Maven AssignmentAssuming, you have GIT, tomcat, maven setup in your development environment;

do the following:

1.At command prompt, what will be command to generate archetypes? Run this command so that you can generate the list of archetypes.

mvn archetypes:generate

2.At command prompt, what will be command to create a new maven projectwith archetype whose description is following:

a. ”A basic starter template using springboot, jpa data, thymeleaf and MVC”

nl.delphinity : springboot-delphinity-archetype

b.Run this command with the number for correct archetype so that new default project is created with springboot, jpa, thymeleaf and MVC.

mvn archetype:generate -DarchetypeGroupId=nl.delphinity -DarchetypeArtifactId=springboot-delphinity-archetype -DgroupId=io.springboot.maven.Assingment -DartifactId=mavenAssingment

3.Once the project is created with the required template:

a.What will be the command to compile the project?

mvn clean Install

b.What will be the command to package the built project?

mvn package

c. What will be command to execute the jar which is created by above command?

mvn spring-boot:run

i.After executing the jar, you should be able to test the REST API in any browser window.



d.What will be command to clean all the compiled class files and jar files in target folder?

mvn clean

e.Change POM.xml to create war executable in place of jar executable in the above project.

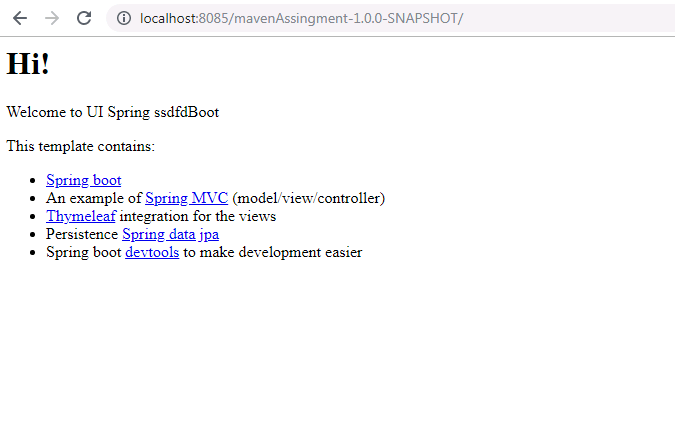
Run the command to create .war executable.



4.Install tomcat in local development environment.

a.Create manageruser in tomcat by creating the config entry in tomcat-users.xml

b.Deploy the war created in step 3.ein Apache Tomcatand access the REST API endpoint exposed by



the deployed war.

i.Hints:

1.You may need to add following dependency in the project

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-tomcat</artifactId>

<scope>provided</scope>

</dependency>

2.And, you may need to add following code to Spring Boot Main Class.

@SpringBootApplication

public class Application extends SpringBootServletInitializer{

public static void main(String[] args) {

SpringApplication.run(Application.class, args);

}

}

3.SpringBootServletInitializer is imported from

org.springframework.boot.web.servlet.support.SpringBootServletInitializer

5.Jetty is the light weight servlet container.Add jetty dependency to the POM.xml

a.You can get the Jetty dependency from

https://mvnrepository.com/artifact/org.eclipse.jetty/jetty-maven-plugin

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-web</artifactId>

<exclusions>

<exclusion>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-tomcat</artifactId>

</exclusion>

</exclusions>

</dependency>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-jetty</artifactId>

</dependency>

b. Execute the project using jetty

