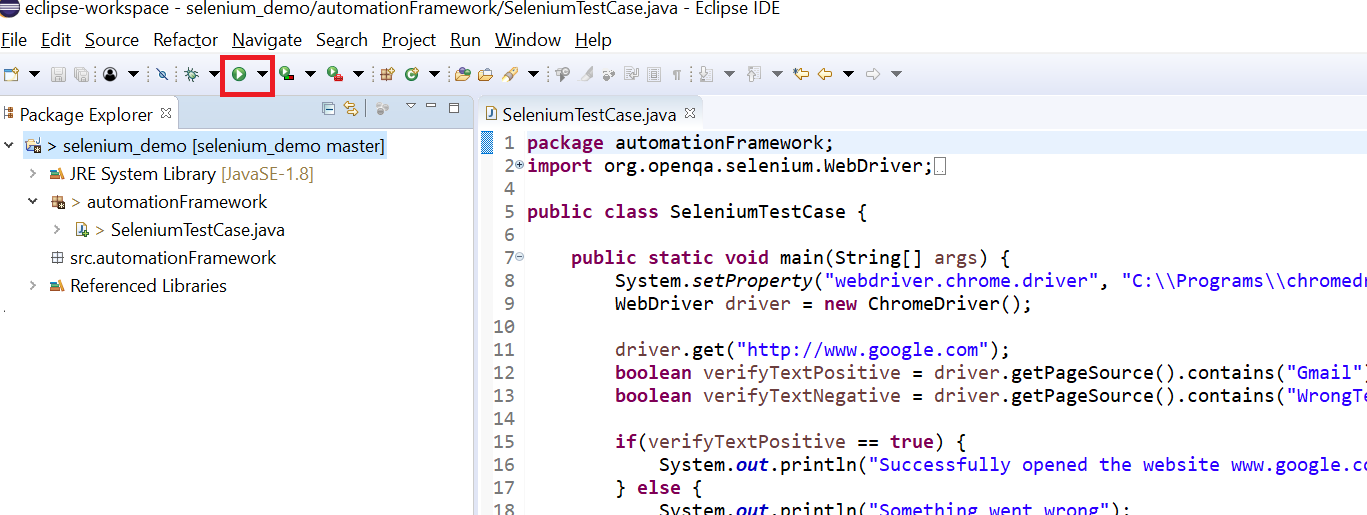
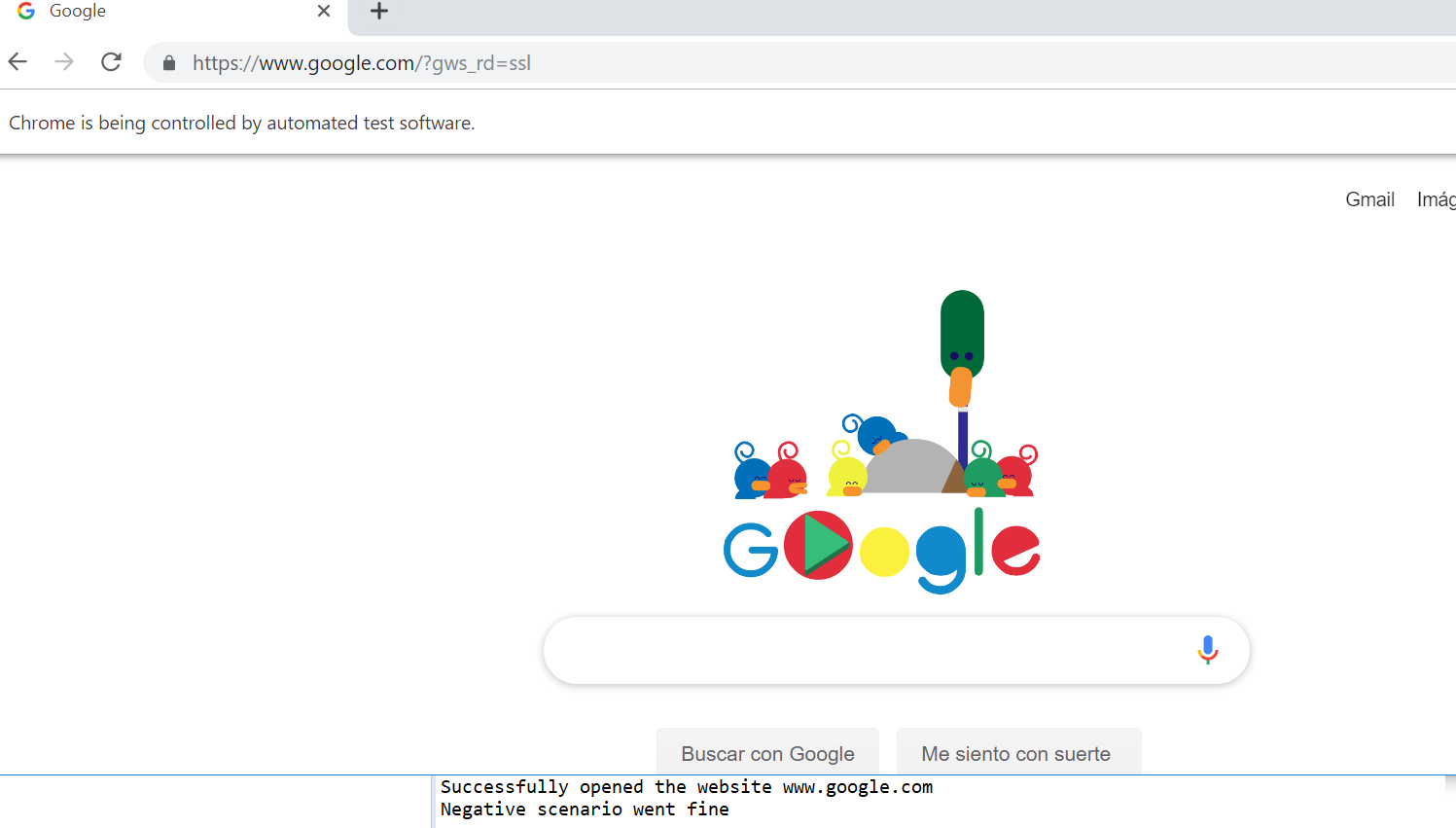
Selenium Demo

