**Creating Microservices for account and loan**

In this hands on exercises, we will create two microservices for a bank. One

microservice for handing accounts and one for handling loans.

Each microservice will be a specific independent Spring RESTful Webservice

maven project having it's own pom.xml. The only difference is that, instead of

having both account and loan as a single application, it is split into two

different applications. These webservices will be a simple service without any

backend connectivity.

Follow steps below to implement the two microservices:

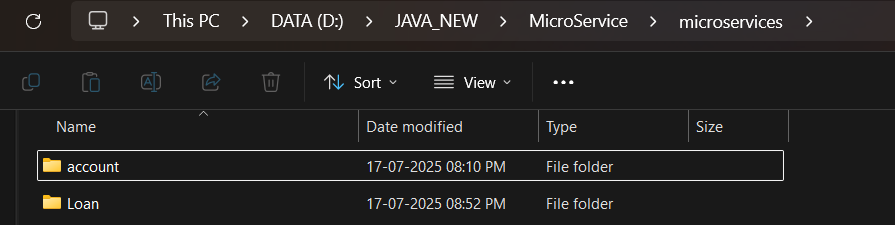
Account Microservice

 Create folder with employee id in D: drive

 Create folder named 'microservices' in the new folder created in

previous step. This folder will contain all the sample projects that we will

create for learning microservices.



 Open https://start.spring.io/ in browser

 Enter form field values as specified below:

