Download Prioritization Tool from the GitHub repository,
https://github.com/dongchengli/Team-8-Prioritization-Tool

Put Downloaded Team-8-Prioritization-Tool-master.zip file inside a new directory or any directory (Directory named simple and with no space between words is recommended, such as Tool), and extract the zip file.
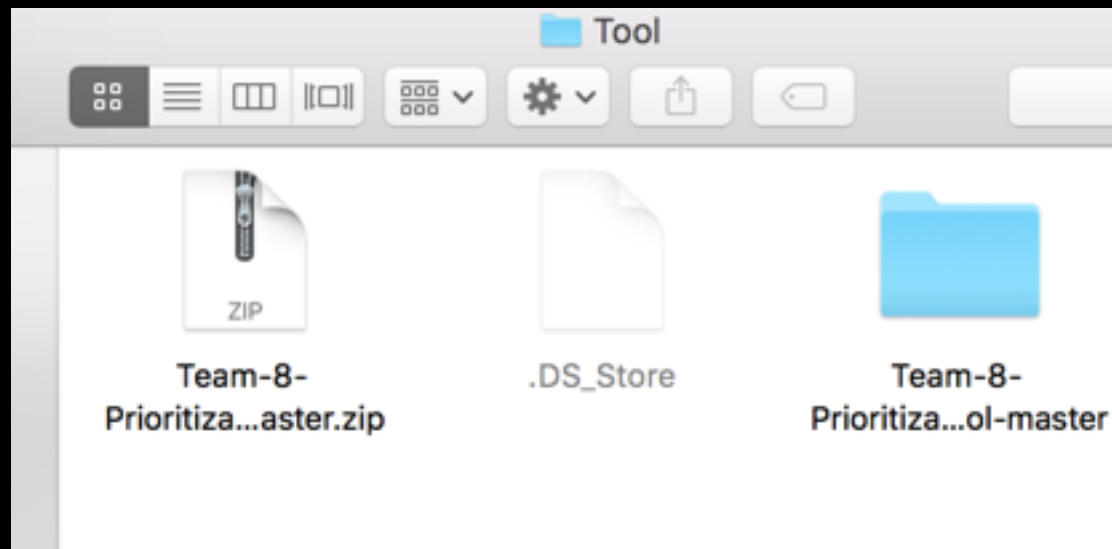
From the commend line, first check the maven, if no maven installed, it need to be installed

```
[lidongchengdeMacBook-Pro:~ Dongcheng$ mvn -v
Apache Maven 3.3.9 (bb52d8502b132ec0a5a3f4c09453c07478323dc5; 2015-11-10T10:41:47-06:00)
Maven home: /Users/Dongcheng/apache-maven-3.3.9
Java version: 1.8.0_73, vendor: Oracle Corporation
Java home: /Library/Java/JavaVirtualMachines/jdk1.8.0_73.jdk/Contents/Home/jre
Default locale: zh_CN, platform encoding: UTF-8
OS name: "mac os x", version: "10.11.5", arch: "x86_64", family: "mac"
lidongchengdeMacBook-Pro:~ Dongcheng$
```

Dongcheng — -bash — 85×27

In the Commend Line Direct to the directory of the CodeCoverage folder inside of the Team-8-Prioritization-Tool-master folder

Then press mvn install, and it will show build successfully

Go to https://github.com/julman99/gson-fire, then download this real world
Github project for testing purpose.

