**Superset ID: 6394725**

**Microservices with Spring Boot 3 and Spring Cloud**

**Hands on:**

**Creating Microservices for account and loan**

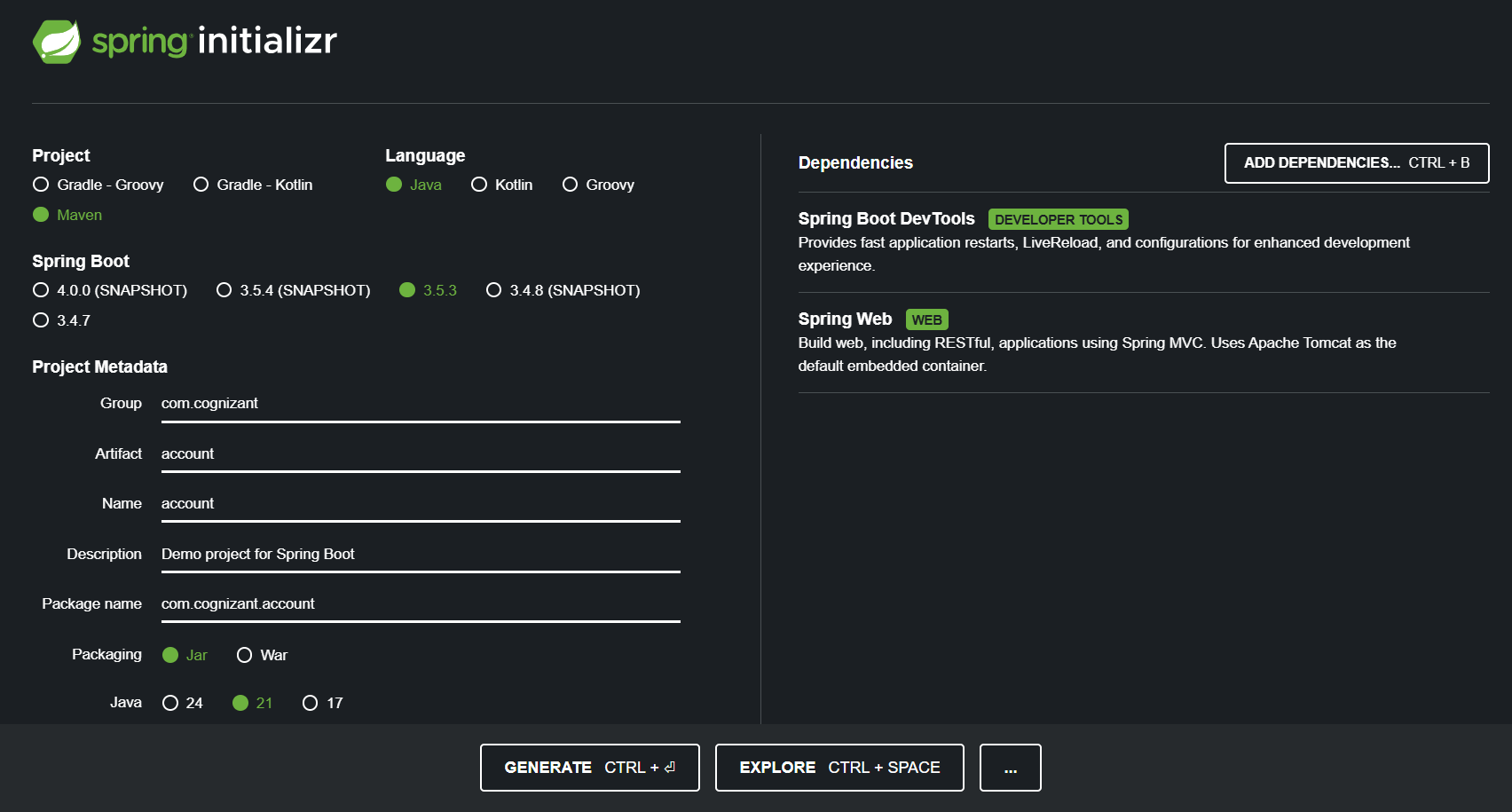
In this hands on exercises, we will create two microservices for a bank. One microservice for handing accounts and one for handling loans. Each microservice will be a specific independent Spring RESTful Webservice maven project having it's own pom.xml. The only difference is that, instead of having both 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.