o Group: com.cognizant

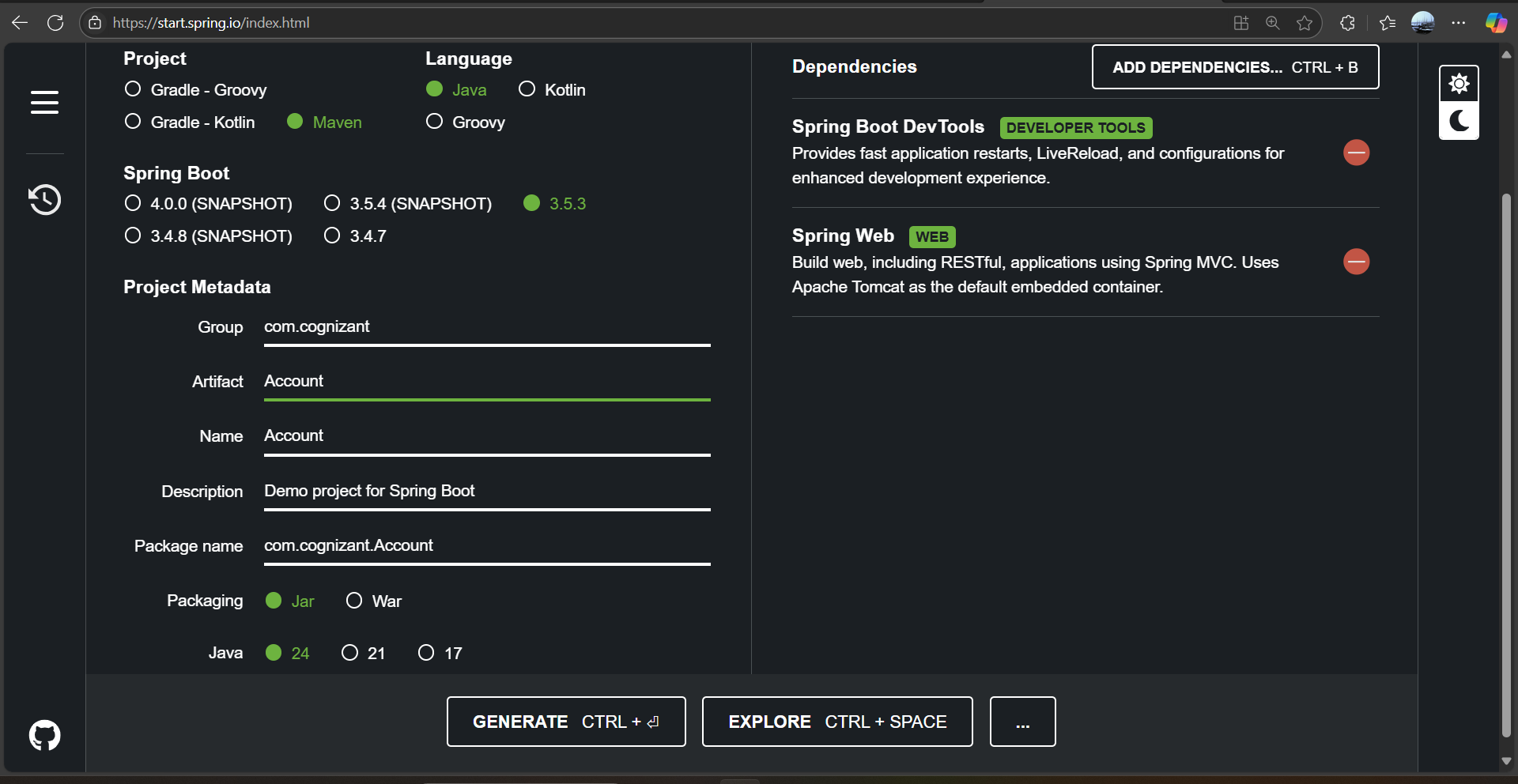
o Artifact: account

 Select the following modules

o Developer Tools > Spring Boot DevTools

o Web > Spring Web

 Click generate and download the zip file



 Extract 'account' folder from the zip and place this folder in the

'microservices' folder created earlier

 Open command prompt in account folder and build using mvn clean

package command

 Import this project in Eclipse and implement a controller method for

getting account details based on account number. Refer specification

below:

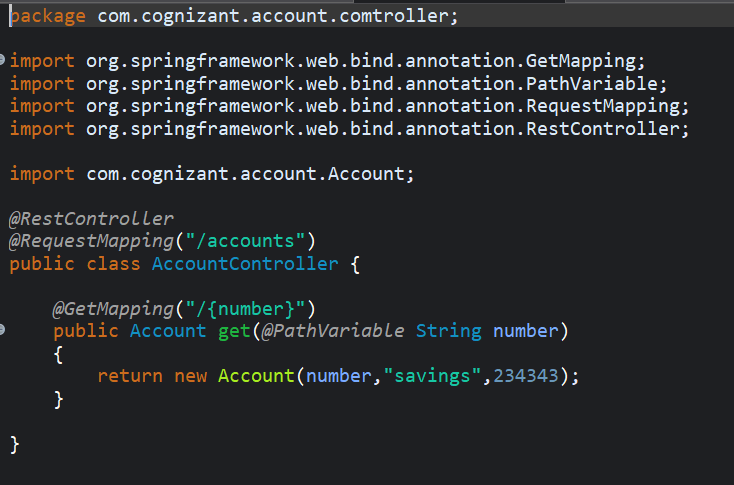
o Method: GET

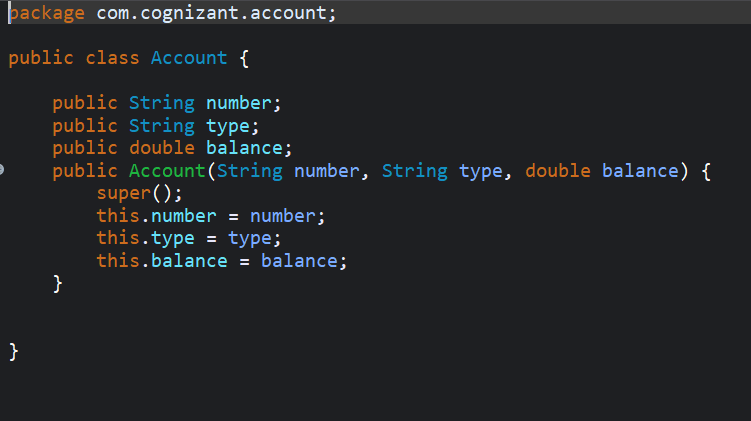
o Endpoint: /accounts/{number}

o Sample Response. Just a dummy response without any backend

connectivity.

