Creating Microservices for account and loan

Microservice:Account

File:AccountApplication.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);

}

}

File:Account.java

package com.cognizant.account;

public class Account {

private String number;

private String type;

private double balance;

public Account(String number, String type, double balance) {

this.number = number;

this.type = type;

this.balance = balance;

}

public String getType() {

return type;

}

public String getNumber() {

return number;

}

public double getBalance() {

return balance;

}

public void setType(String type) {

this.type = type;

}

public void setNumber(String number) {

this.number = number;

}

public void setBalance(double balance) {

this.balance = balance;

}

}

File:AccountController.java

package com.cognizant.account.controller;

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

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

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

import com.cognizant.account.Account;

*@RestController*

public class AccountController {

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

public Account getAccount(*@PathVariable* String number) {

//Sample Response. Just a dummy response without any backend.

return new Account(number,"savings",234343);

}}

Microservice:Loan

File: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);

}

}

File:Loan.java

package com.cognizant.loan;

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;

}

public int getTenure() {

return tenure;

}

public void setTenure(int tenure) {

this.tenure = tenure;

}

public double getEmi() {

return emi;

}

public void setEmi(double emi) {

this.emi = emi;

}

public double getLoan() {

return loan;

}

public void setLoan(double loan) {

this.loan = loan;

}

public String getType() {

return type;

}

public void setType(String type) {

this.type = type;

}

public String getNumber() {

return number;

}

public void setNumber(String number) {

this.number = number;

}

}

File:LoanController.java

package com.cognizant.loan.controller;

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

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

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

import com.cognizant.loan.Loan;