In this demo we will show how to create a Selenium script to test a webpage. We will use the Selenium Driver for Chrome, Eclipse as our IDE and will navigate through a webpage.

# Initial setup

1. Download Eclipse.
   1. <https://www.eclipse.org/downloads/>
2. Install the IDE for Java Developers
3. 
4. Download Chrome Selenium Driver.
   1. <http://chromedriver.chromium.org/downloads>
5. Extract the contents in the following folder.
   1. **Note: that the folder name is hardcoded in the code we will be using, so make sure to put it in this exact location.**
   2. C:\Programs\chromedriver\_win32
   3. 
6. Download Selenium Java libraries and likewise place them in a folder.
   1. <https://www.seleniumhq.org/download/>
   2. For example: C:\Programs\selenium-java-3.141.59
   3. 

# Running the Selenium Test Case

1. Download a Selenium Script Test case.
   1. For example you can use: <https://github.com/mperales90/CQAcademy/tree/master/selenium_demo>
2. Import the Java project in Eclipse.
   1. New 🡪 Java Project 🡪 Browse 🡪 Navigate to folder with Selenium Git repo 🡪 Click Select Folder 🡪 Click Next
   2. 
   3. 
3. Go to the Libraries Tab 🡪 Select the Classpath folder 🡪 Click on Add External Jars 🡪 Import the Selenium Java libraries 🡪 Add the previously downloaded selenium-java jars 🡪 Finish
4. Do this twice for the Selenium libraries inside the main folder and again for the ones inside the lib folder.
5. 
   1. 
   2. 
6. Open SeleniumTestCase.java and run it.
   1. 
7. You will be able to see Chrome opening and the validations on the Console Output.
   1. 
   2. Explain how the code is validating text in the webpage for a Positive and a Negative test case scenario.

JMeter Demo

In this demo we will explain how to run a performance test of a website using JMeter.

