**Creating Microservices for account and loan**

In these hands-on exercises, we will create two microservices for a bank. One microservice for handing accounts and one for handling loans.

**1. Introduction**

This project demonstrates how to build two independent microservices using Spring Boot:

* **Account Service** on port 8080
* **Loan Service** on port 8081

**2. Project Structure**

microservices/

├── account/

│ ├── src/

│ └── pom.xml

└── loan/

├── src/

└── pom.xml

Each service contains:

controller package with a REST controller

application.properties to define port

**3. Dependencies (in pom.xml)**

**Add the following common dependencies to both account and loan:**

<dependencies>

<dependency>

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

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

</dependency>

<dependency>

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

<artifactId>spring-boot-devtools</artifactId>

<scope>runtime</scope>

<optional>true</optional>

</dependency>

<dependency>

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

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

<scope>test</scope>

</dependency>

</dependencies>

**4. Microservice Configuration**

**account/src/main/resources/application.properties :** server.port=8080

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

**5. Controller Implementation**

**AccountController.java**

package com.cognizant.account.controller;

import org.springframework.web.bind.annotation.GetMapping;

import org.springframework.web.bind.annotation.RestController;

@RestController

public class AccountController {

@GetMapping("/account")

public String getAccountDetails() {

return "Account details from Account Microservice";

}

}

**LoanController.java**

package com.cognizant.loan.controller;

import org.springframework.web.bind.annotation.GetMapping;

import org.springframework.web.bind.annotation.RestController;

@RestController

public class LoanController {

@GetMapping("/loan")

public String getLoanDetails() {

return "Loan details from Loan Microservice";

