

Spring boot First Example with configurations.

We need to follow some steps to configure spring boot project.

Pre-Requisite:

- Jdk 1.8
- Latest eclipse
- MySQL – required in future.

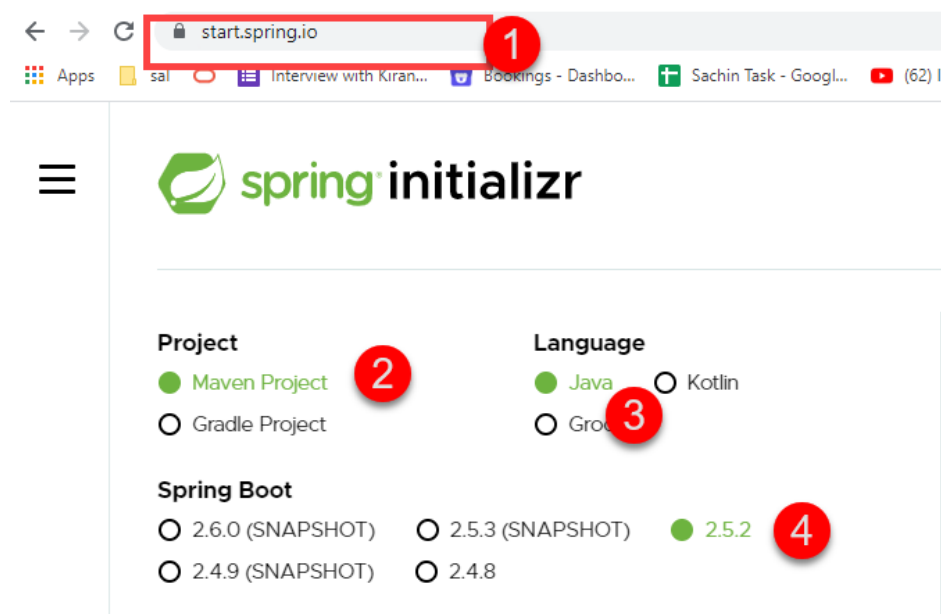
To install java and eclipse, follow this document, if you already have latest eclipse and java 8 skip this step.

<https://javabykiran.in/core-java/JBKSETUP001-java-eclipse-setup.pdf>

Step #1

Open website <https://start.spring.io/>

Select below details.



Project Metadata

- Group
 - This is a folder where all other spring boot project you want to store.
 - enter **com.javabykiran**
- Artifact
 - Your current project name.
 - enter **basicMicroserviceProject**
- Name
 - Any name
- Description
 - First basic microservice or REST webservice project
- Package name
 - com.javabykiran.controller
- Packaging
 - war: This is to create web project.
 - jar: This is to create API project.
 - select **jar**.
- Java
 - version 8 we will be using.
 - **version 8**

Click on dependencies.

Add web , devtools

The screenshot shows the Spring Initializr web application configuration page. Red circles with numbers 1 through 7 highlight specific elements:

- 1**: Maven Project (selected under Project)
- 2**: 2.5.2 (selected under Spring Boot)
- 3**: basicMicroserviceProject (in the Name field of Project Metadata)
- 4**: ADD DEPENDENCIES button
- 5**: Spring Web (selected under Dependencies)
- 6**: Spring Boot DevTools (selected under Dependencies)
- 7**: GENERATE button

The form includes sections for Project, Language, Spring Boot, Project Metadata, and Dependencies. The Project Metadata section contains fields for Group, Artifact, Name, Description, and Package name. The Dependencies section lists Spring Web and Spring Boot DevTools.