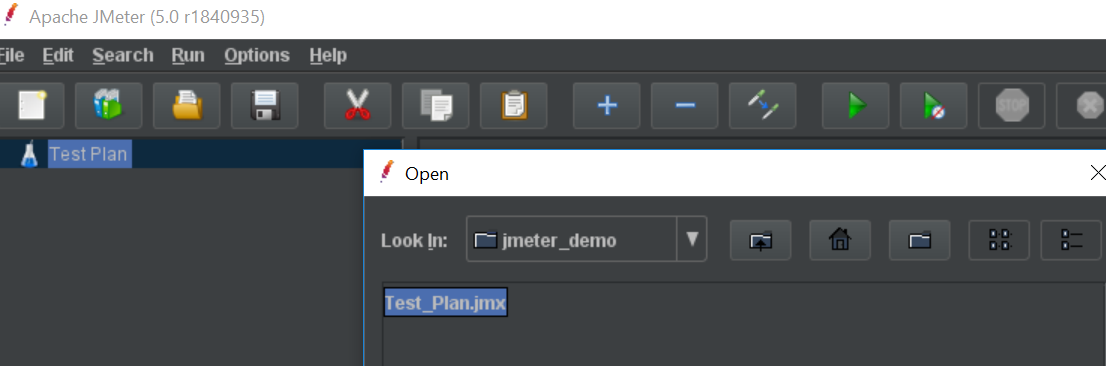
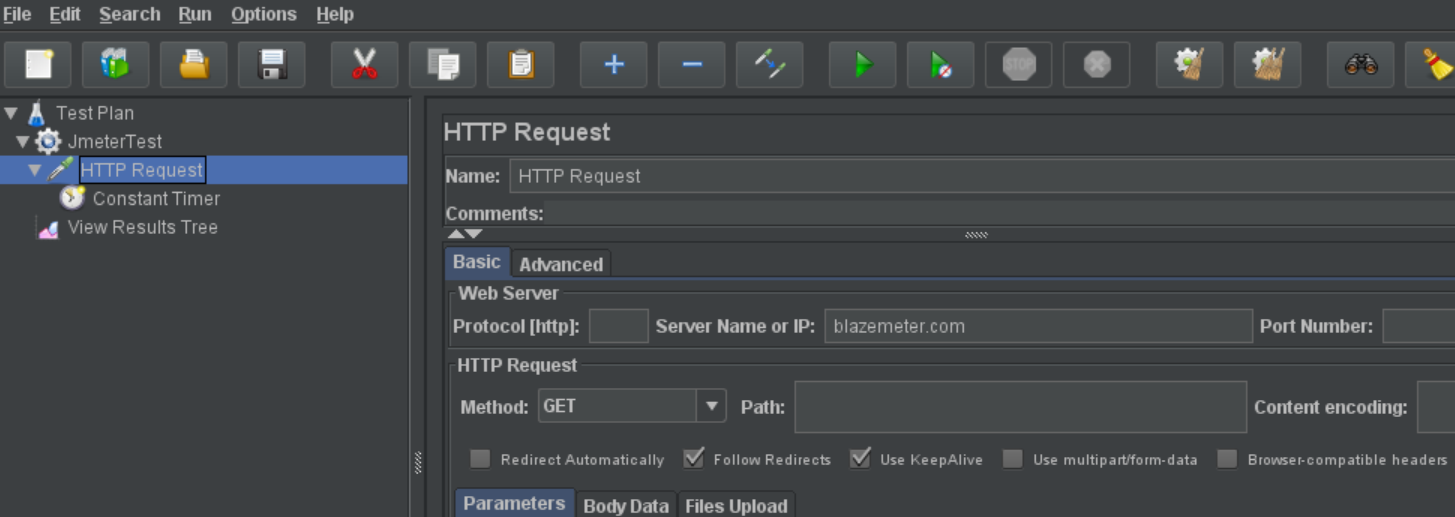
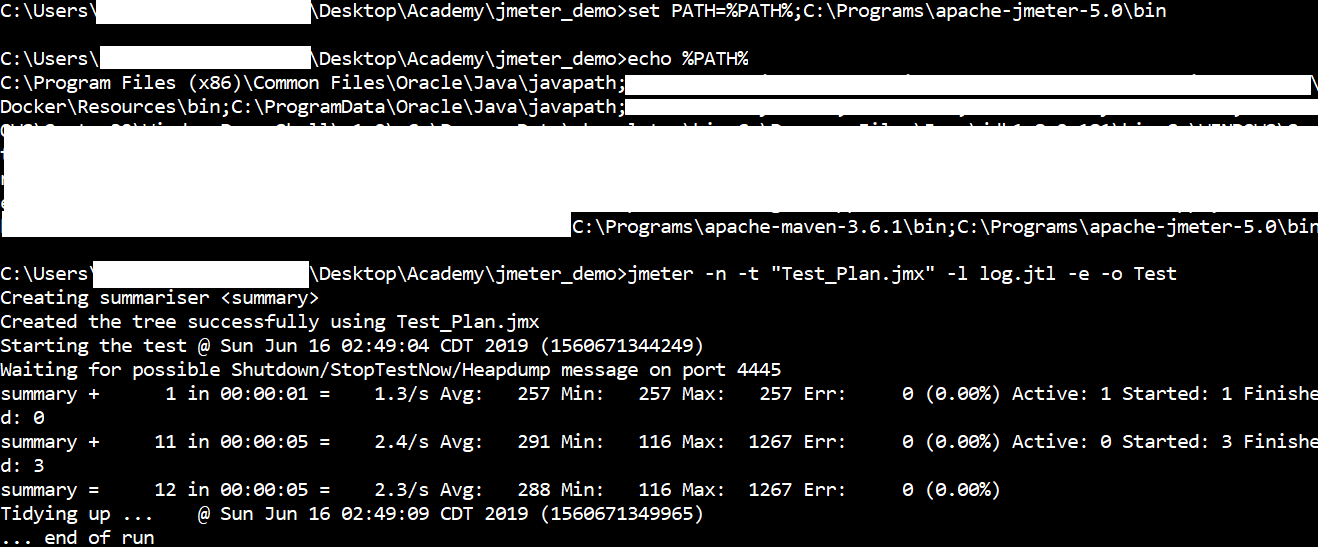