**Step-1: Create folders**

1. Open D: drive
2. Create a folder with your **employee ID** (e.g., D:\22341A05A1)
3. Inside that, create a folder named: microservices
4. Final path: D:\22341A05A1\microservices

**Step-2: Create Account Microservice**

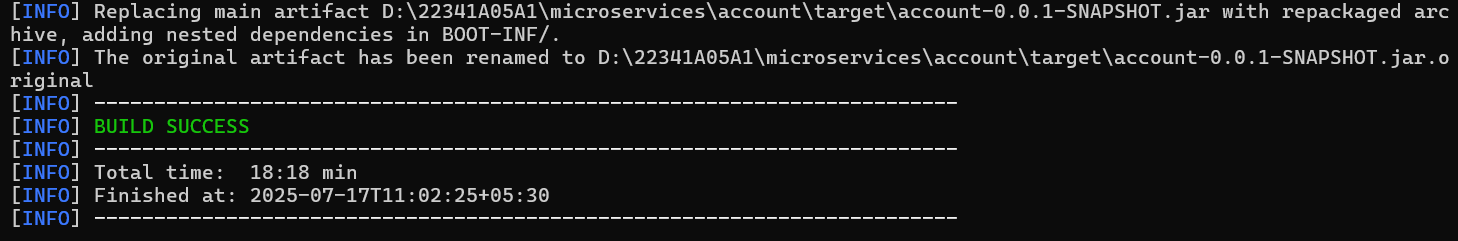
1. **Generate Spring Boot Project**
   1. Go to: [https://start.spring.io](https://start.spring.io/)
   2. Fill:
      * **Group**: com.cognizant
      * **Artifact**: account
   3. Add Dependencies:
      * Spring Boot DevTools
      * Spring Web
   4. Click **Generate**, download and extract the zip
   5. Place the extracted account folder into D:\22341A05A1\microservices



### **Build the Account Service**

* Open Command Prompt and run:

cd D:\22341A05A1\microservices\account  
mvn clean package



1. **Import into Eclipse**
   1. Open Eclipse
   2. Import the Maven project:
      1. File > Import > Existing Maven Projects
      2. Select the account folder
2. **Create Controller**

* Create a class AccountController.java inside com.cognizant.account.controller package:

package com.cognizant.account.controller;

import org.springframework.web.bind.annotation.\*;

import java.util.Map;

*@RestController*

*@RequestMapping*("/accounts")

public class AccountController {

*@GetMapping*("/{number}")

public Map<String, Object> getAccountDetails(*@PathVariable* String number) {

return Map.*of*(

"number", number,

"type", "savings",

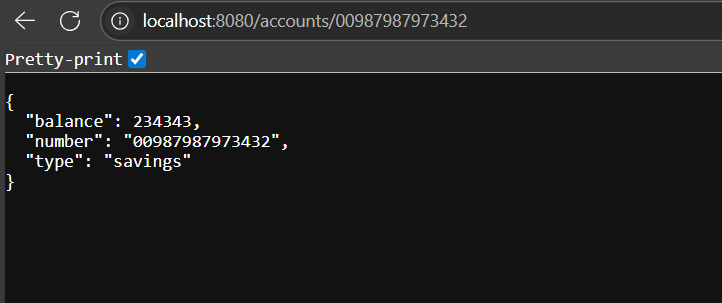
"balance", 234343

);