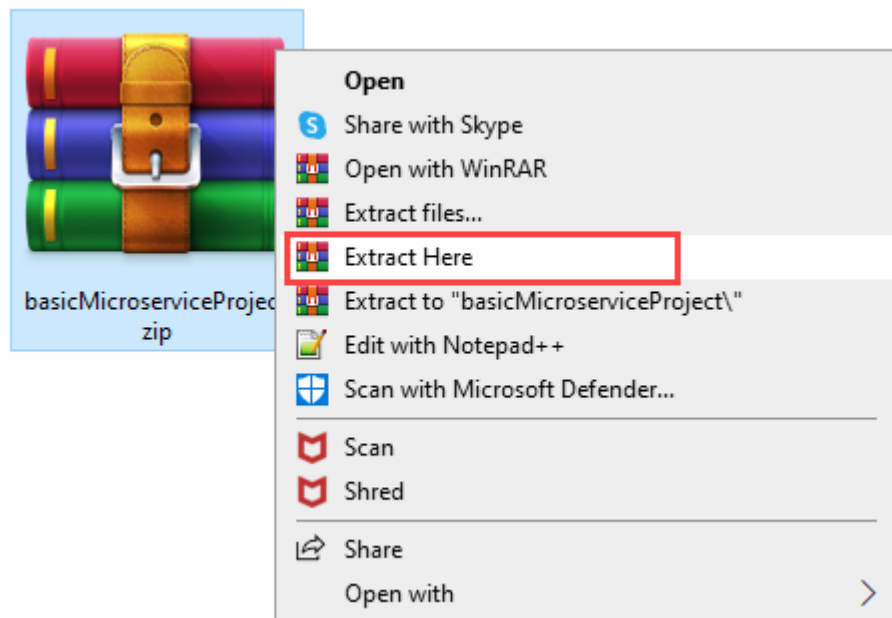
nts > syllabus > advance java syllabus > springboot > springbootproject