Put Downloaded gson-fire-master.zip file inside a new directory or any directory (Directory named simple and with no space between words is recommended, such as RealWorldProject), and extract the zip file.
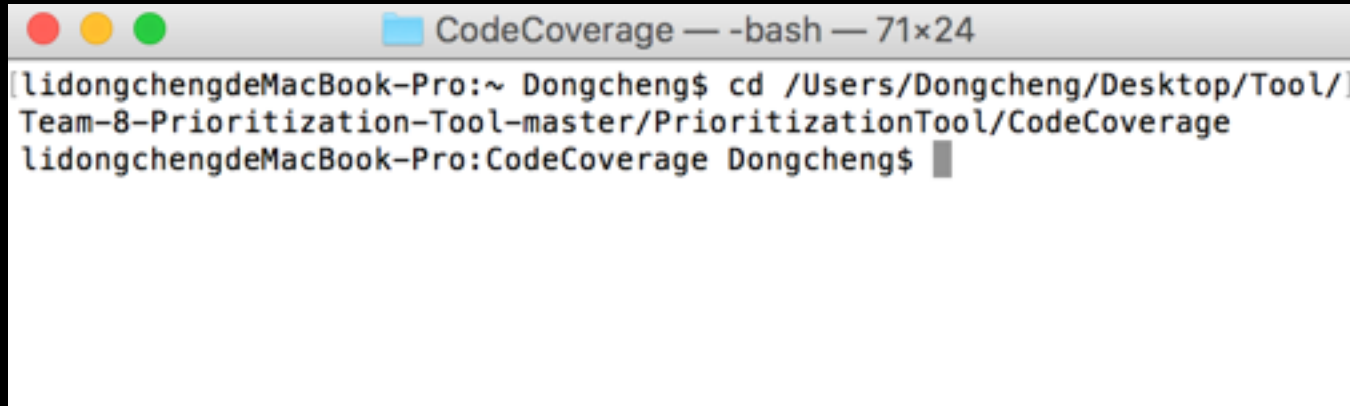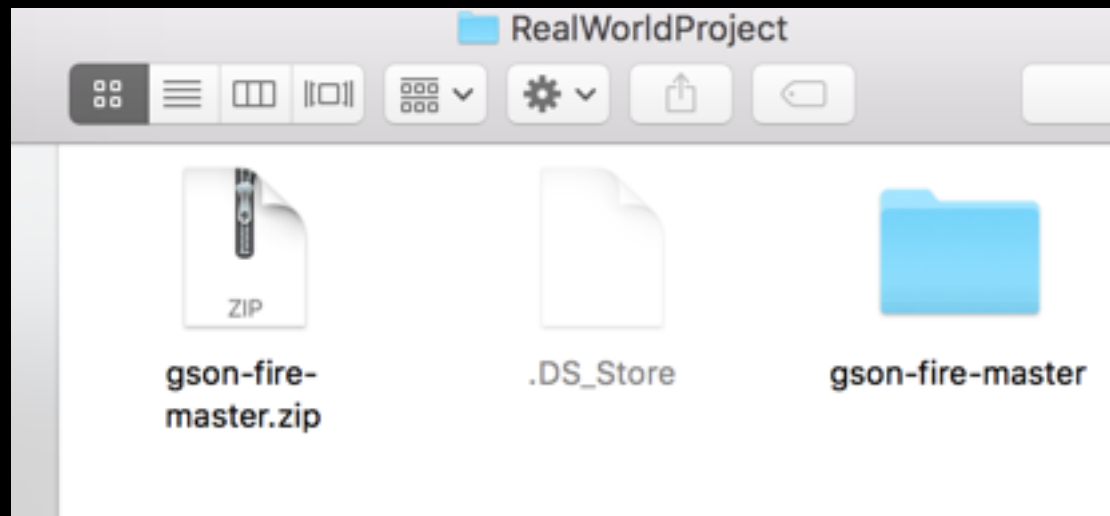
Open the pom.xml file in the root directory of the project
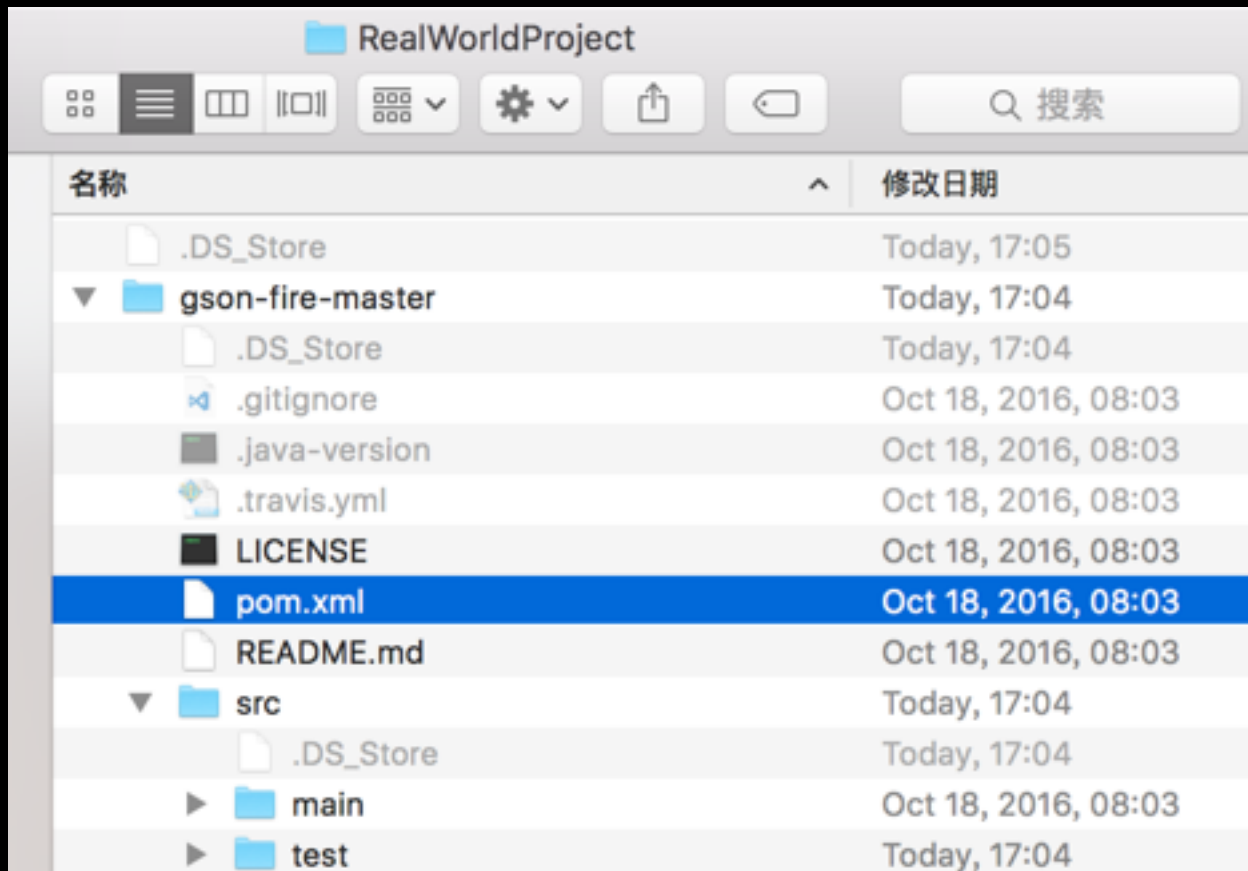
First you need to add dependencies

