How to Add Selenium Testing to maven heart (pm.xml file)

UNIT TESTING is a type of software testing where individual units or components of a software are tested. The purpose is to validate that each unit of the software code performs as expected. Unit Testing is done during the development (coding phase) of an application by the developers. Unit Tests isolate a section of code and verify its correctness. A unit may be an individual function, method, procedure, module, or object.

Why Unit Testing?

Unit Testing is important because software developers sometimes try saving time doing minimal unit testing and this is myth because inappropriate unit testing leads to high-cost Defect fixing during System Testing, Integration Testing and even Beta Testing after application is built. If proper unit testing is done in early development, then it saves time and money in the end.

What is Integration Testing?

INTEGRATION TESTING is defined as a type of testing where software modules are integrated logically and tested as a group. A typical software project consists of multiple software modules, coded by different programmers. The purpose of this level of testing is to expose defects in the interaction between these software modules when they are integrated. Integration Testing focuses on checking data communication amongst these modules. Hence it is also termed as 'I & T' (Integration and Testing), 'String Testing' and sometimes 'Thread Testing

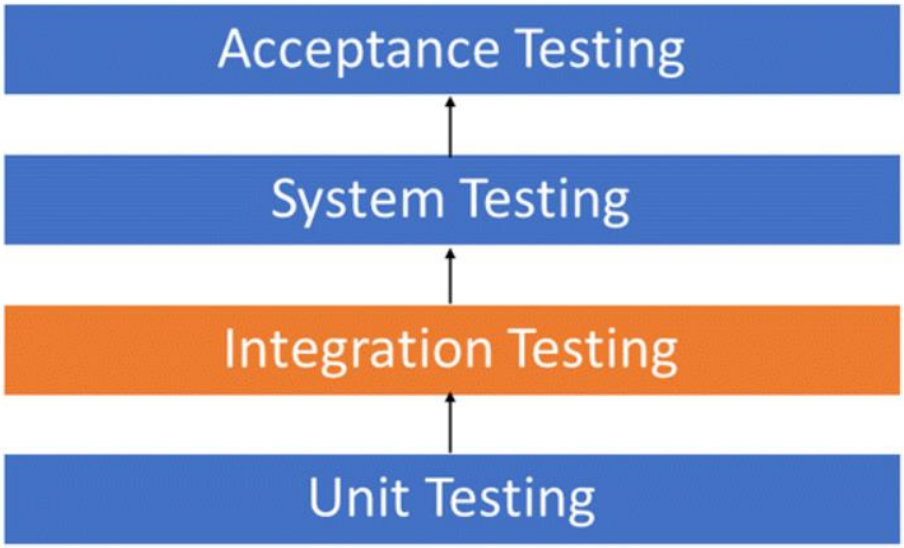
System testing

System testing, also referred to as system-level tests or system-integration testing, is the **process in which a quality assurance (QA) team evaluates how the various components of an application interact together in** the full, integrated system or application.

Acceptance testing

In DeVops engineering and its various subdisciplines, acceptance testing is a test conducted to determine if the requirements of a specification of the software are met.

Why do Integration Testing?