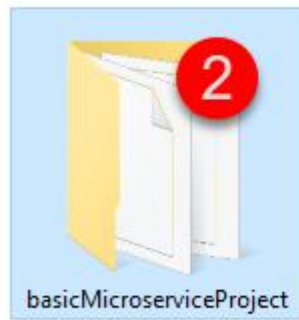
basicMicroserviceProject.
zip

Step #2

Extract project

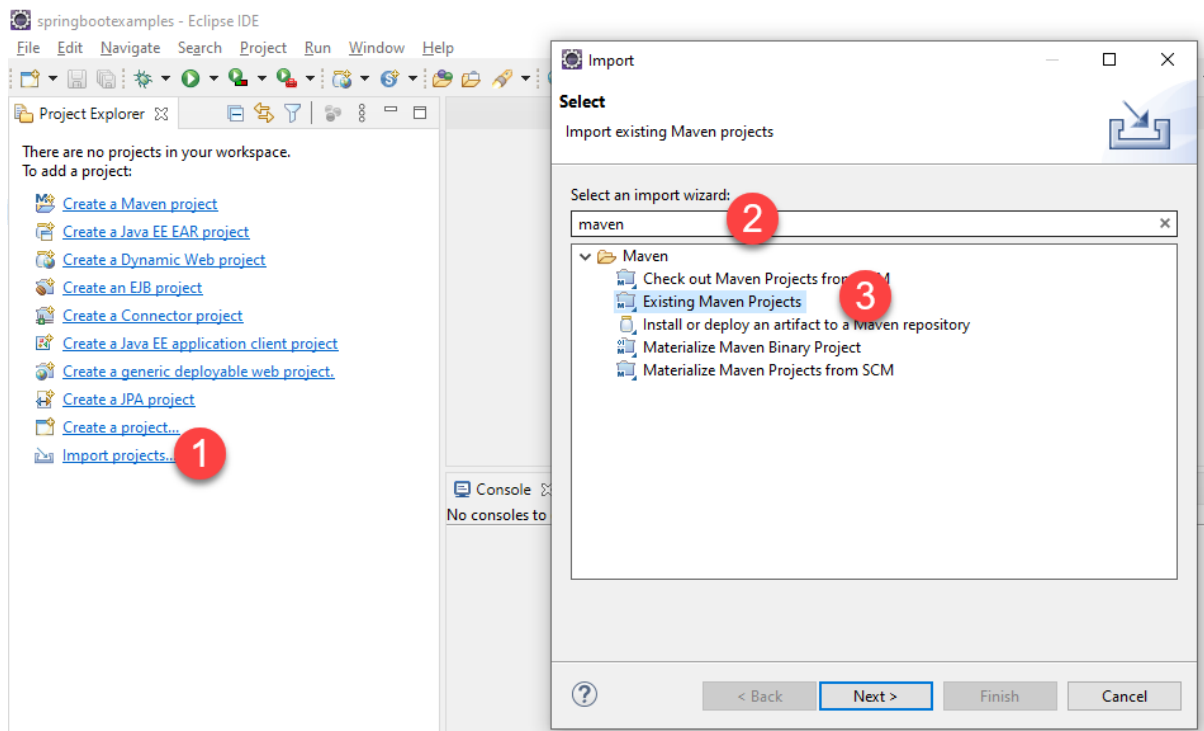


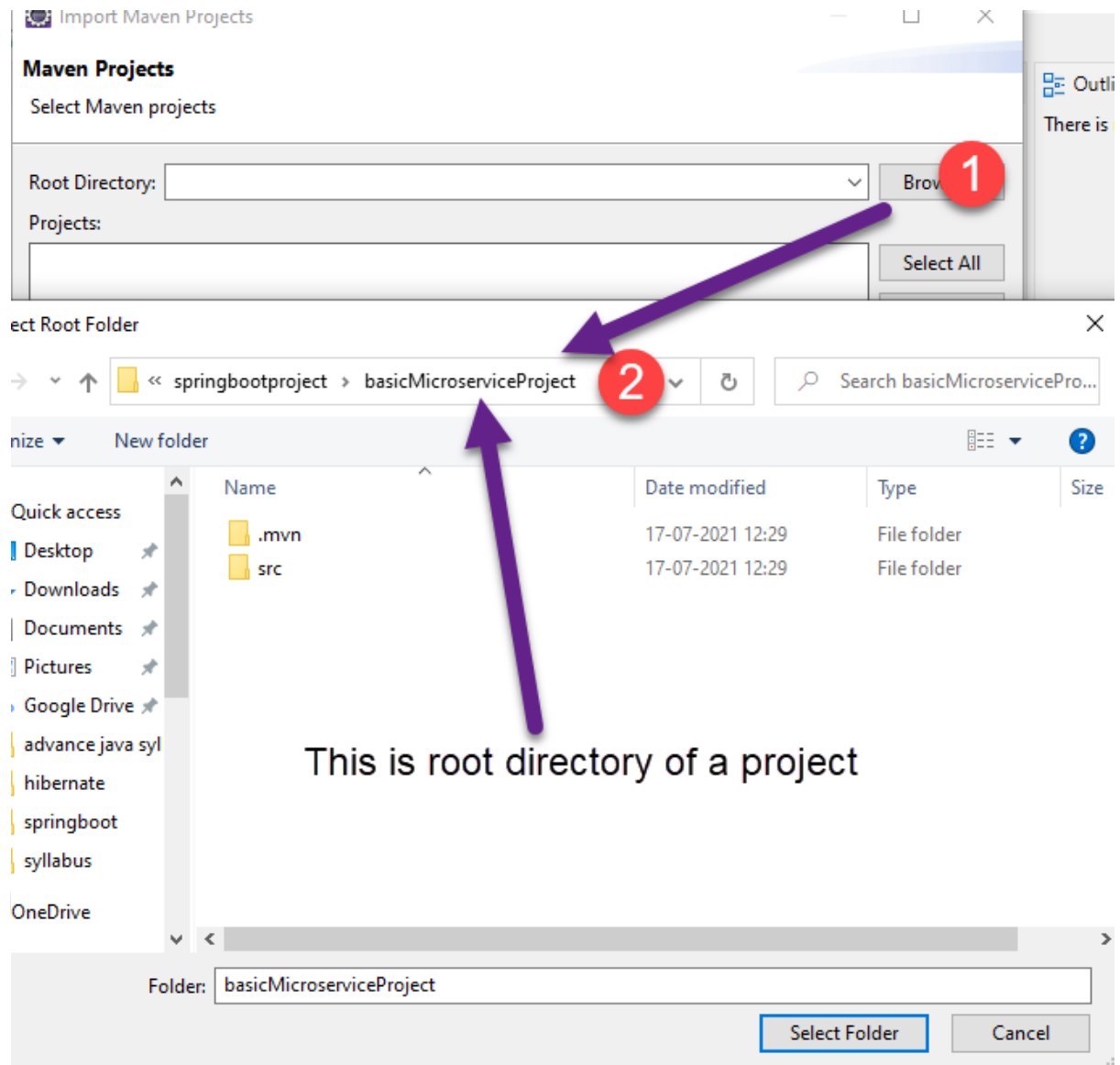
ts > syllabus > advance java syllabus > springboot > springbootproject