Such as in the example

```xml
<dependency>
        <groupId>edu.utdallas</groupId>
        <artifactId>code-coverage</artifactId>
        <version>1.0-SNAPSHOT</version>
        <scope>test</scope>
  </dependency>
  <dependency>
        <groupId>org.ow2.asm</groupId>
        <artifactId>asm-all</artifactId>
        <version>5.1</version>
        <scope>test</scope>
  </dependency>
```

```xml
<dependencies>
    <dependency>
        <groupId>edu.utdallas</groupId>
        <artifactId>code-coverage</artifactId>
        <version>1.0-SNAPSHOT</version>
        <scope>test</scope>
    </dependency>
    <dependency>
        <groupId>org.ow2.asm</groupId>
        <artifactId>asm-all</artifactId>
        <version>5.1</version>
        <scope>test</scope>
    </dependency>
    <dependency>
        <groupId>com.google.code.gson</groupId>
        <artifactId>gson</artifactId>
        <version>2.7</version>
    </dependency>
    <dependency>
        <groupId>junit</groupId>
        <artifactId>junit</artifactId>
        <version>4.10</version>
        <scope>test</scope>
    </dependency>
</dependencies>
```

Next you need to add the plugins

Such as in the example

```
<build>
  <plugins>
    <plugin>
      <groupId>org.apache.maven.plugins</groupId>
      <artifactId>maven-dependency-plugin</artifactId>
      <version>2.3</version>
      <executions>
        <execution>
          <goals>
            <goal>properties</goal>
          </goals>
        </execution>
      </executions>
            </plugin>

    <plugin>
      <groupId>org.apache.maven.plugins</groupId>
      <artifactId>maven-surefire-plugin</artifactId>
      <version>2.19.1</version>
      <configuration>
      <argLine>-javaagent:/YourPath/code-coverage-1.0-SNAPSHOT.jar=$
{project.groupId}</argLine>
        <properties>
          <property>
            <name>listener</name>
            <value>edu.utdallas.JUnitExecutionListener</value>
          </property>
        </properties>
        </configuration>
      </plugin>
    </plugins>
</build>
```

```
<build>
    <plugins>
        <plugin>
            <groupId>org.apache.maven.plugins</groupId>
            <artifactId>maven-dependency-plugin</artifactId>
            <version>2.3</version>
            <executions>
                <execution>
                    <goals>
                        <goal>properties</goal>
                    </goals>
                </execution>
            </executions>
        </plugin>
        <plugin>
            <groupId>org.apache.maven.plugins</groupId>
            <artifactId>maven-surefire-plugin</artifactId>
            <version>2.19.1</version>
            <configuration>
                <argLine>-javaagent:/Users/Dongcheng/Desktop/Tool/Team-8-
                Prioritization-Tool-master/PrioritizationTool/CodeCoverage/
                target/code-coverage-1.0-SNAPSHOT.jar=${project.groupId}</
                argLine>
                <properties>
                    <property>
                        <name>listener</name>
                        <value>edu.utdallas.JUnitExecutionListener</value>
                    </property>
                </properties>
            </configuration>
        </plugin>
        <plugin>
            <groupId>org.apache.maven.plugins</groupId>
            <artifactId>maven-compiler-plugin</artifactId>
            <version>3.1</version>
            <configuration>
                <source>1.6</source>
                <target>1.6</target>
            </configuration>
```

```
    </plugin>
    <plugin>
        <groupId>org.apache.maven.plugins</groupId>
        <artifactId>maven-surefire-plugin</artifactId>
        <version>2.19.1</version>
        <configuration>
            <argLine>-javaagent:/YourPath/code-coverage-1.0-SNAPSHOT.jar=$
                {project.groupId}</argLine>
            <properties>
                <property>
                    <name>listener</name>
                    <value>edu.utdallas.JUnitExecutionListener</value>
                </property>
            </properties>
        </configuration>
    </plugin>
```