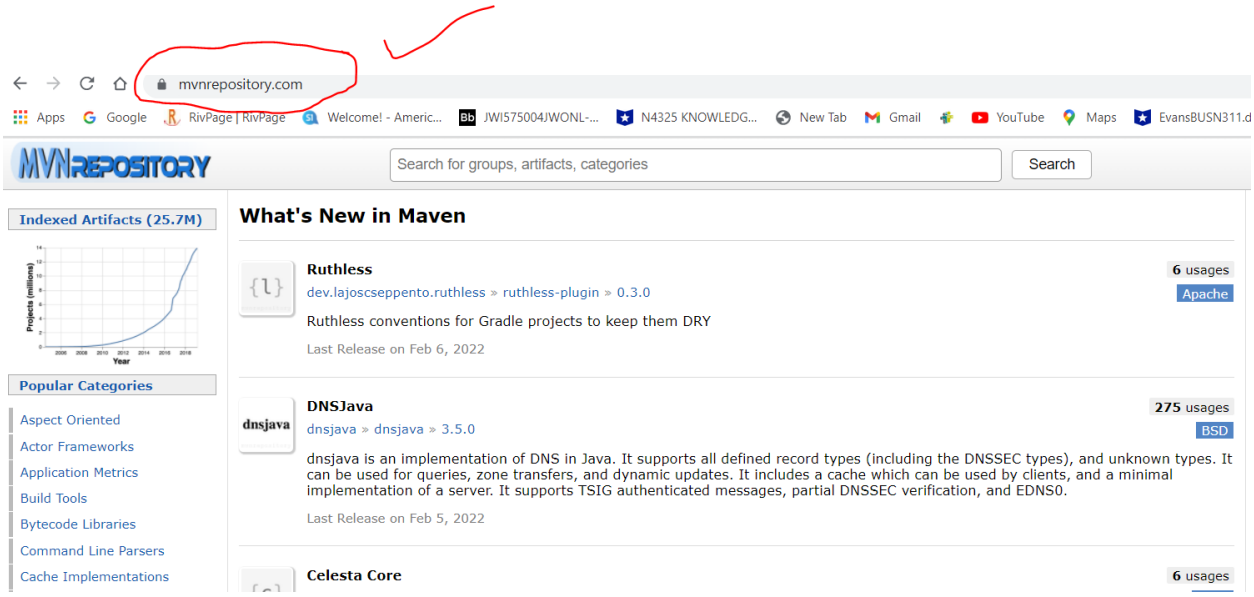
Usually, the Unit and Integration tests are done manually. However, selenium has ability to do the unit and integration test which saves the company a lot of time and money.

Step 1

We need to get the selenium dependencies so that we can add them to the pom.xml file