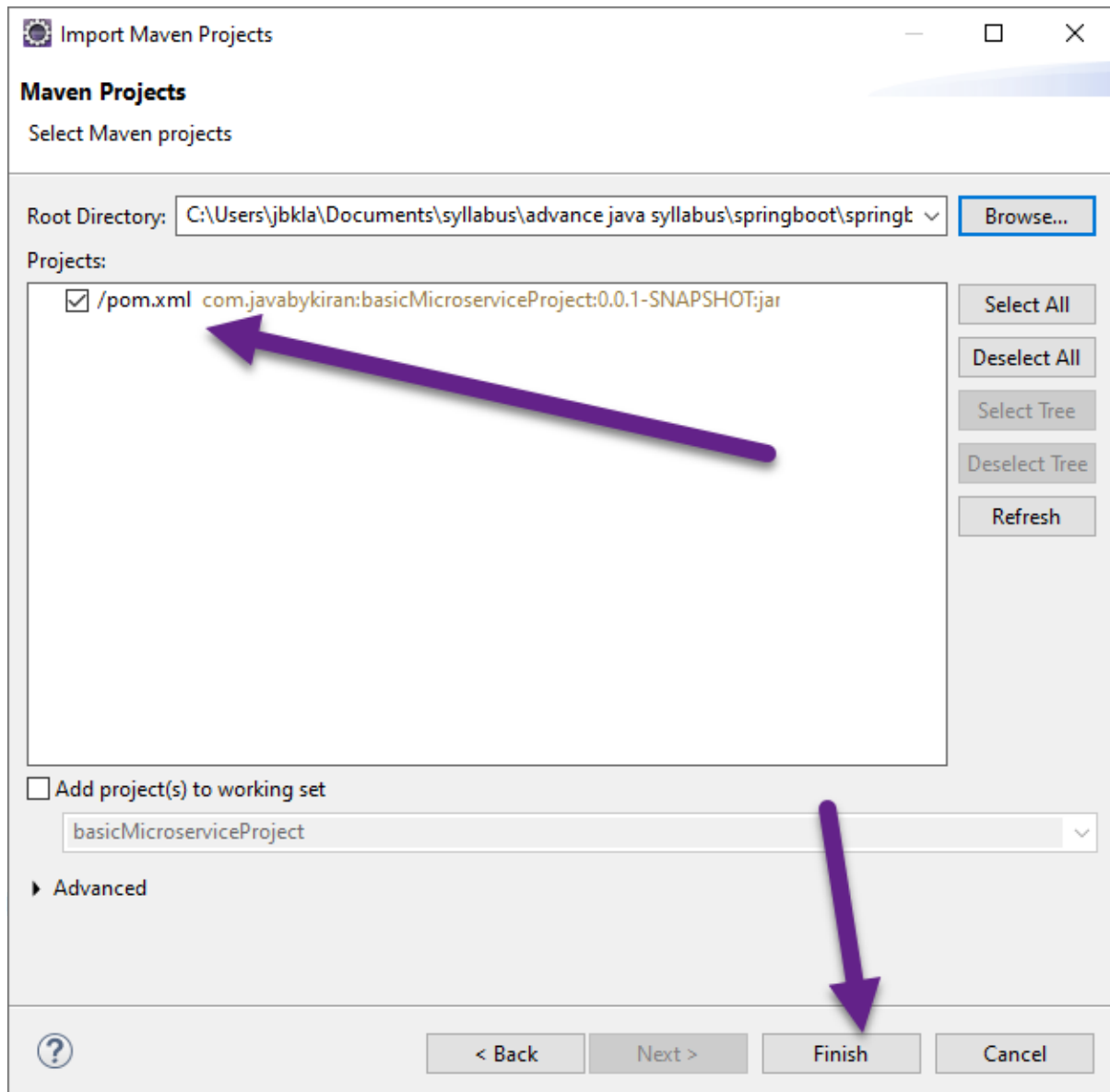
Step #3

Import project to eclipse.





Once you click on Select Folder you can see pom.xml will be imported.