*@RestController*

public class LoanController {

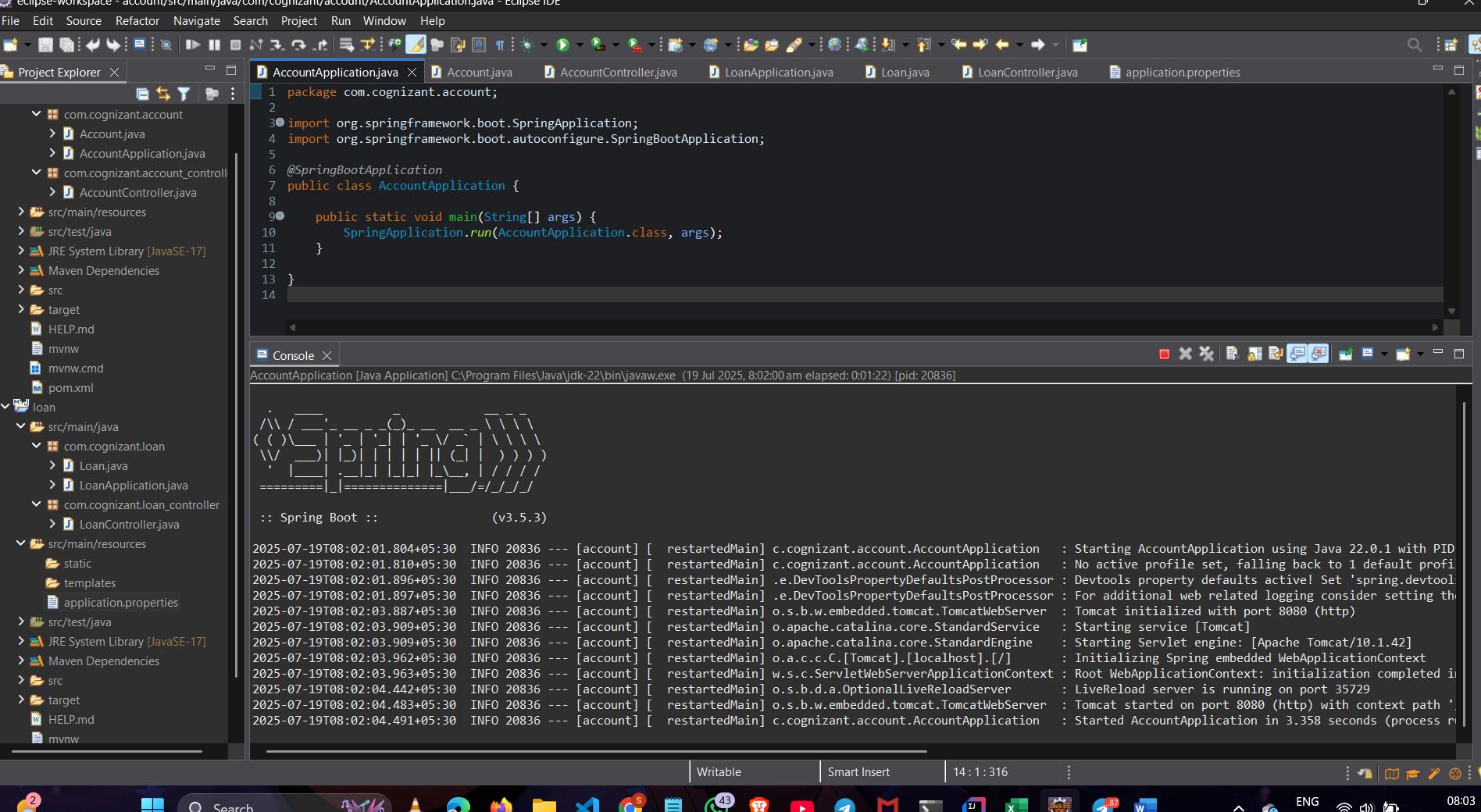
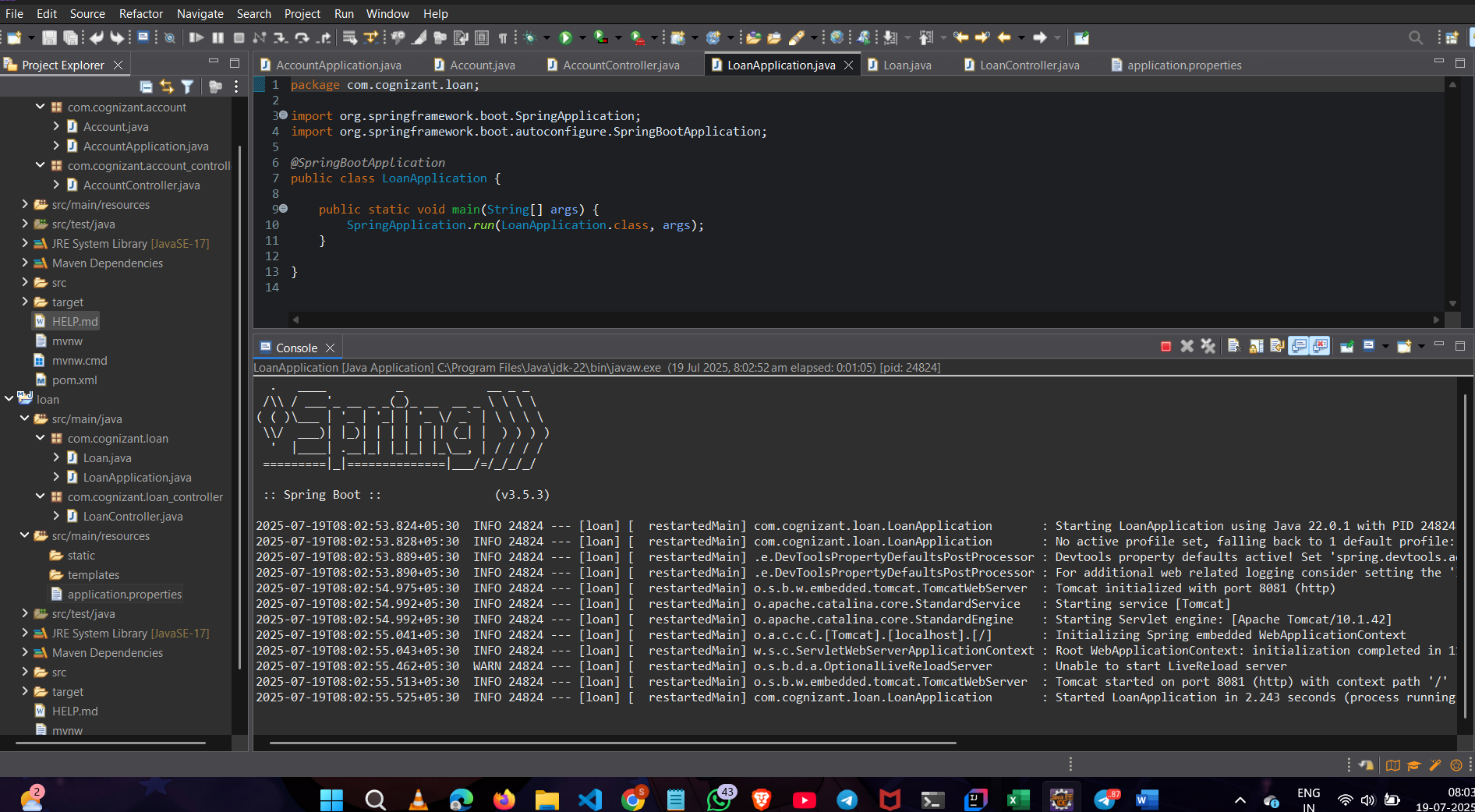