The YourPath here, is the path of the code-coverage-1.0-SNAPSHOT.jar file, which we generated by mvn install the PrioritizationTool, this jar file is created under the target directory of the CodeCoverage
For example, in the example realworld project the YourPath is "/Users/Dongcheng/Desktop/Tool/Team-8-Prioritization-Tool-master/PrioritizationTool/CodeCoverage/target/".

▼ 📁 Team-8-Prioritization-Tool-master
    📄 .DS_Store
    ▼ 📁 PrioritizationTool
        📄 .DS_Store
        ▼ 📁 CodeCoverage
            ▪️ .classpath
            📄 .DS_Store
            ⋈ .gitignore
            ▪️ .project
            ▶ 📁 .settings
            📄 pom.xml
            ▶ 📁 src
            ▼ 📁 target
                ▶ 📁 classes
                📄 code-coverage-1.0-SNAPSHOT.jar
                ▶ 📁 generated-sources
                ▶ 📁 generated-test-sources
                ▶ 📁 maven-archiver
                ▶ 📁 maven-status
                ▶ 📁 test-classes
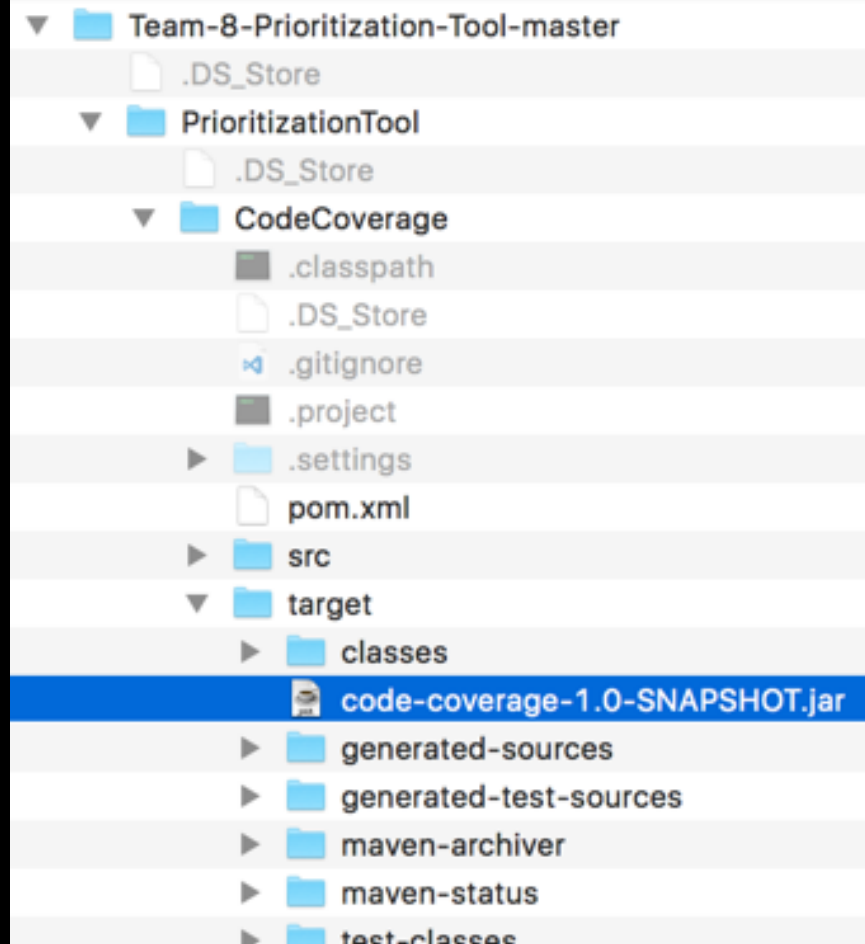
First Run