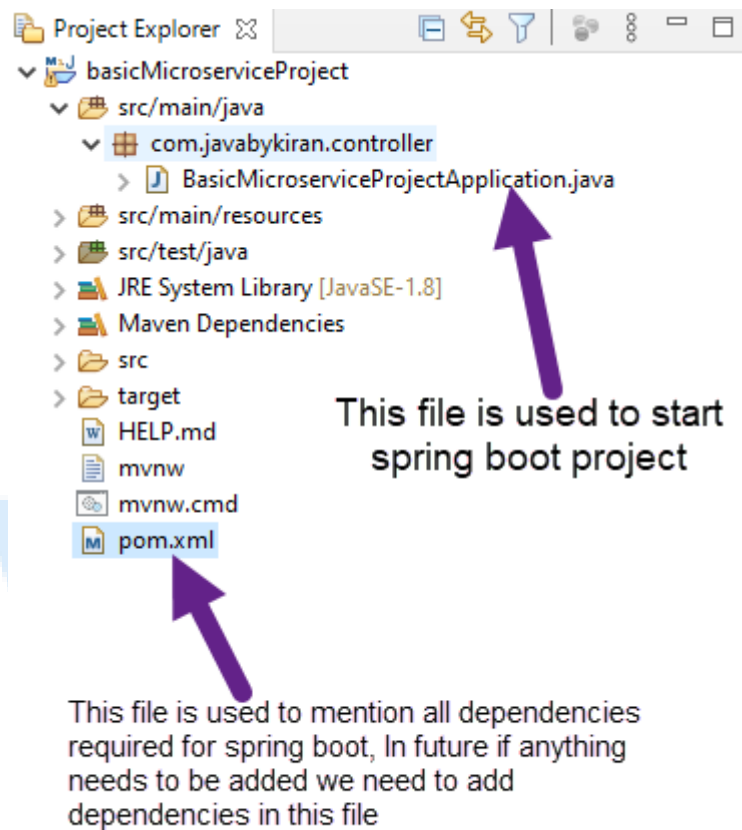
Once click on finish. Wait for dependencies to get downloaded in our project. It may take few minutes depending upon your internet speed.

Below is project structure after downloading all dependencies. Make sure you do not have any errors in problem windows. Most of the errors occur if we do not have proper version of java or maven or eclipse.

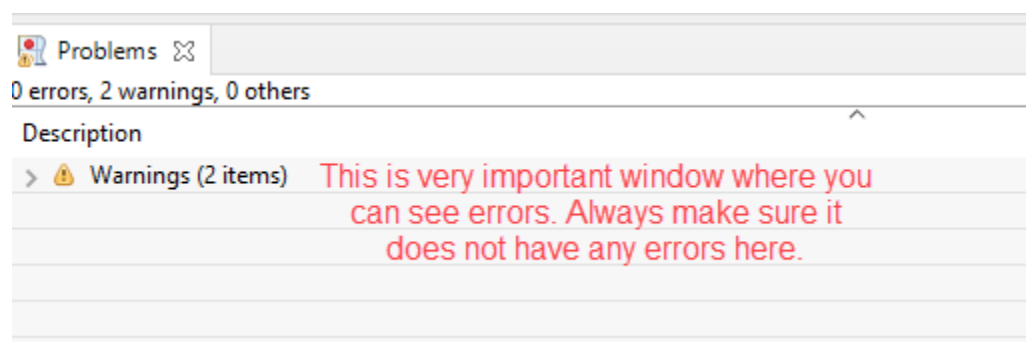
Open problem window – window – show view - problems.

Step #4

Check if out setup is correct.



This is important window as below. For everything you do in eclipse always keep this window open for seeing any errors.



Just for time being we will see how errors will look like.

