SPRING BOOT SETUP Using
SPRING INITIALIZR and ECLIPSE
- by Raghu

➢ Open web browser and type "https://start.spring.io/", press enter.

➢ It will be navigated to SPRING INITIALIZR web site.

Artifact coordinates

Group

com.example

Artifact

demo

Add Spring Boot Starters and dependencies to your applicatio

Search for dependencies

Web, Security, JPA, Actuator, Devtools...

Selected Dependencies

click here

Generate Project alt + ⏎

Don't know what to look for? Want more options? Switch to the full version.

➢ Then click on "Switch to Full Version" . Observe the above screen.

➢ Fill required details and then press "Generate Project".

➢ Look at below screen for complete details...

SPRING INITIALIZR bootstrap your application now

https://start.spring.io

Generate a [Maven Project ▼] with [Java ▼] and Spring Boot [1.5.9 ▼] (1)

Project Metadata (2)

Artifact coordinates

Group

com.example

Artifact

demo

Name

demo

Description

Demo project for Spring Boot

Package Name (3)

com.app

Packaging

Jar

Java Version (4)

7

Too many options? Switch back to the simple version.

(6)

Generate Project alt + ↵

Dependencies

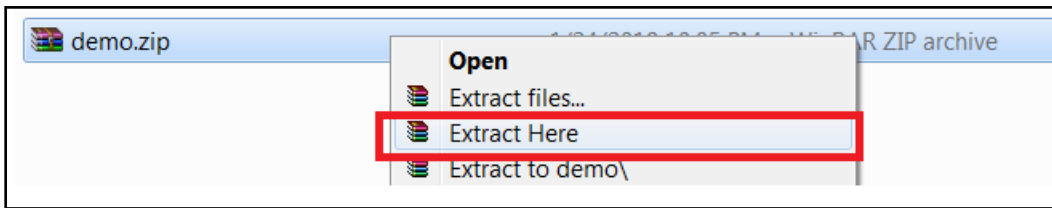Add Spring Boot Starters and dependencies to your application

Search for dependencies

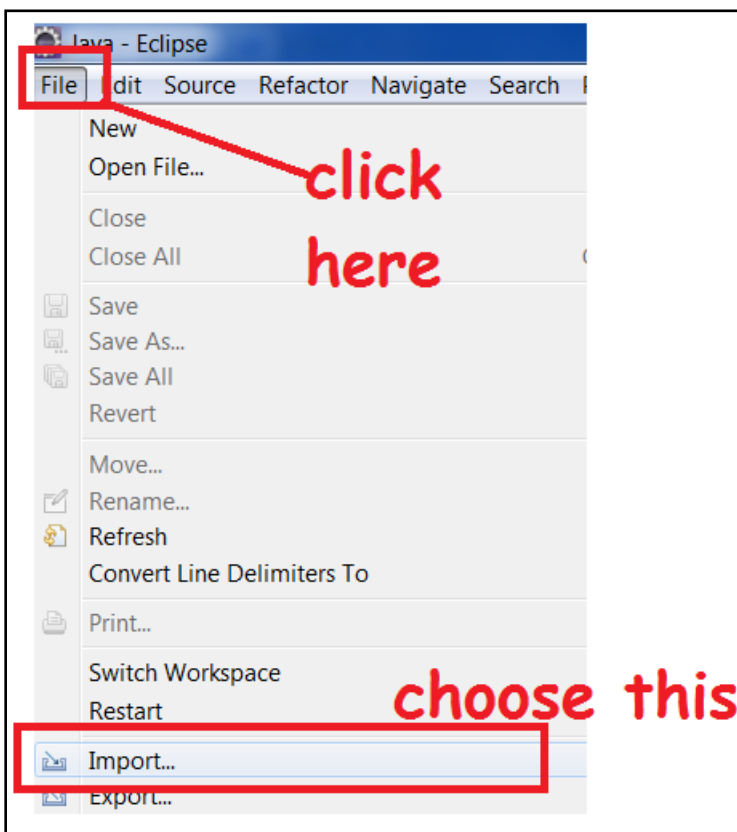Web, Security, JPA, Actuator, Devtools...