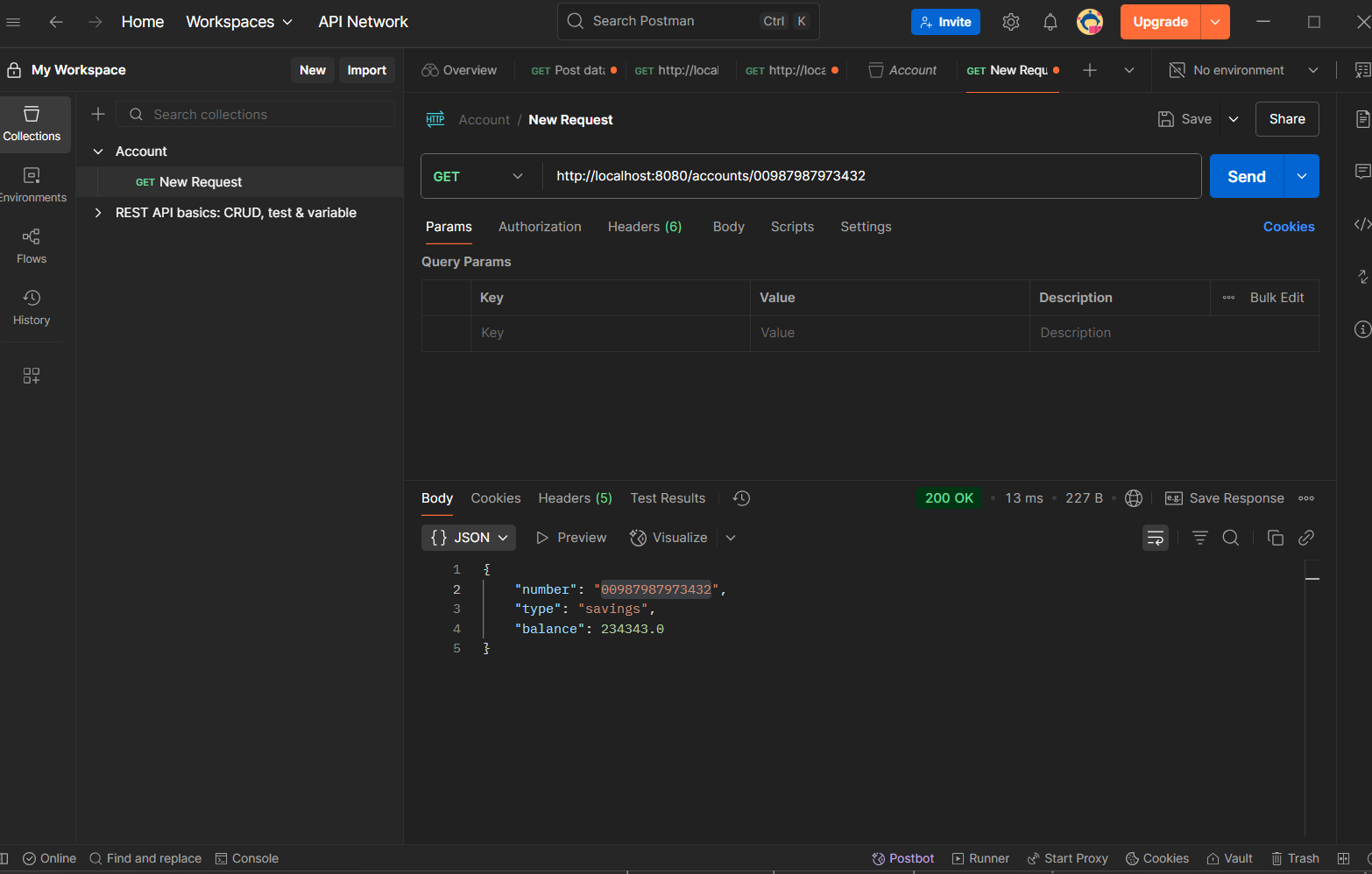
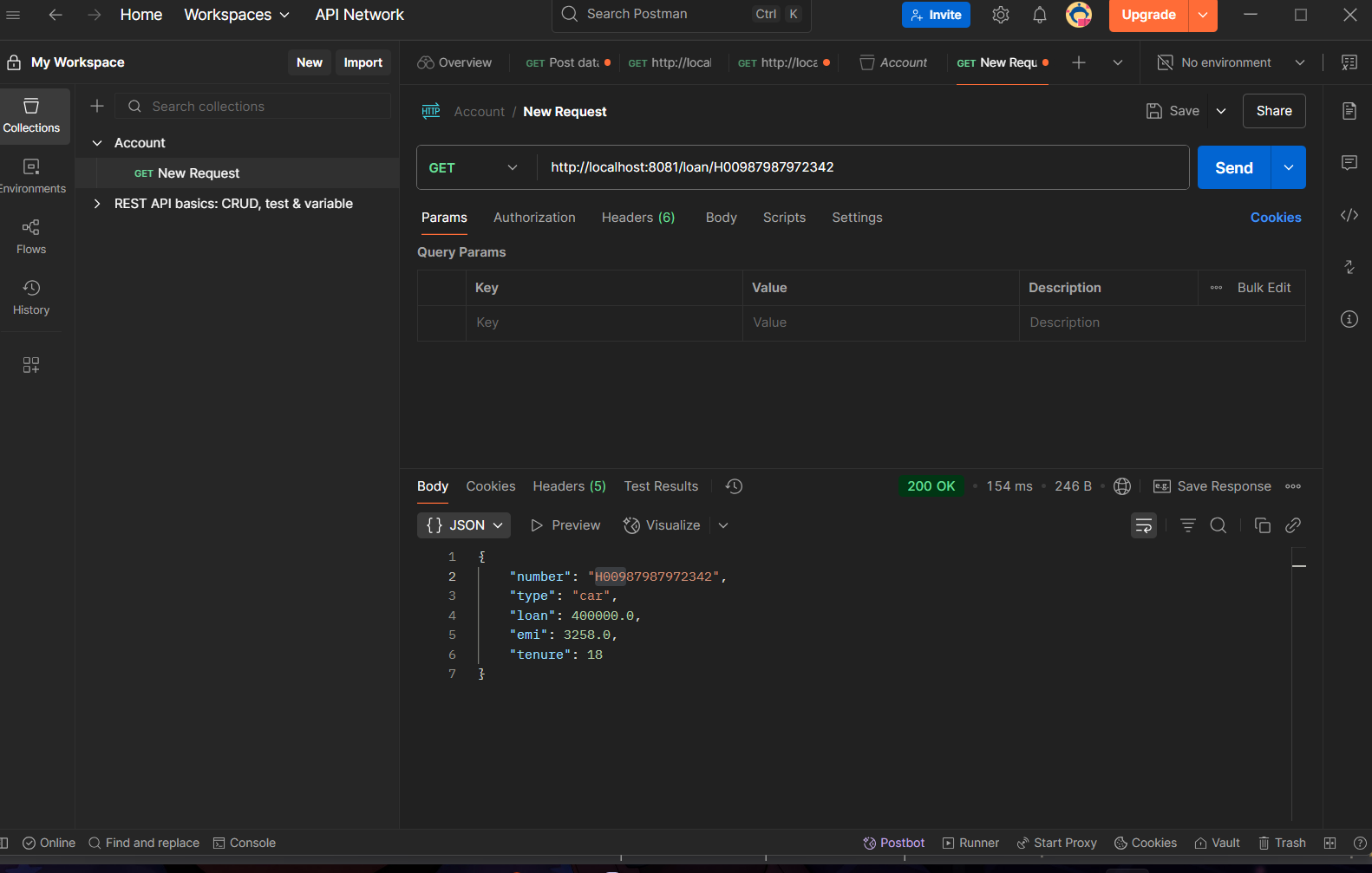
*@GetMapping*("/loan/{number}")