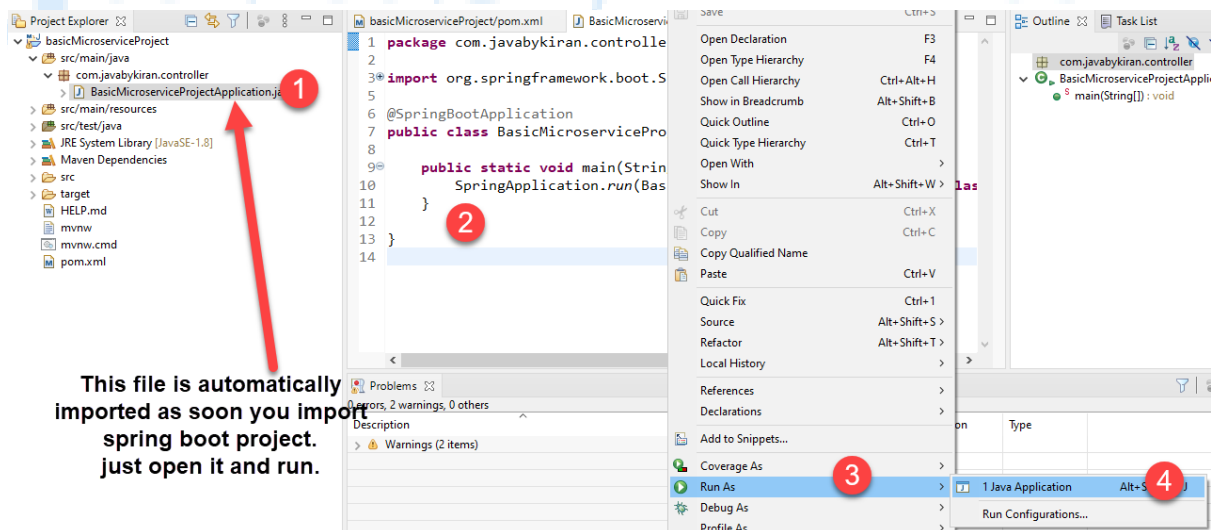
Problems					
1 error, 2 warnings, 0 others					
Description	Resource	Path	Location	Type	
Errors (1 item)					
Error resolving version for plugin 'org.springframework.boot'	pom.xml	/basicMicroserviceProject	line 1	Maven pom L...	
Warnings (2 items)					

This is a error in project

There are many project in eclipse. Here we get to know which project has error. Recommended is close all other inactive projects so that errors will not overlap here.

Step #5

It's time to check if our spring boot project is running correctly.



After running this console should have below output.



```
BasicMicroserviceProjectApplication [Java Application] C:\Program Files\Java\jdk-11.0.9\bin\javaw.exe (17-Jul-2021, 6:36:24 PM)

:: Spring Boot :: (v2.5.2)

2021-07-17 18:36:25.679 INFO 19160 --- [ restartedMain ] .j.c.BasicMicroserviceProjectApplication : Starting BasicMicroserviceProjectApplication using Java 11.0.9 on JavaByK
2021-07-17 18:36:25.682 INFO 19160 --- [ restartedMain ] .j.c.BasicMicroserviceProjectApplication : No active profile set, falling back to default profiles: default
2021-07-17 18:36:25.757 INFO 19160 --- [ restartedMain ] .e.DevToolsPropertyDefaultsPostProcessor : Devtools property defaults active! Set 'spring.devtools.add-properties' to
2021-07-17 18:36:25.757 INFO 19160 --- [ restartedMain ] .e.DevToolsPropertyDefaultsPostProcessor : For additional web related logging consider setting the 'logging.level.wel
2021-07-17 18:36:27.000 INFO 19160 --- [ restartedMain ] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port(s): 8080 (http)
2021-07-17 18:36:27.013 INFO 19160 --- [ restartedMain ] o.apache.catalina.core.StandardService : Starting service [Tomcat]
2021-07-17 18:36:27.014 INFO 19160 --- [ restartedMain ] org.apache.catalina.core.StandardEngine : Starting Servlet engine: [Apache Tomcat/9.0.48]
2021-07-17 18:36:27.094 INFO 19160 --- [ restartedMain ] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring embedded WebApplicationContext
2021-07-17 18:36:27.095 INFO 19160 --- [ restartedMain ] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization completed in 1337 ms
2021-07-17 18:36:27.497 INFO 19160 --- [ restartedMain ] o.s.b.d.a.OptionalLiveReloadServer : LiveReload server is running on port 35729
2021-07-17 18:36:27.523 INFO 19160 --- [ restartedMain ] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port(s): 8080 (http) with context path ''
2021-07-17 18:36:27.533 INFO 19160 --- [ restartedMain ] .j.c.BasicMicroserviceProjectApplication : Started BasicMicroserviceProjectApplication in 2.265 seconds (JVM running
```

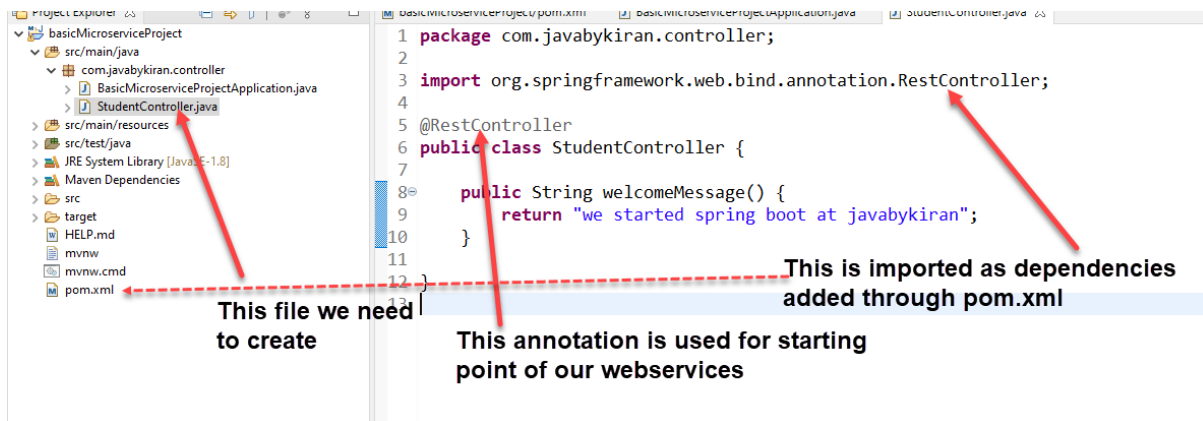
Now it is time to write simple first webservice.

Step #6

1. Create controller class.
 - a. This is a class which will be hit by client first.
 - b. This class takes a request and send response back to client.
 - i. Response can be json or xml format.
 - ii. Our focus will be on json format.
2. Use all annotations required for rest API.
 - a. @RestController
 - i. This annotation is used at class level.
 - ii. This is mandatory if we want to create rest webservice.
 - b. @RequestMapping
 - i. @GetMapping
 - ii. @PostMapping
 - iii. @PutMapping
 - iv. etc.

Step #7

Create CustomerController.java as below.