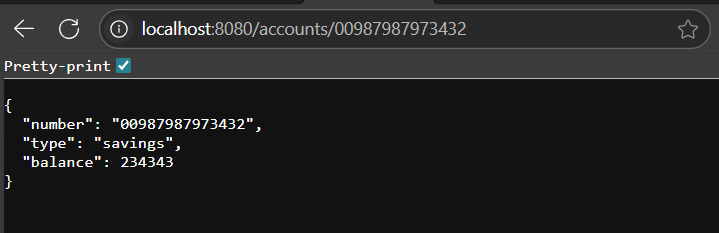
{ number: "00987987973432", type: "savings", balance: 234343 }





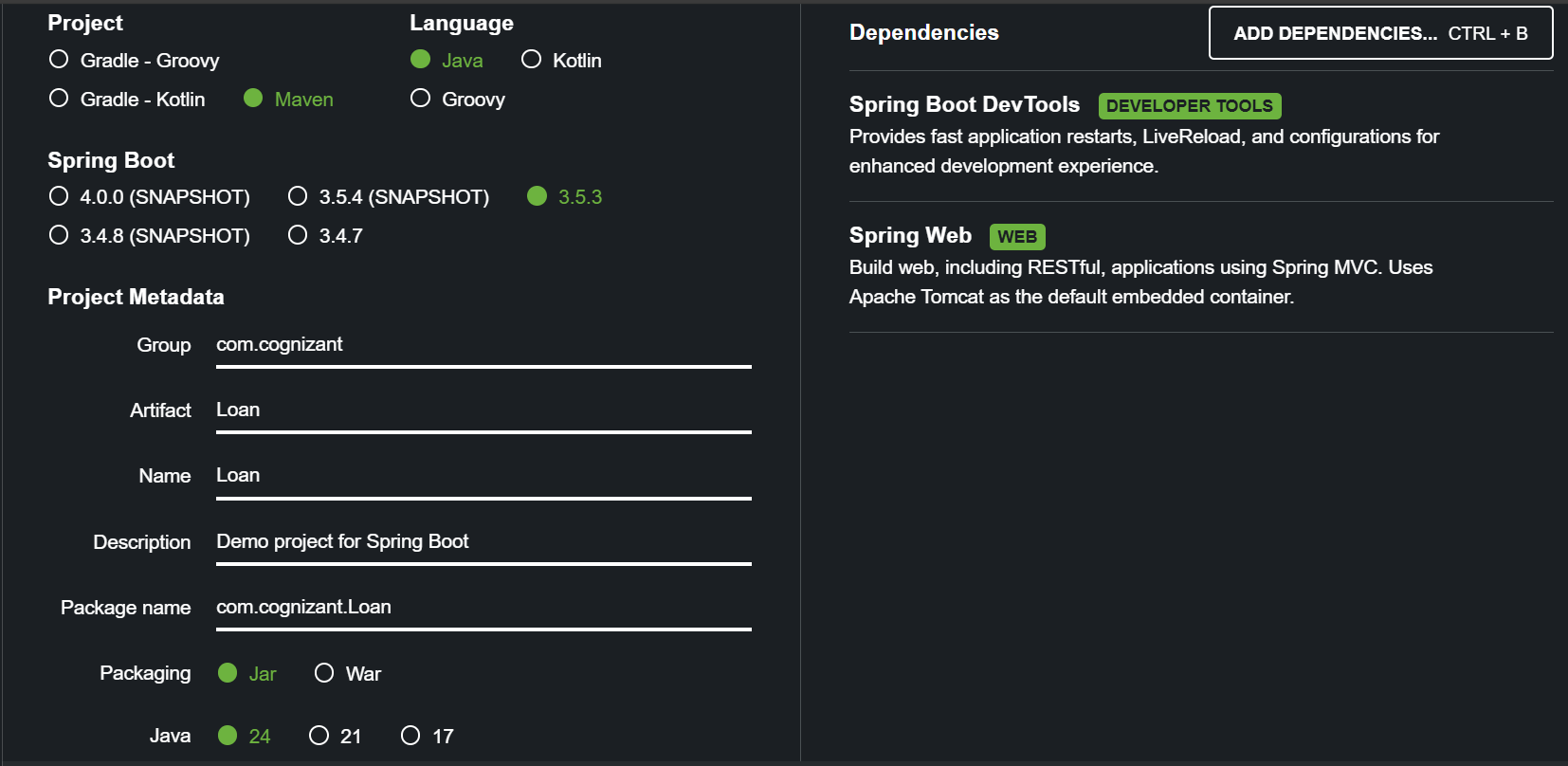
 Launch by running the application class and test the service in browser