}

}

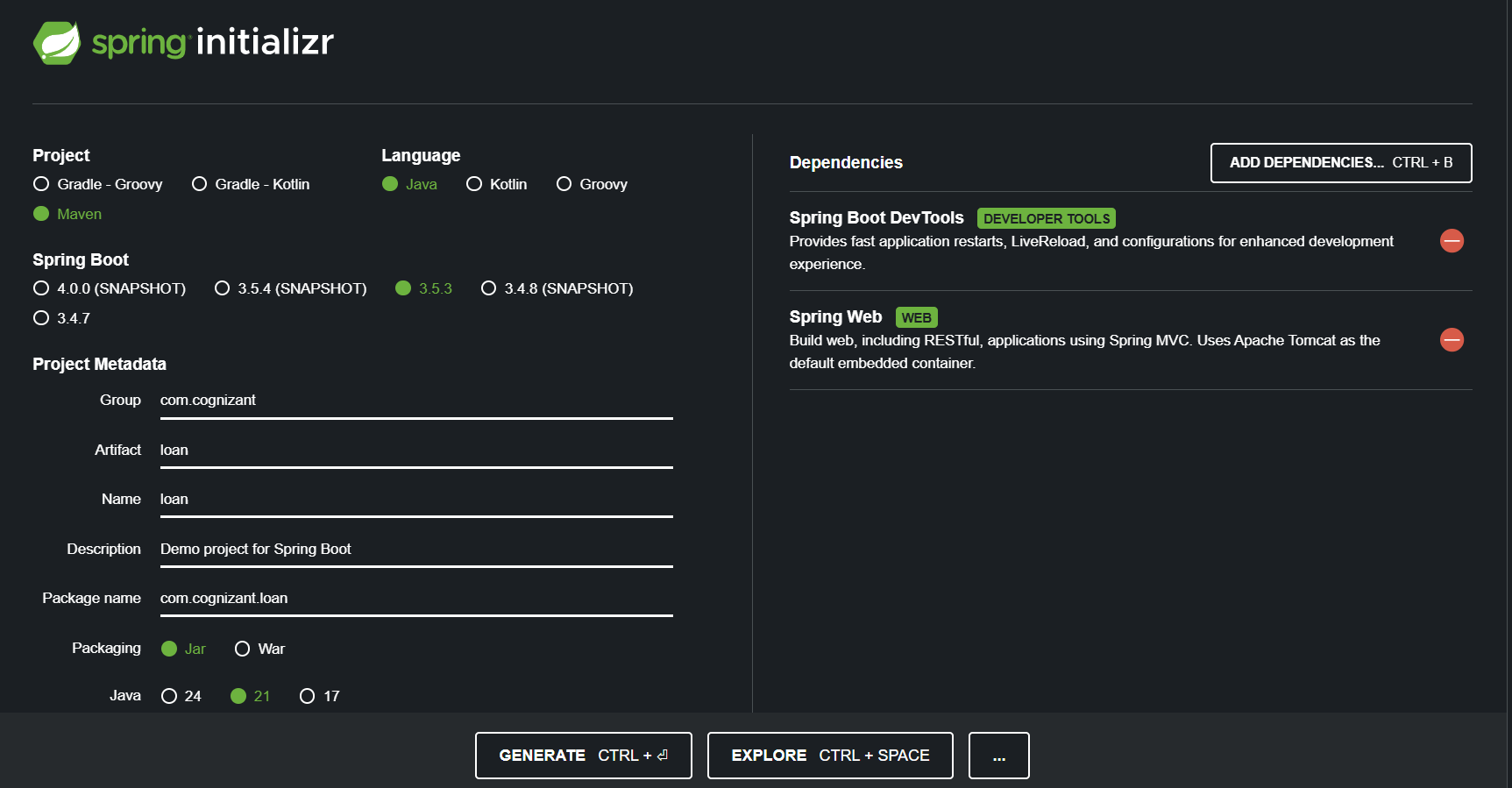
1. **Run and Test**
   1. Run the main class AccountApplication.java
   2. Open browser:  
       <http://localhost:8080/accounts/00987987973432>