}

}

**6. Build & Run**

Run from terminal or IDE: mvn clean install

**Run Services**

**Account:** mvn spring-boot:run

**Loan**: mvn spring-boot:run

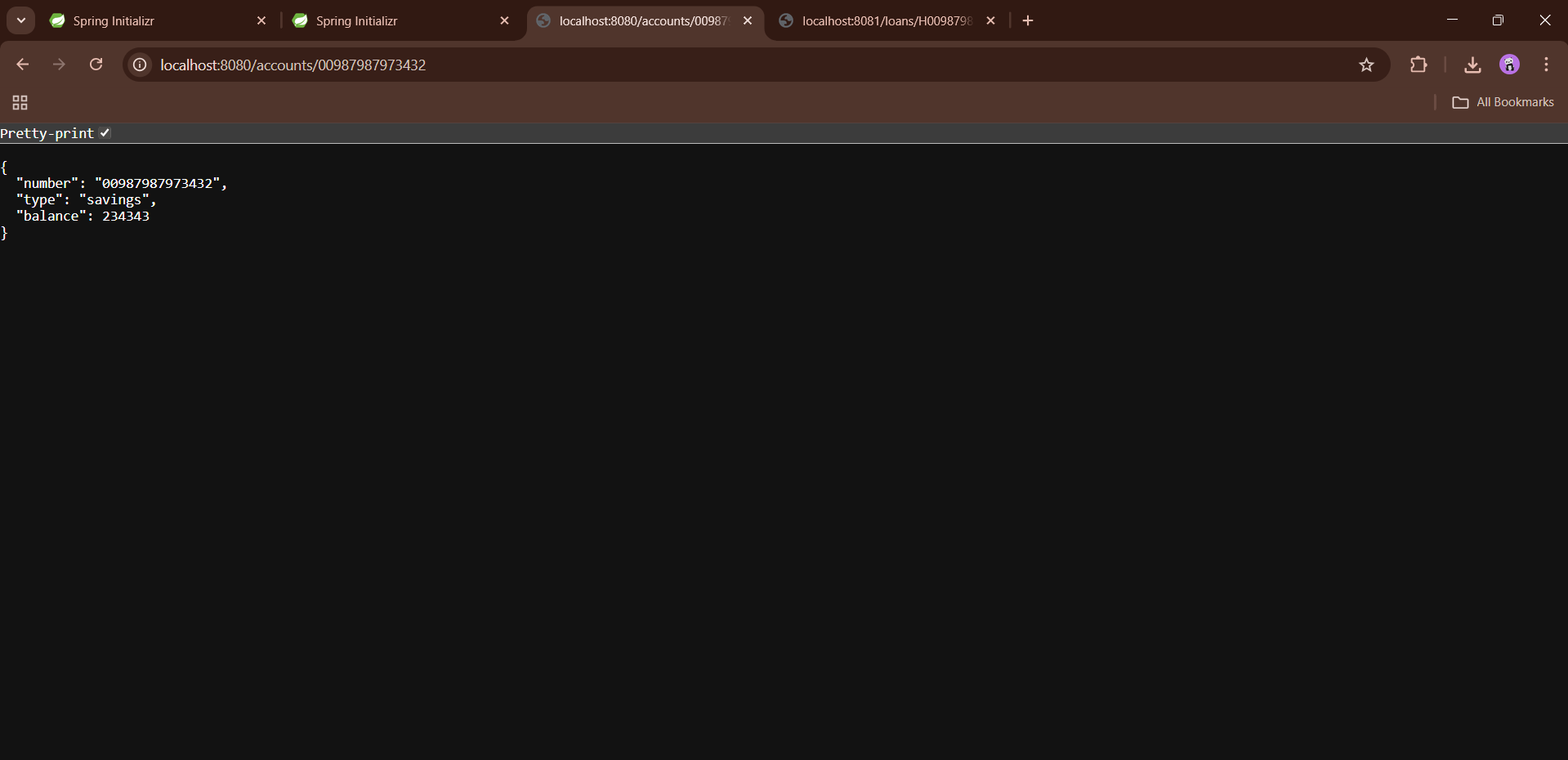
**7. Testing**

Using a browser or Postman:

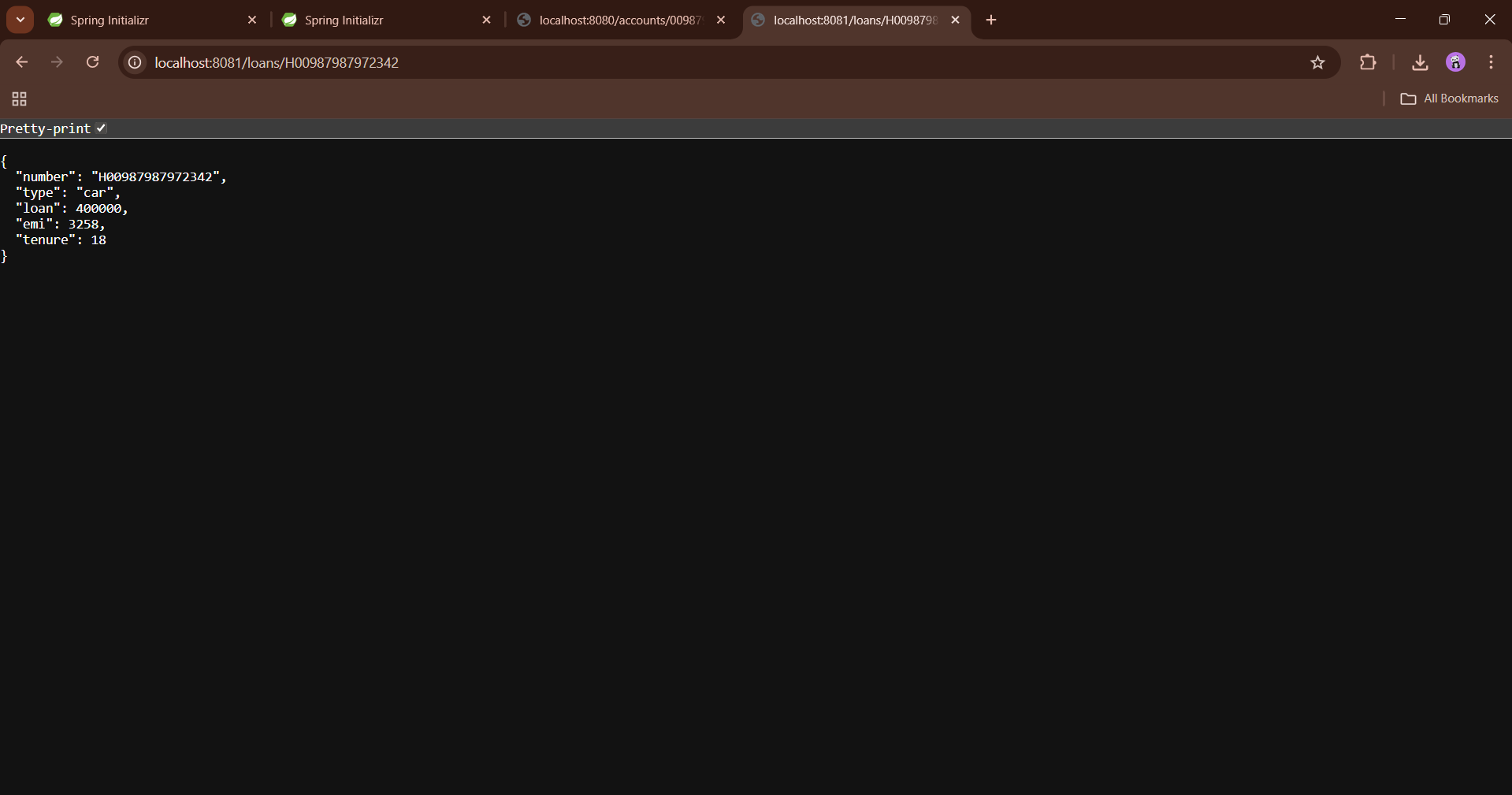
* GET http://localhost:8080/account  
  ➝ Account details from Account Microservice
* GET http://localhost:8081/loan  
  ➝ Loan details from Loan Microservice

**OUTPUT:**

**Account**

****

**Loan:**

****