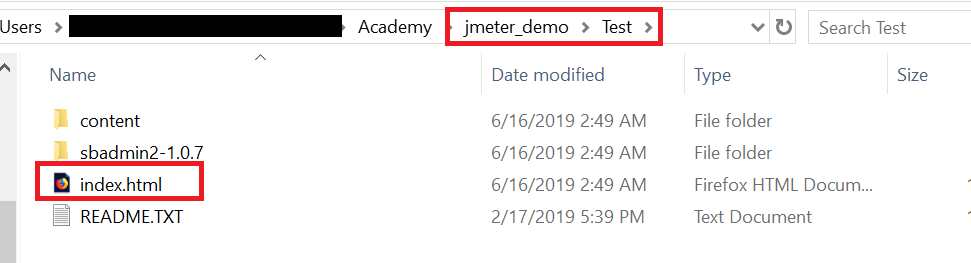
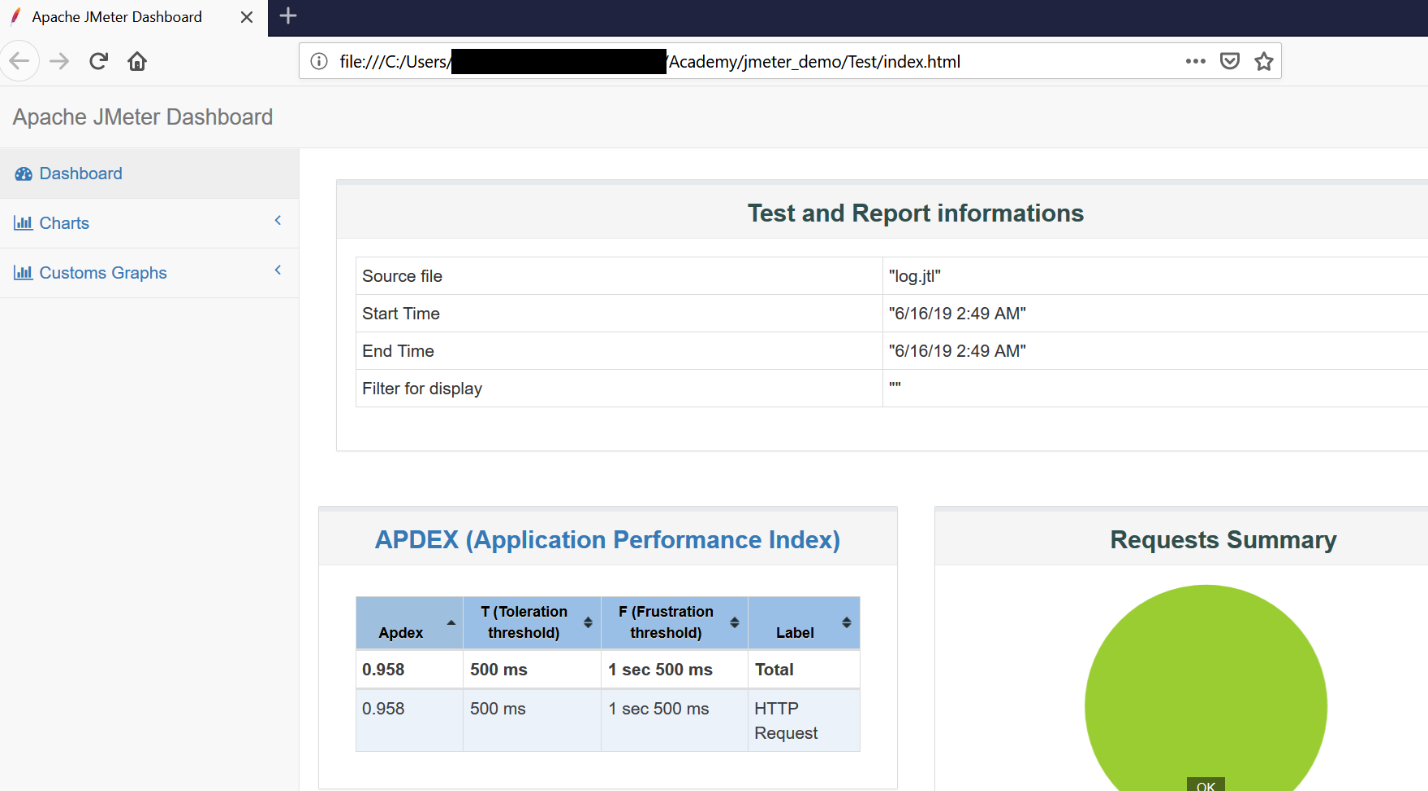
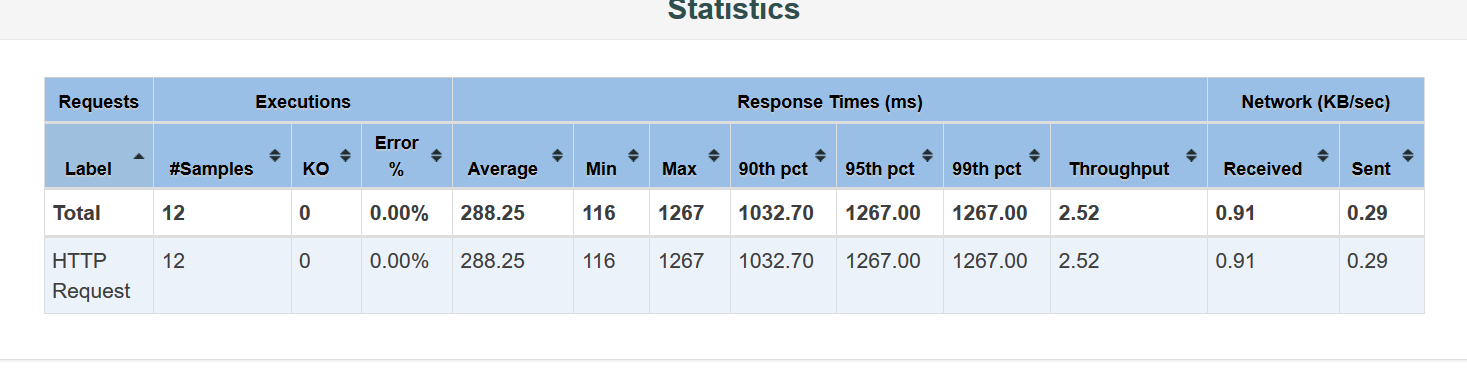
# Initial setup