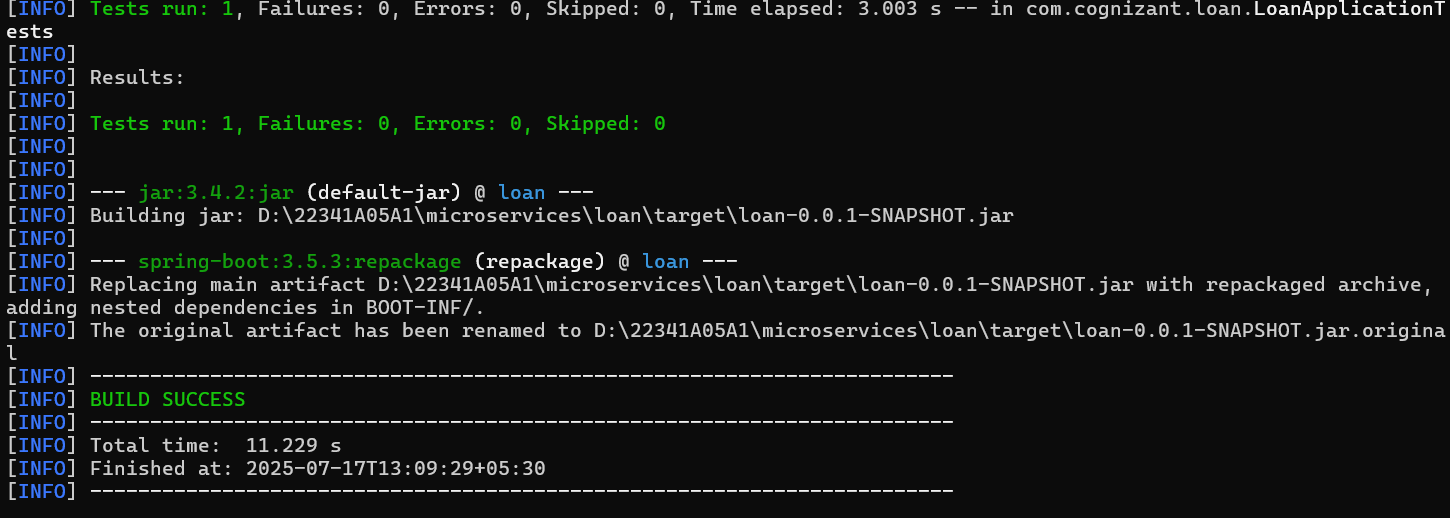
**Step-3: Create Loan Microservice**

### **Generate Spring Boot Project**

* Go to: [https://start.spring.io](https://start.spring.io/)
* Fill:
  + **Group**: com.cognizant
  + **Artifact**: loan
* Add Dependencies:
  + Spring Boot DevTools
  + Spring Web
* Download, extract, and place it in:  
   D:\22341A05A1\microservices\loan



1. **Build the Loan Service**
   1. cd D:\22341A05A1\microservices\loan  
      mvn clean package



1. **Import into Eclipse**
   1. Use: File > Import > Existing Maven Projects
   2. Select the loan project

### **Change Port to Avoid Conflict**

* Open src/main/resources/application.properties
* Add: server.port=8081

1. **Create Controller**
   1. Create class LoanController.java in com.cognizant.loan.controller package:

package com.cognizant.loan.controller;

import org.springframework.web.bind.annotation.\*;

import java.util.Map;

*@RestController*

*@RequestMapping*("/loans")

public class LoanController {

*@GetMapping*("/{number}")

public Map<String, Object> getLoanDetails(*@PathVariable* String number) {

return Map.*of*(

"number", number,

"type", "car",

"loan", 400000,

"emi", 3258,

"tenure", 18

);

}

}

1. **Run and Test**
   1. Run the LoanApplication.java main class
   2. Open browser:  
       <http://localhost:8081/loans/H00987987972342>