Selected Dependencies (5)

➤ It will download one ZIP File ex:"demo.zip"

Downloads   🔍 Search downloads

Today

demo.zip
https://start.spring.io/starter.zip?type=maven-project&language=java&bootVersion...
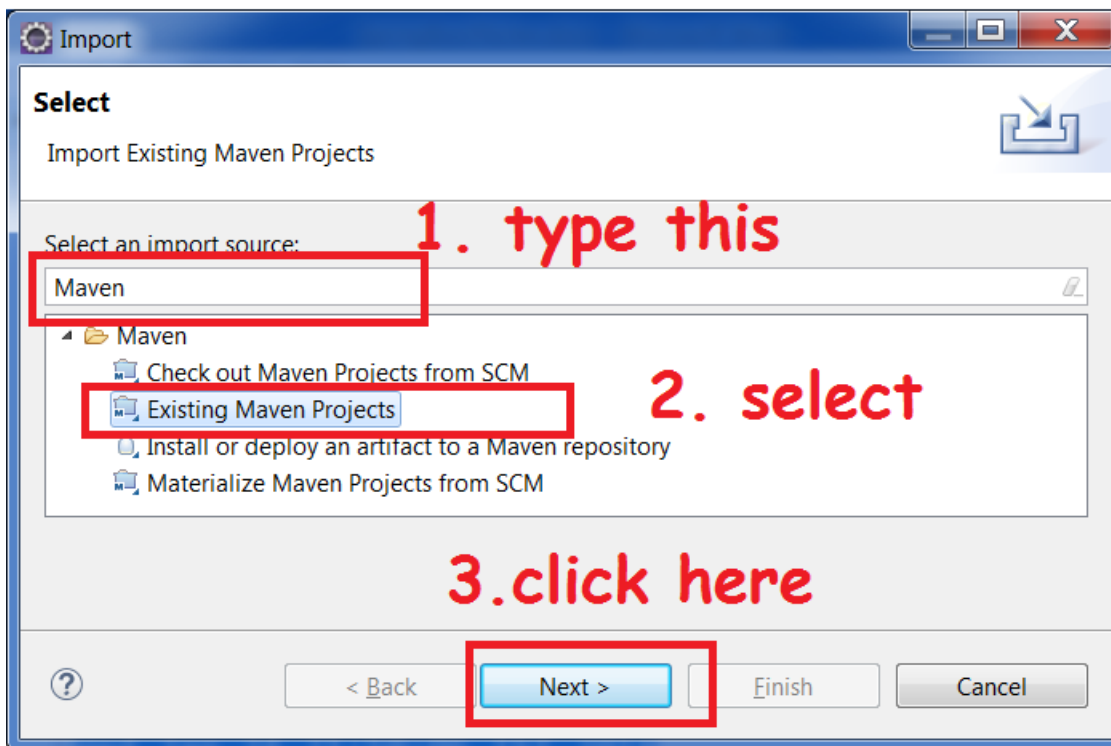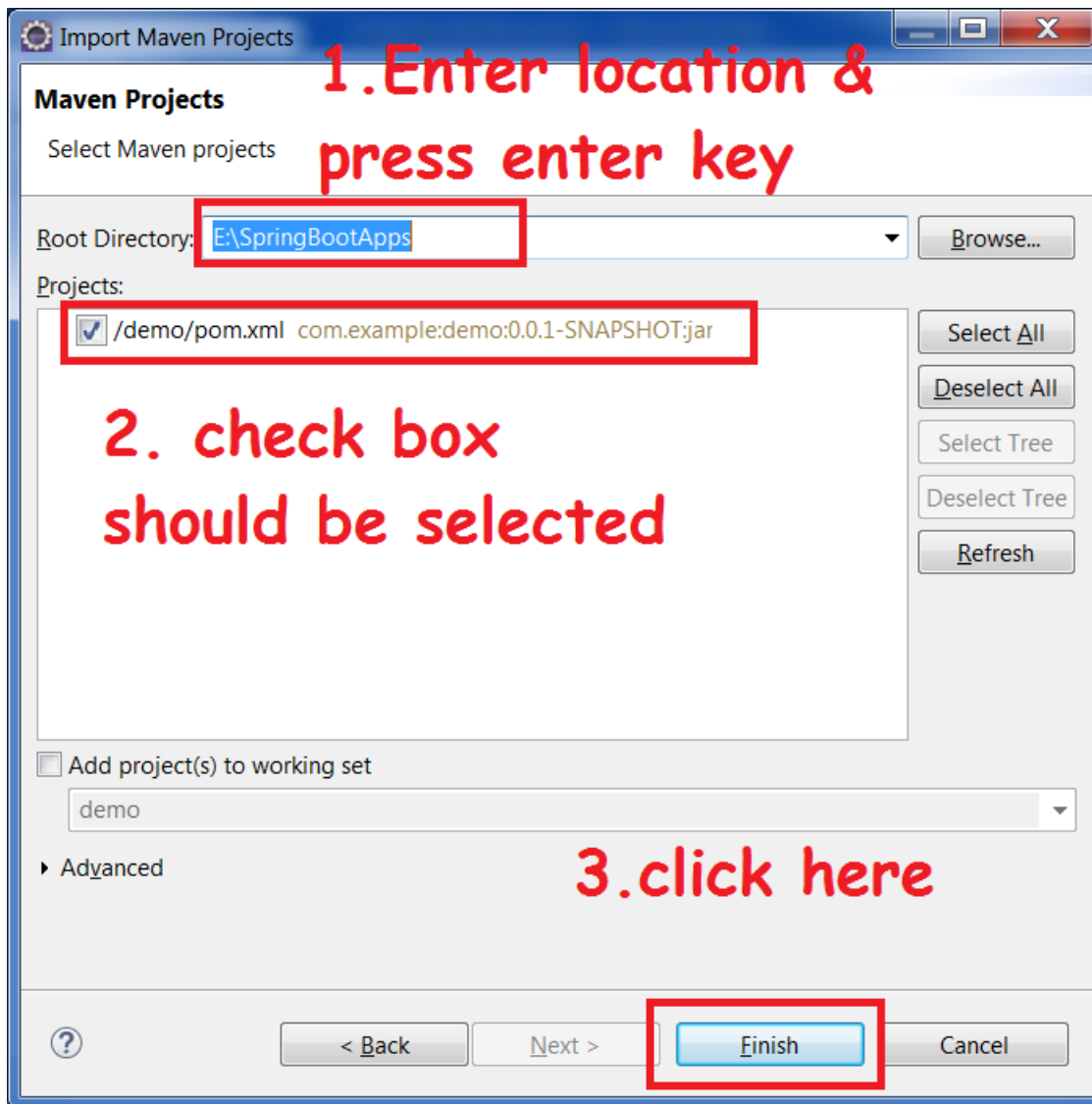
Show in folder
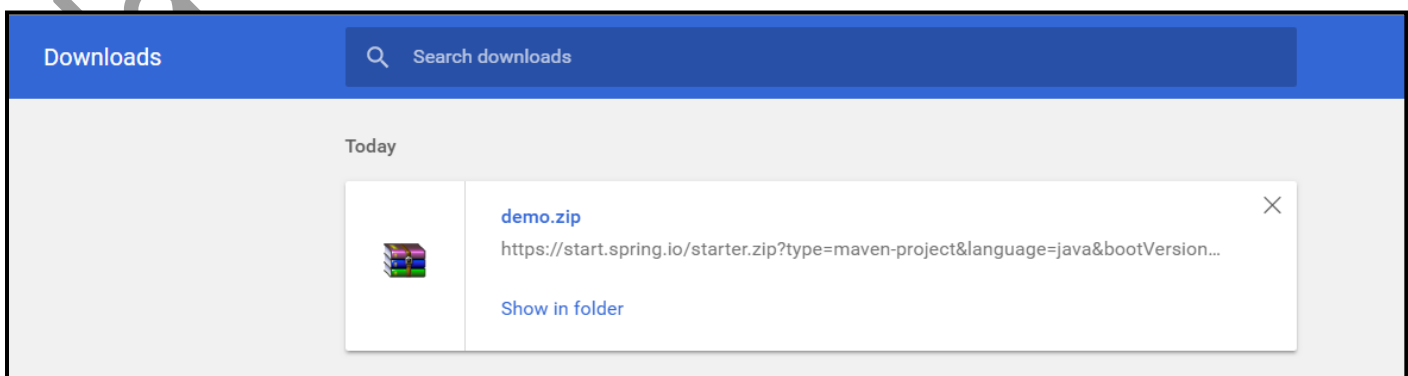
➢ Extract this ZIP into one folder

➢ Open Eclipse Now
➢ Go to File menu , here choose "Import" option.