1. Download JMeter.

* 1. <https://jmeter.apache.org/download_jmeter.cgi>
  2. Extract it in the location where you want to install it.
     1. For example: C:\Programs\apache-jmeter-5.4.3

1. Add JMeter to your environment variables.
   1. If you don’t have admin access, run the following command in the command prompt.
      1. set PATH=%PATH%;C:\Programs\apache-jmeter-5.4.3\bin
      2. Note that this command only works for the session of the command prompt in which you ran it. If you close it you will need to run it again.

# Running the JMeter Test

1. Open JMeter by going to C:\Programs\apache-jmeter-5.4.3\bin\jmeter.sh or jmeterw.cmd (C:\Programs\apache-jmeter-5.3\bin)
2. Download a sample JMeter script.
   1. For example you can use: <https://github.com/mperales90/CQAcademy/tree/master/jmeter_demo>
3. Open the Test\_Plan.jmx
   1. Show how the Test\_Plan.jmx is configured to run transactions against blazemeter.com site.
   2. 
   3. 
4. Open the command prompt and navigate to the location of your JMeter script.
   1. Note that we downloaded the JMeter version for Windows, so it will not run in bash.
5. Add JMeter to the env variables:
   1. set PATH=%PATH%;C:\Programs\apache-jmeter-5.4.3\bin
6. Run the JMeter script and generate reports.
   1. jmeter -n -t "Test\_Plan.jmx" -l log.jtl -e -o Test
   2. 
7. Navigate to the folder where your JMeter script is located with the Windows explorer.
   1. Notice how reports were generated.
   2. Open the index.html file to see the report summary.
   3. 
   4. 
   5. 
   6. Explain the different graphics.