In the Commend Line, Direct to the directory of the pom.xml file inside of the test program then mvn clean, after that, mvn test

```
lidongchengdeMacBook-Pro:CodeCoverage Dongcheng$ cd /Users/Dongcheng/]
Desktop/RealWorldProject/gson-fire-master
lidongchengdeMacBook-Pro:gson-fire-master Dongcheng$ █
```

```
lidongchengdeMacBook-Pro:gson-fire-master Dongcheng$ mvn clean█
```

```
lidongchengdeMacBook-Pro:gson-fire-master Dongcheng$ mvn test█
```

In the console it will output the default class execution order and calculated total and additional prioritizations for the each test method

Then it will show build success

Second Run

First make two copies of the pom.xml file, one for Total Strategy and one for Additional strategy, rename them according to the Prioritization strategy they're using
For example, in the we name pom_test_AS for test program using Additional strategy, and pom_test_TS for test program using Total strategy.
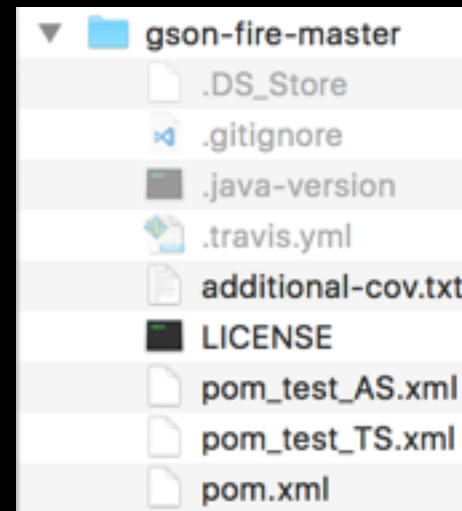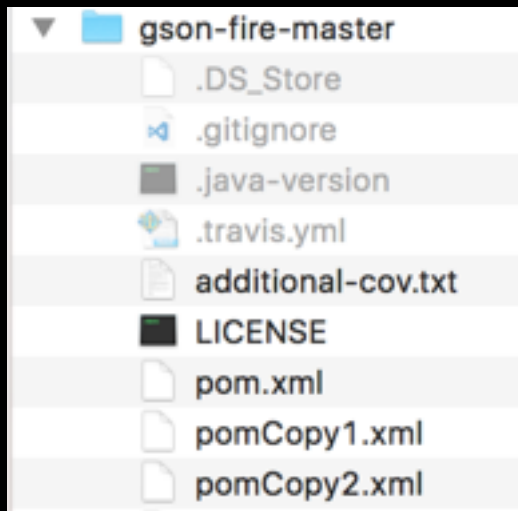
Then we add "Includes" after the beginning of the <configuration> tag or right before the <argLine> tag, inside of the "maven-surefire-plugin", add to the pom file according to the strategy its using

If using total strategy:

<includes><include>**/FeatureTestSuite.java</include></includes>

If using additional strategy:

<includes><include>**/FeatureTestSuiteAdditional.java</include></includes>

Such as in the test program, we add
<includes><include>**/FeatureTestSuite.java</include></includes>
In the pom_test_TS.xml, and we add

```
<plugin>
    <groupId>org.apache.maven.plugins</groupId>
    <artifactId>maven-surefire-plugin</artifactId>
    <version>2.19.1</version>
    <configuration>
        <includes><include>**/FeatureTestSuiteAdditional.java</include></includes>
        <argLine>-javaagent:/Users/Dongcheng/Desktop/Tool/Team-8-Prioritization-
            Tool-master/PrioritizationTool/CodeCoverage/target/code-coverage-1.0-
            SNAPSHOT.jar=${project.groupId}</argLine>
        <properties>
            <property>
                <name>listener</name>
                <value>edu.utdallas.JUnitExecutionListener</value>
            </property>
        </properties>
    </configuration>
</plugin>
```

Finally, if we want to test program using Total strategy, under the directory of the test program, we just type "mvn test -f pom_test_TS.xml" or if we want to test program using Additional strategy we just type "mvn test -f pom_test_AS.xml"

```
lidongchengdeMacBook-Pro:gson-fire-master Dongcheng$ mvn test -f pom_test_TS.xml
```

```
lidongchengdeMacBook-Pro:gson-fire-master Dongcheng$ mvn test -f pom_test_AS.xml
```

In the console it will output the manipulated class execution order using the choice of user