➢ One new window will appear, there enter "Maven" in search screen and select "Existing Maven Project"

➢ Click on Next button, in next stage Enter Root Directory details , i.e. where your project is downloaded (folder location of your project)
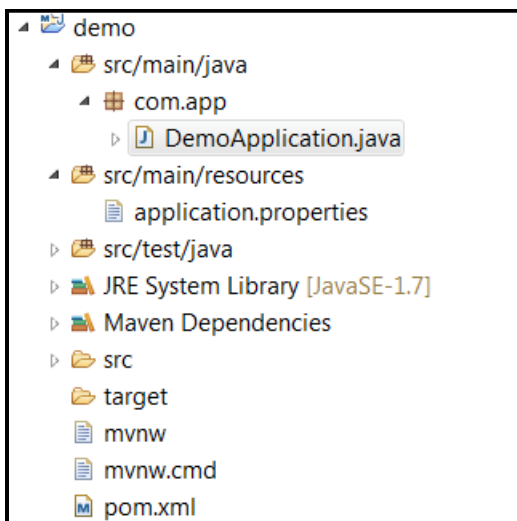
> Now It takes some time to download Jars and to do other setup wait for few minutes.. (internet connection is required for all this setup for first time).

Note: For application.properties details link is:

https://docs.spring.io/spring-boot/docs/current/reference/html/common-application-properties.html

Folder Structure:

```
▲ 👜 demo
   ▲ 🗁 src/main/java
      ▲ ⊞ com.app
         ▷ 🗋 DemoApplication.java
   ▲ 🗁 src/main/resources
         📄 application.properties
   ▷ 🗁 src/test/java
   ▷ 🛋 JRE System Library [JavaSE-1.7]
   ▷ 🛋 Maven Dependencies
   ▷ 🗁 src
      🗁 target
      📄 mvnw
      📄 mvnw.cmd
      Ⓜ pom.xml
```

1.pom.xml

```xml
<?xml version="1.0" encoding="UTF-8"?>
<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/xsd/maven-4.0.0.xsd">
     <modelVersion>4.0.0</modelVersion>

     <groupId>com.example</groupId>
     <artifactId>demo</artifactId>
     <version>0.0.1-SNAPSHOT</version>
     <packaging>jar</packaging>

     <name>demo</name>
     <description>Demo project for Spring Boot</description>

     <parent>
          <groupId>org.springframework.boot</groupId>
          <artifactId>spring-boot-starter-parent</artifactId>
          <version>1.5.9.RELEASE</version>
          <relativePath/> <!-- lookup parent from repository -->
     </parent>

     <properties>
       <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
       <project.reporting.outputEncoding>UTF-8</project.reporting.outputEncoding>
```

```xml
          <java.version>1.7</java.version>
     </properties>


     <dependencies>
          <dependency>
               <groupId>org.springframework.boot</groupId>
               <artifactId>spring-boot-starter</artifactId>
          </dependency>


          <dependency>
               <groupId>org.springframework.boot</groupId>
               <artifactId>spring-boot-starter-test</artifactId>
               <scope>test</scope>
          </dependency>
     </dependencies>


     <build>
          <plugins>
               <plugin>
                    <groupId>org.springframework.boot</groupId>
                    <artifactId>spring-boot-maven-plugin</artifactId>
               </plugin>
          </plugins>
     </build>


</project>
```

2.application.properties

```
logging.level.com.app=OFF
logging.level.org.springframework=OFF
```

3.Spring Boot Starter Class

```java
package com.app;

import org.springframework.boot.Banner;
import org.springframework.boot.CommandLineRunner;
import org.springframework.boot.SpringApplication;
```

```java
import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication
public class DemoApplication implements CommandLineRunner{

    public static void main(String[] args) {
        //SpringApplication.run(DemoApplication.class, args);
        SpringApplication app=new SpringApplication(DemoApplication.class);
        app.setBannerMode(Banner.Mode.OFF);
        app.run(args);
    }
    @Override
    public void run(String... args) throws Exception {
        System.out.println("Hello");
        System.exit(0);
    }
}
}
```

Run as Main method for Output:(Ctrl+F11)

Hello