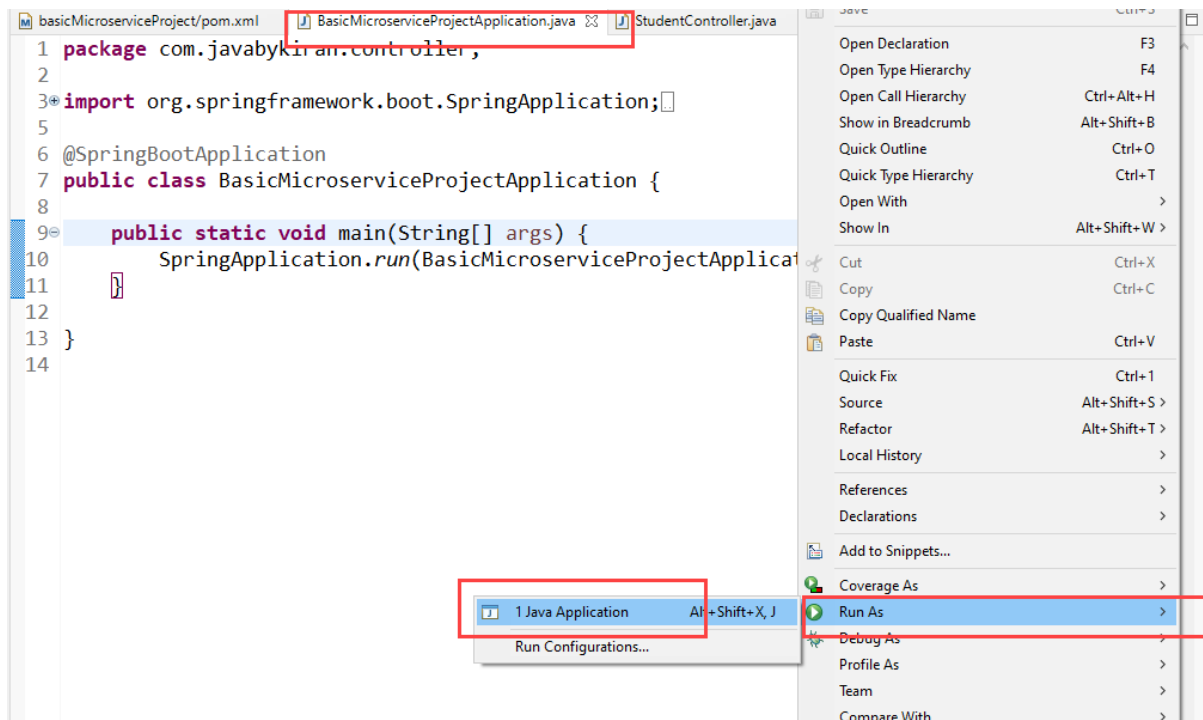


We have not yet added endpoint for our service. So add at method level @RequestMapping annotation



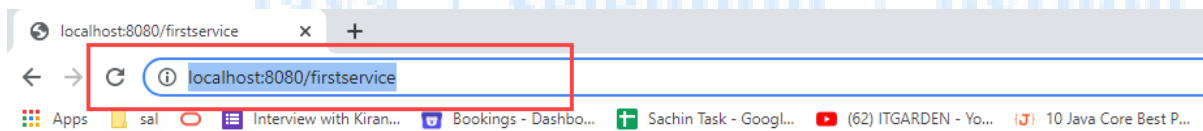
Go to browser. This is simple service we have created we need to run this service.

First start a project from main method.



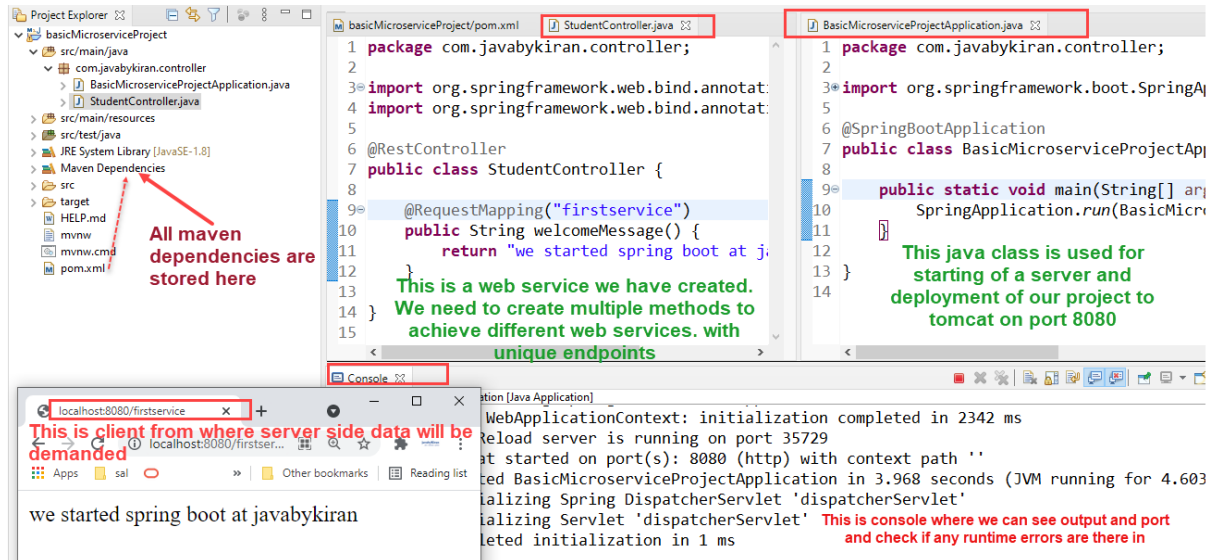
Check console if you see any exceptions if not go to browser.

In address bar hit url <http://localhost:8080/firstservice>



we started spring boot at javabykiran

Summary



Homework:

java | selenium | python

Read tutorial from below link to know in depth of spring boot. Screens used might be different than this, but concept is same.

<https://www.jbktutorials.com/spring-boot/introduction-to-spring-boot.php#gsc.tab=0>