Loan Microservice



 Follow similar steps specified for Account Microservice and implement a

service API to get loan account details

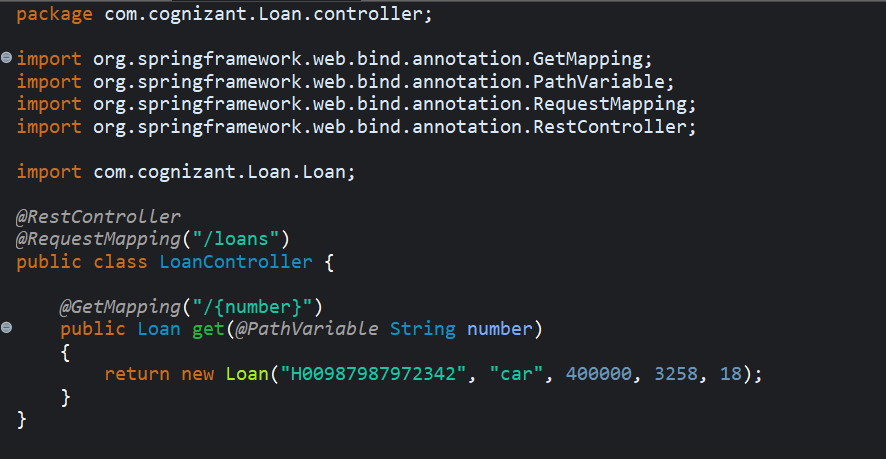


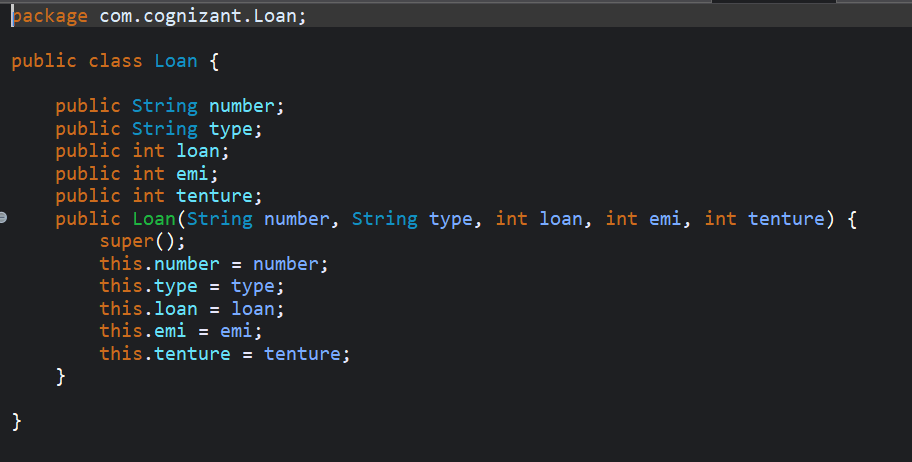
o Method: GET

o Endpoint: /loans/{number}

o Sample Response. Just a dummy response without any backend

connectivity.





{ number: "H00987987972342", type: "car", loan: 400000, emi: 3258, tenure:

18 }

 Launching this application by having account service already running

 This launch will fail with error that the bind address is already in use

 The reason is that each one of the service is launched with default port

number as 8080. Account service is already using this port and it is not

available for loan service.

