**WEEK-5 MICROSERVICES**

**MICROSERVICES WITH API GATEWAY**

**SUPERSET ID:6393676**

**Creating Microservices for account and loan**

**SOLUTION**

**Account MIcroservices**

**AccountApplication.java**

package com.cognizant.account;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication

public class AccountApplication {

public static void main(String[] args) {

SpringApplication.run(AccountApplication.class, args);

}

}

**AccountController.java**

**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

);

}

}

**Pom.xml**

<?xml version="1.0" encoding="UTF-8"?>

<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 https://maven.apache.org/xsd/maven-4.0.0.xsd">

<modelVersion>4.0.0</modelVersion>

<parent>

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

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

<version>3.5.3</version>

<relativePath/> <!-- lookup parent from repository -->

</parent>

<groupId>com.cognizant</groupId>

<artifactId>account</artifactId>

<version>0.0.1-SNAPSHOT</version>

<name>account</name>

<description>Demo project for Spring Boot</description>

<url/>

<licenses>

<license/>

</licenses>

<developers>

<developer/>

</developers>

<scm>

<connection/>

<developerConnection/>

<tag/>

<url/>

</scm>

<properties>

<java.version>17</java.version>

</properties>

<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>

<build>

<plugins>

<plugin>

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

<artifactId>spring-boot-maven-plugin</artifactId>

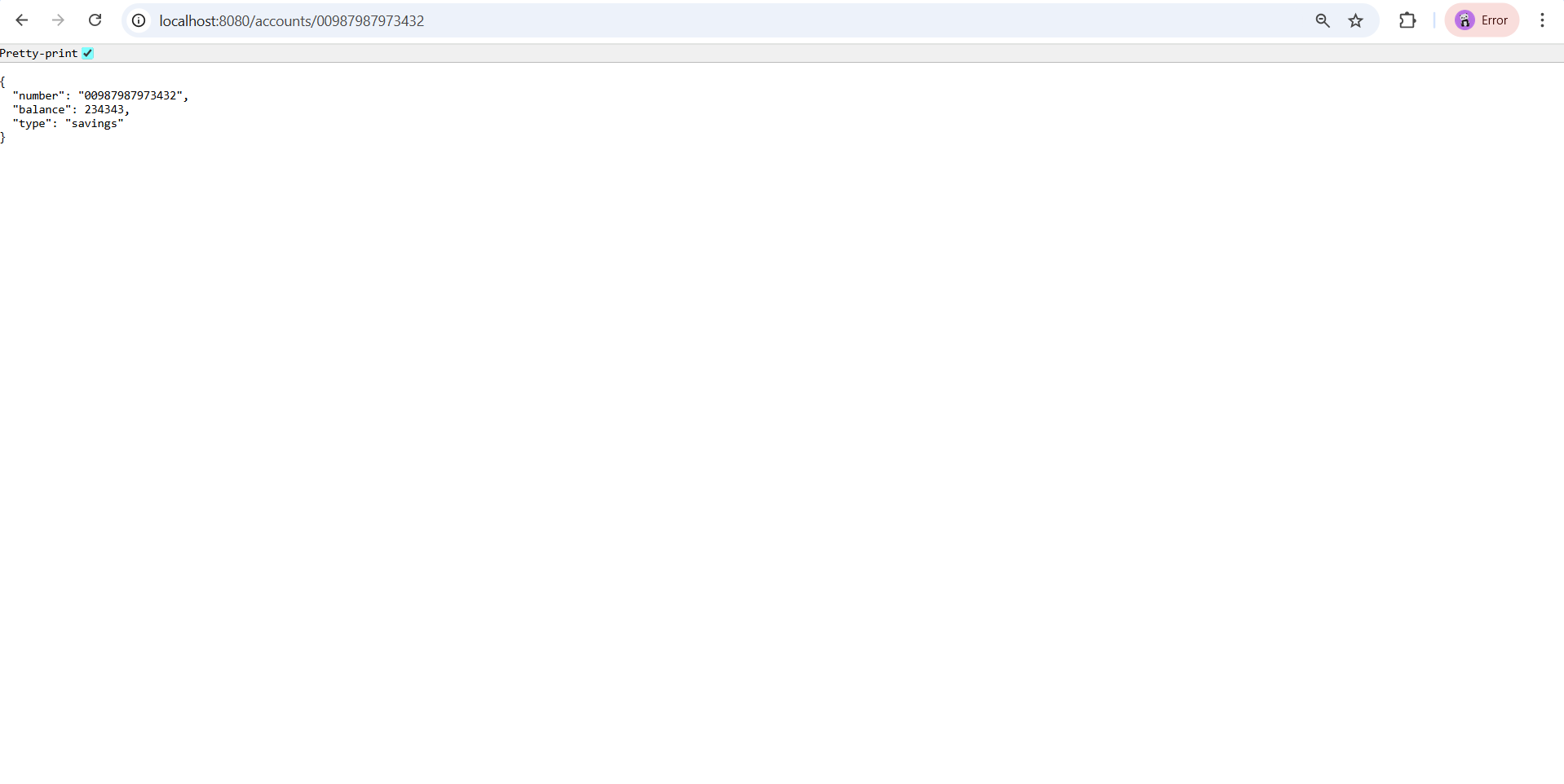
</plugin>

</plugins>

</build>

</project>

**OUTPUT**

**Loan Microservice**

**LoanApplication.java**

package com.cognizant.loan;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication

public class LoanApplication {

public static void main(String[] args) {

SpringApplication.run(LoanApplication.class, args);

}

}

**LoanController.java**

package com.cognizant.loan.controller;

import com.cognizant.loan.model.Loan;

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

@RestController

@RequestMapping("/loans")

public class LoanController {

@GetMapping("/{number}")

public Loan getLoanDetails(@PathVariable String number) {

// Dummy response

return new Loan(

number,

"car",

400000,

3258,

18

);

}

}

**Loan.java**

package com.cognizant.loan.model;

public class Loan {

private String number;

private String type;

private double loan;

private double emi;

private int tenure;

public Loan(String number, String type, double loan, double emi, int tenure) {

this.number = number;

this.type = type;

this.loan = loan;

this.emi = emi;

this.tenure = tenure;

}

// Getters and Setters

public String getNumber() { return number; }

public void setNumber(String number) { this.number = number; }

public String getType() { return type; }

public void setType(String type) { this.type = type; }

public double getLoan() { return loan; }

public void setLoan(double loan) { this.loan = loan; }

public double getEmi() { return emi; }

public void setEmi(double emi) { this.emi = emi; }

public int getTenure() { return tenure; }

public void setTenure(int tenure) { this.tenure = tenure; }

}

**application.properties**

server.port=8081

**OUTPUT:**