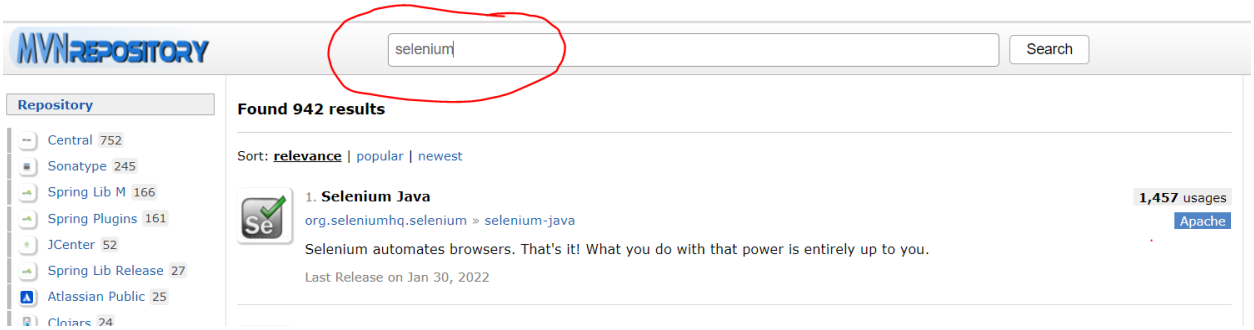
Go to

Link: https://mvnrepository.com/



Step 2

In the mvnrepository, type **selenium** in the search box

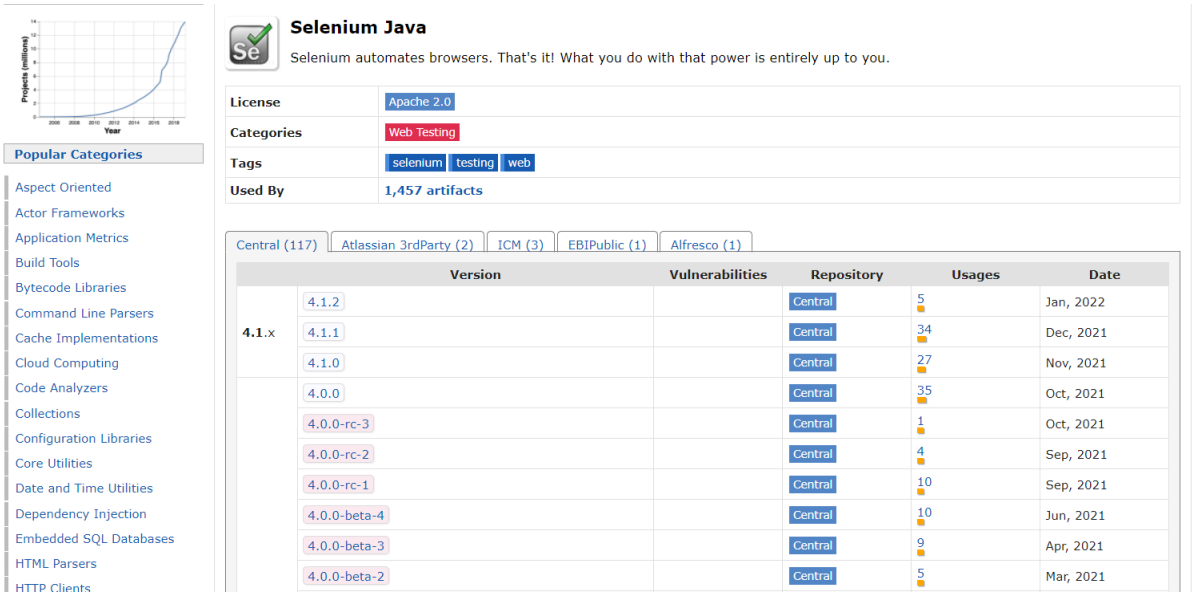


Step 3

Select the selenium java and click on the link

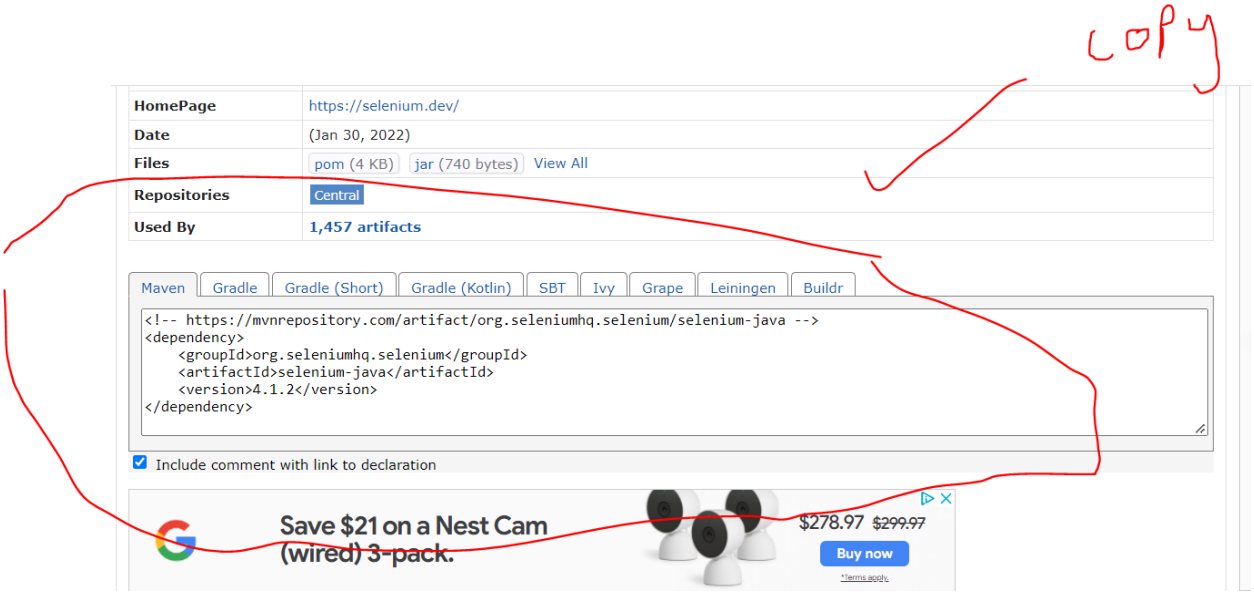
Step 4

Select the latest version of 4.1.2



Step 5

Copy the html links



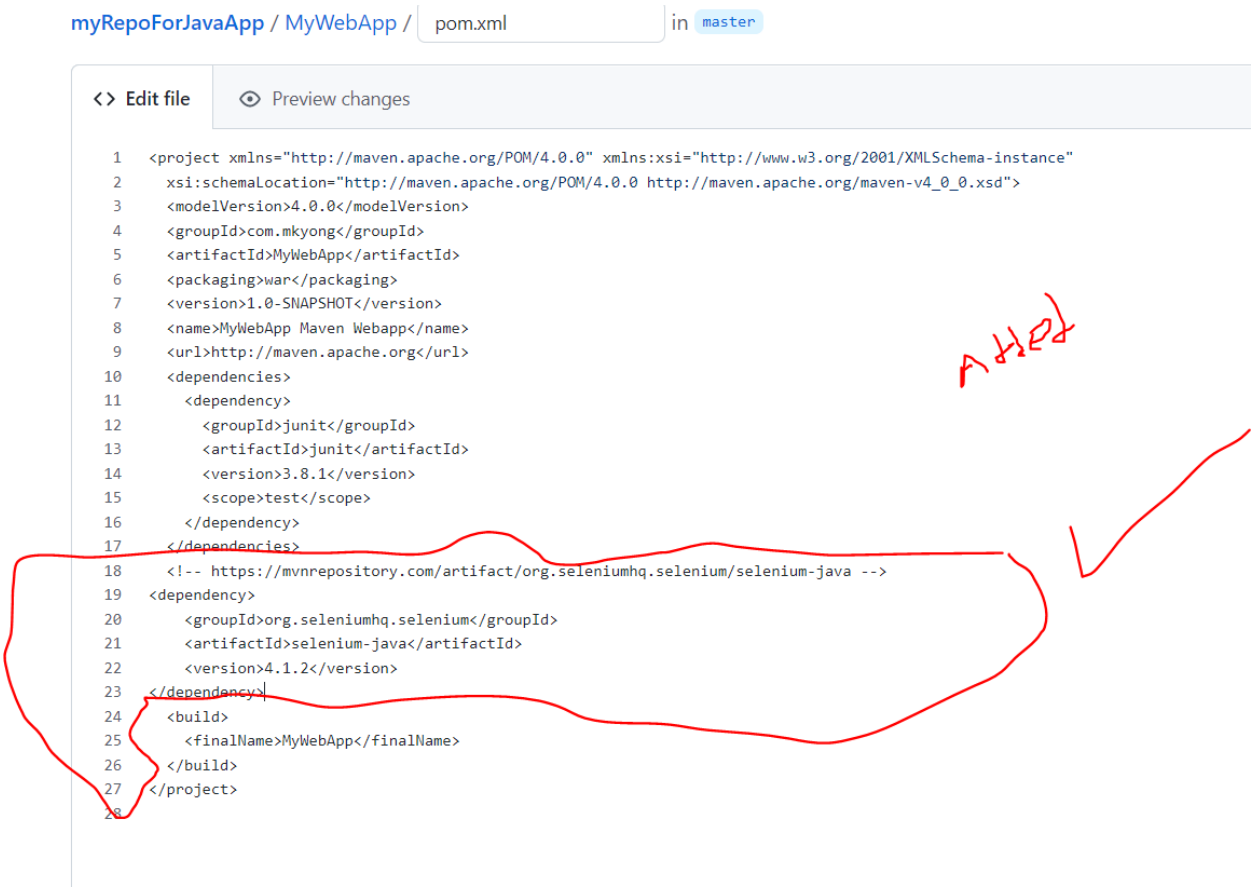
Step 6

Go to

Link: https://github.com/joshking1/myRepoForJavaApp.git

Step 7

Locate the pom.xml file in this repo and add the html code you copied from the maven repository




```
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/maven-v4_0_0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>com.mkyong</groupId>
  <artifactId>MyWebApp</artifactId>
  <packaging>war</packaging>
  <version>1.0-SNAPSHOT</version>
  <name>MyWebApp Maven Webapp</name>
  <url>http://maven.apache.org</url>
  <dependencies>
    <dependency>
      <groupId>junit</groupId>
      <artifactId>junit</artifactId>
      <version>3.8.1</version>
      <scope>test</scope>
    </dependency>
  </dependencies>
  <!-- https://mvnrepository.com/artifact/org.seleniumhq.selenium/selenium-java -->
<dependency>
    <groupId>org.seleniumhq.selenium</groupId>
    <artifactId>selenium-java</artifactId>
    <version>4.1.2</version>
</dependency>
  <build>
    <finalName>MyWebApp</finalName>
  </build>
</project>
```