**Error in console**

. \_\_\_\_ \_ \_\_ \_ \_

/\\ / \_\_\_'\_ \_\_ \_ \_(\_)\_ \_\_ \_\_ \_ \ \ \ \

( ( )\\_\_\_ | '\_ | '\_| | '\_ \/ \_` | \ \ \ \

\\/ \_\_\_)| |\_)| | | | | || (\_| | ) ) ) )

' |\_\_\_\_| .\_\_|\_| |\_|\_| |\_\\_\_, | / / / /

=========|\_|==============|\_\_\_/=/\_/\_/\_/

:: Spring Boot :: (v3.5.3)

2025-07-17T20:59:47.236+05:30 INFO 10532 --- [Loan] [ restartedMain] com.cognizant.Loan.LoanApplication : Starting LoanApplication using Java 24.0.1 with PID 10532 (D:\JAVA\_NEW\MicroService\microservices\Loan\target\classes started by ishit in D:\JAVA\_NEW\MicroService\microservices\Loan)

2025-07-17T20:59:47.241+05:30 INFO 10532 --- [Loan] [ restartedMain] com.cognizant.Loan.LoanApplication : No active profile set, falling back to 1 default profile: "default"

2025-07-17T20:59:47.334+05:30 INFO 10532 --- [Loan] [ restartedMain] .e.DevToolsPropertyDefaultsPostProcessor : Devtools property defaults active! Set 'spring.devtools.add-properties' to 'false' to disable

2025-07-17T20:59:47.334+05:30 INFO 10532 --- [Loan] [ restartedMain] .e.DevToolsPropertyDefaultsPostProcessor : For additional web related logging consider setting the 'logging.level.web' property to 'DEBUG'

2025-07-17T20:59:48.573+05:30 INFO 10532 --- [Loan] [ restartedMain] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port 8080 (http)

2025-07-17T20:59:48.598+05:30 INFO 10532 --- [Loan] [ restartedMain] o.apache.catalina.core.StandardService : Starting service [Tomcat]

2025-07-17T20:59:48.598+05:30 INFO 10532 --- [Loan] [ restartedMain] o.apache.catalina.core.StandardEngine : Starting Servlet engine: [Apache Tomcat/10.1.42]

2025-07-17T20:59:48.651+05:30 INFO 10532 --- [Loan] [ restartedMain] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring embedded WebApplicationContext

2025-07-17T20:59:48.652+05:30 INFO 10532 --- [Loan] [ restartedMain] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization completed in 1316 ms

2025-07-17T20:59:49.133+05:30 WARN 10532 --- [Loan] [ restartedMain] o.s.b.d.a.OptionalLiveReloadServer : Unable to start LiveReload server

2025-07-17T20:59:49.159+05:30 WARN 10532 --- [Loan] [ restartedMain] ConfigServletWebServerApplicationContext : Exception encountered during context initialization - cancelling refresh attempt: org.springframework.context.ApplicationContextException: Failed to start bean 'webServerStartStop'

2025-07-17T20:59:49.183+05:30 INFO 10532 --- [Loan] [ restartedMain] .s.b.a.l.ConditionEvaluationReportLogger :

Error starting ApplicationContext. To display the condition evaluation report re-run your application with 'debug' enabled.

2025-07-17T20:59:49.212+05:30 ERROR 10532 --- [Loan] [ restartedMain] o.s.b.d.LoggingFailureAnalysisReporter :

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

APPLICATION FAILED TO START

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Description:

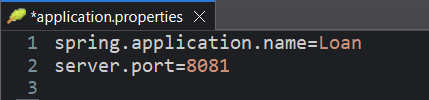
Web server failed to start. Port 8080 was already in use.

Action:

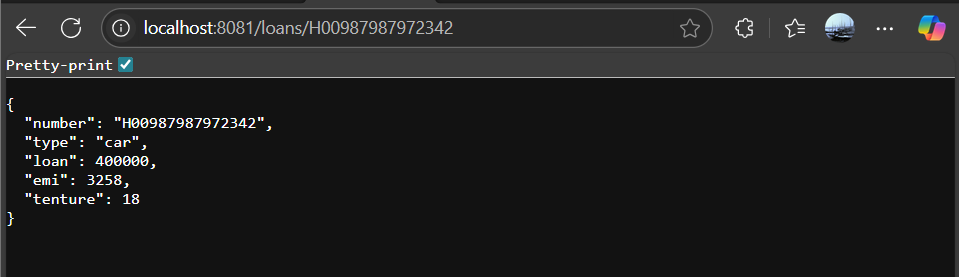
Identify and stop the process that's listening on port 8080 or configure this application to listen on another port.

 Include "server.port" property with value 8081 and try launching the

Application



 Test the service with 8081 port



Now we have two microservices running on different ports.