public Loan loanDetails(*@PathVariable* String number) {

return new Loan(number,"car", 400000, 3258, 18 );

}

}

OUTPUT:

Create Eureka Discovery Server and register microservices

File:EurekaDiscoveryServerApplication.java

package com.cognizant.eureka\_discovery\_server;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

import org.springframework.cloud.netflix.eureka.server.EnableEurekaServer;

*@SpringBootApplication*

*@EnableEurekaServer*

public class EurekaDiscoveryServerApplication {

public static void main(String[] args) {

SpringApplication.*run*(EurekaDiscoveryServerApplication.class, args);

}

}

File:application.properties(eureka server)

spring.application.name=eureka-discovery-server

server.port = 8761

eureka.client.fetch-registry = false

eureka.client.register-with-eureka = false

File:GreetServiceApplication.java

package com.cognizant.greet\_service;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

*@SpringBootApplication*

public class GreetServiceApplication {

public static void main(String[] args) {

SpringApplication.*run*(GreetServiceApplication.class, args);

}

}

File:application.properties(eureka client)

spring.application.name=eureka-discovery-server

server.port = 8761

eureka.client.fetch-registry = false

eureka.client.register-with-eureka = false

File:GreetController.java

package com.cognizant.greet\_service;

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

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

*@RestController*

public class GreetController {

*@GetMapping*("/greet")

public String greetService() {

return "Hello World";

OUTPUT:

