**WEEK – 5**

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

**2.Microservices with API gateway**

**Creating Microservices for account and loan**

**ACCOUNT MICROSEVICE**

**File Name : 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);

}

}

**File Name : 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 java.util.Map;

@RestController

public class AccountController {

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

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

return Map.of(

"number", number,

"type", "savings",

"balance", 234343

);

}

}

**File Name : application.properties**

server.port=8080

**File Name : pom.xml**

<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 http://maven.apache.org/xsd/maven-4.0.0.xsd">

<modelVersion>4.0.0</modelVersion>

<groupId>com.cognizant</groupId>

<artifactId>account</artifactId>

<version>0.0.1-SNAPSHOT</version>

<name>account</name>

<description>Account Microservice</description>

<packaging>jar</packaging>

<parent>

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

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

<version>3.2.5</version>

<relativePath/>

</parent>

<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>

</dependency>

</dependencies>

<build>

<plugins>

<plugin>

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

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

</plugin>

</plugins>

</build>

</project>

**LOAN MICROSERVICE**

**File Name : 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 Name : 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 java.util.Map;

@RestController

public class LoanController {

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

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

return Map.of(

"number", number,

"type", "car",

"loan", 400000,

"emi", 3258,

"tenure", 18

);

}

}

**File Name : application.properties**

server.port=8081

**File Name : pom.xml**

<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 http://maven.apache.org/xsd/maven-4.0.0.xsd">

<modelVersion>4.0.0</modelVersion>

<groupId>com.cognizant</groupId>

<artifactId>loan</artifactId>

<version>0.0.1-SNAPSHOT</version>

<name>loan</name>

<description>Loan Microservice</description>

<packaging>jar</packaging>

<parent>

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

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

<version>3.2.5</version>

<relativePath/>

</parent>

<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>

</dependency>

</dependencies>

<build>

<plugins>

<plugin>

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

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

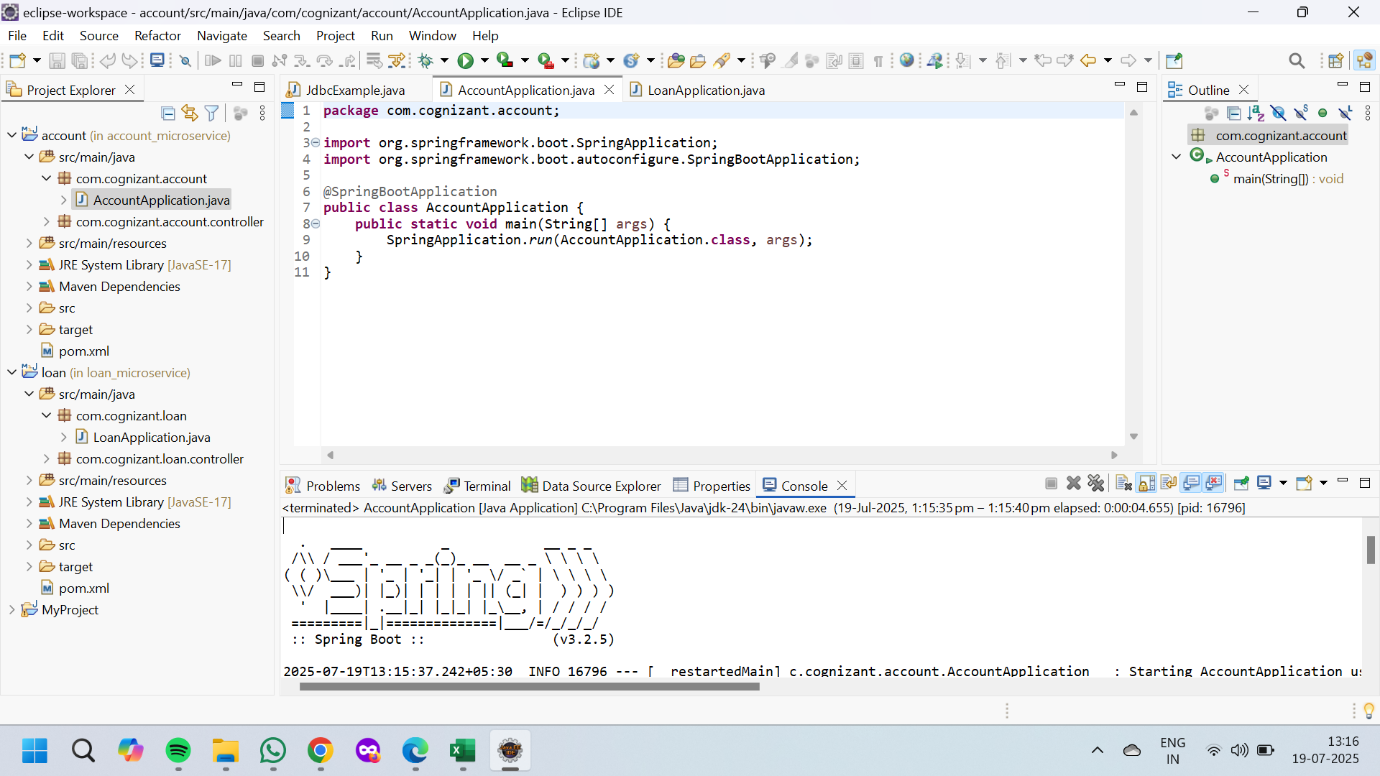
</plugin>

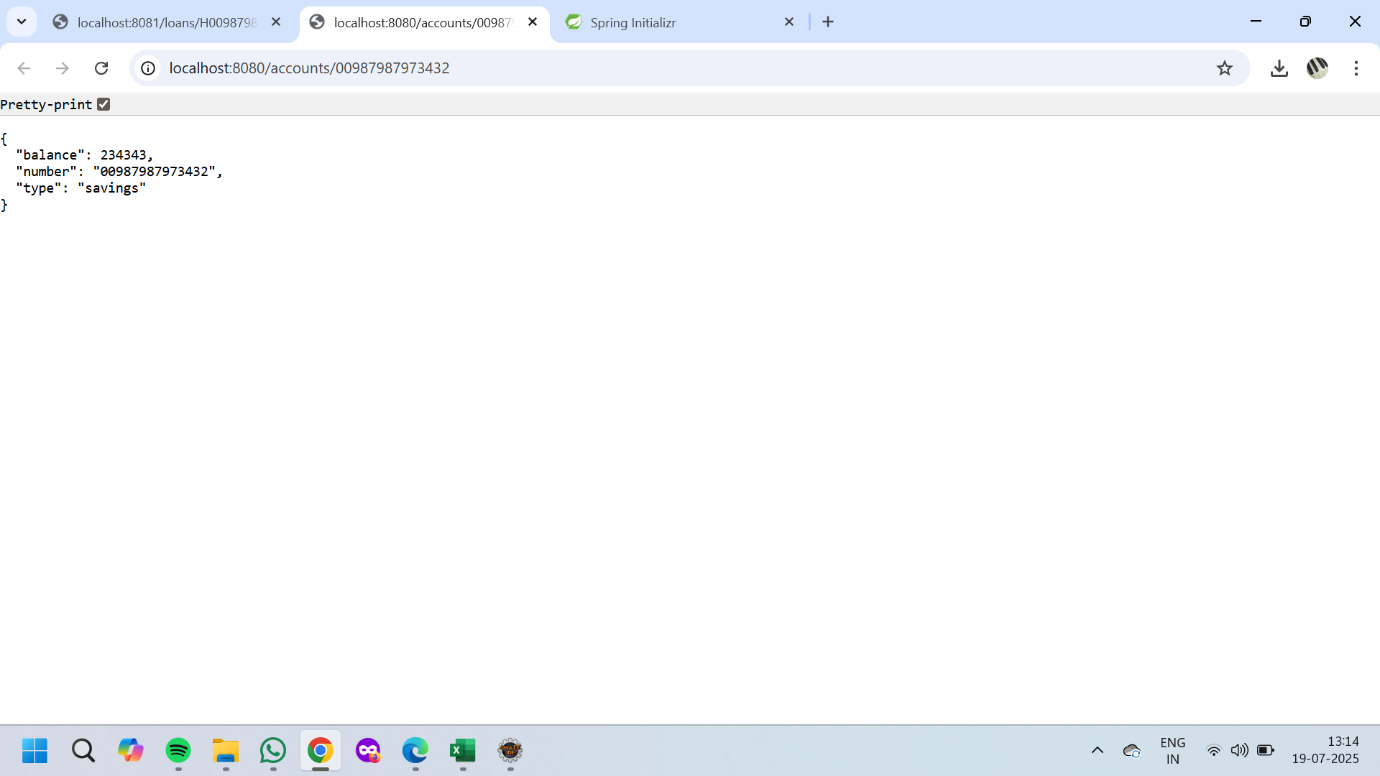
</plugins>

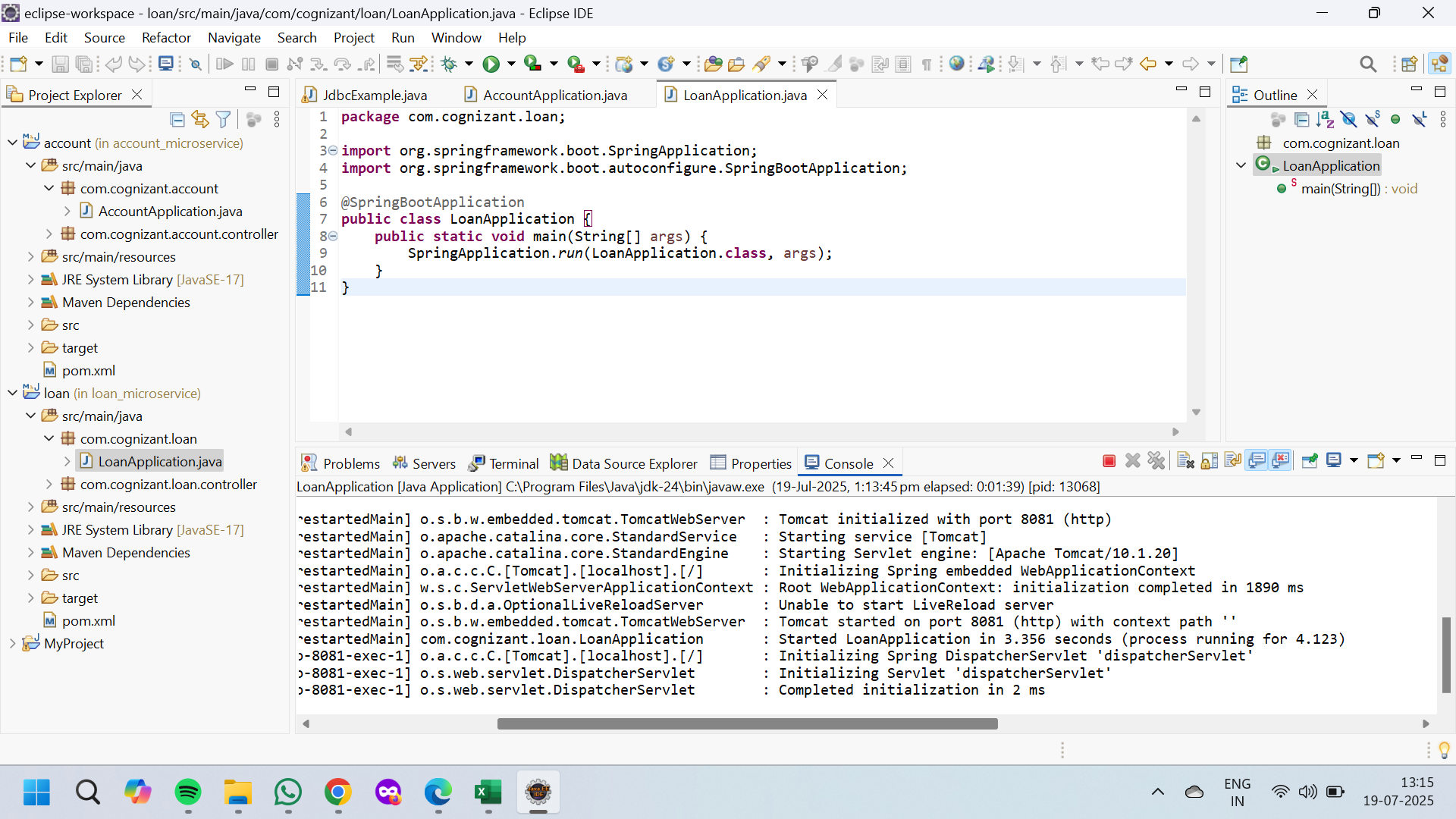
</build>

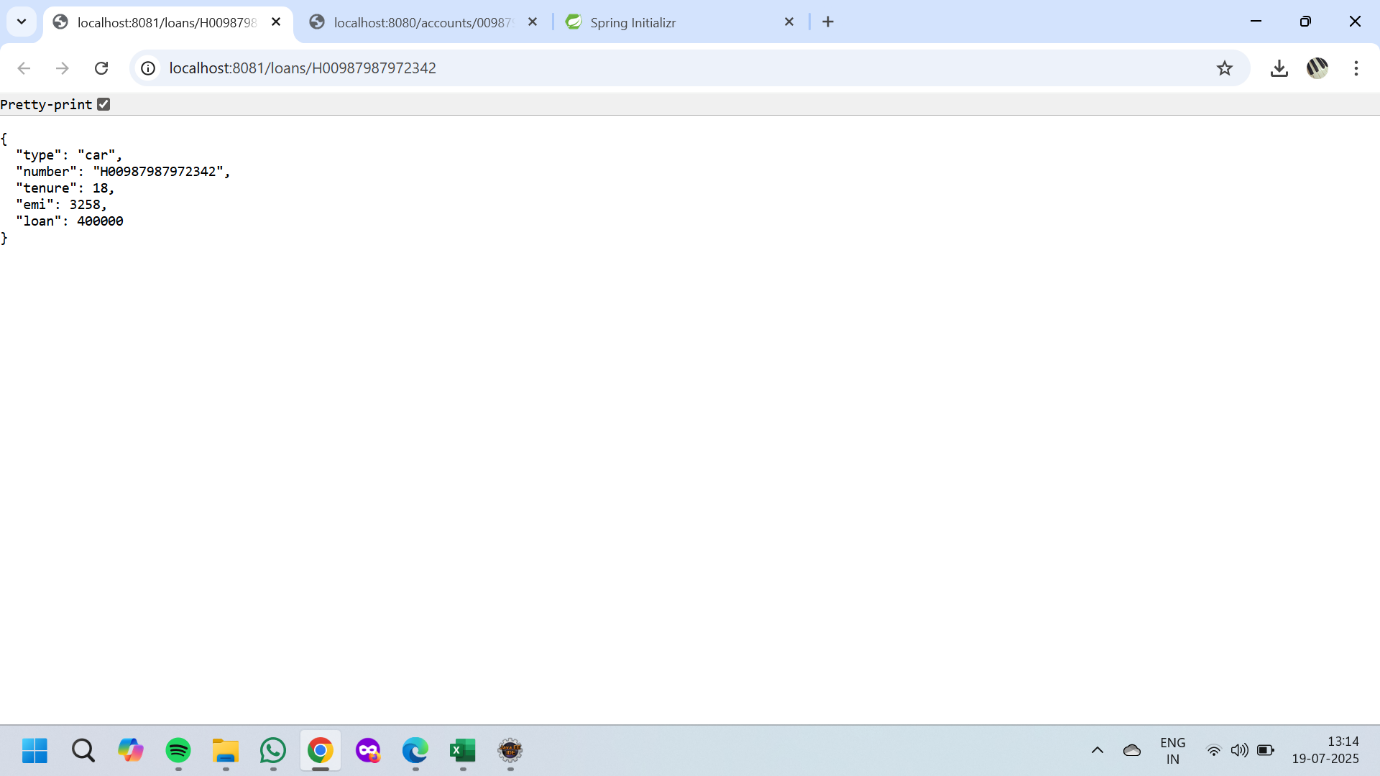
</project>

**OUTPUT:**

****

****

****

****

**2.Microservices with API gateway**

**Create Eureka Server and register microservices**

**eureka-discovery-server**

**File Name : application.properties**

server.port=8762

eureka.client.register-with-eureka=false

eureka.client.fetch-registry=false

**File Name : EurekaDiscoveryServerApplication.java**

package com.cognizant.eurekadiscoveryserver;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

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

@EnableEurekaServer

@SpringBootApplication

public class EurekaDiscoveryServerApplication {

public static void main(String[] args) {

SpringApplication.run(EurekaDiscoveryServerApplication.class, args);

}

}

**greet-service**

**File Name : application.properties**

server.port=8082

spring.application.name=greet-service

eureka.client.service-url.defaultZone=http://localhost:8762/eureka

**File Name : GreetServiceApplication.java**

package com.cognizant.greetservice;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

import org.springframework.cloud.client.discovery.EnableDiscoveryClient;

@EnableDiscoveryClient

@SpringBootApplication

public class GreetServiceApplication {

public static void main(String[] args) {

SpringApplication.run(GreetServiceApplication.class, args);

}

}

**File Name : controller.java**

package com.cognizant.greetservice;

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

@RestController

public class Controller {

@GetMapping("/greet")

public String greet() {

return "Hello World";

}

}

**account-service**

**File Name : application.properties**

server.port=8080

spring.application.name=account-service

eureka.client.service-url.defaultZone=http://localhost:8762/eureka

**File Name : AccountServiceApplication.java**

package com.cognizant.accountservice;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

import org.springframework.cloud.client.discovery.EnableDiscoveryClient;

@EnableDiscoveryClient

@SpringBootApplication

public class AccountServiceApplication {

public static void main(String[] args) {

SpringApplication.run(AccountServiceApplication.class, args);

}

}

**File Name : Controller.java**

package com.cognizant.accountservice;

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

import java.util.Map;

@RestController

@RequestMapping("/accounts")

public class Controller {

@GetMapping("/{number}")

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

return Map.of("number", number, "type", "savings", "balance", 234343);

}

}

**loan-service**

**File Name : application.properties**

server.port=8081

spring.application.name=loan-service

eureka.client.service-url.defaultZone=http://localhost:8762/eureka

**File Name : LoanServiceApplication.java**

package com.cognizant.loanservice;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

import org.springframework.cloud.client.discovery.EnableDiscoveryClient;

@EnableDiscoveryClient

@SpringBootApplication

public class LoanServiceApplication {

public static void main(String[] args) {

SpringApplication.run(LoanServiceApplication.class, args);

}

}

**File Name : Controller.java**

package com.cognizant.loanservice;

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

import java.util.Map;

@RestController

@RequestMapping("/loans")

public class Controller {

@GetMapping("/{number}")

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

return Map.of("number", number, "type", "car", "loan", 400000, "emi", 3258, "tenure", 18);

}

}

**api-gateway**

**File Name : application.properties**

server.port=9090

spring.application.name=api-gateway

eureka.client.service-url.defaultZone=http://localhost:8762/eureka

spring.cloud.gateway.discovery.locator.enabled=true

spring.cloud.gateway.discovery.locator.lower-case-service-id=true

**File Name : ApiGatewayApplication.java**

package com.cognizant.apigateway;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

import org.springframework.cloud.client.discovery.EnableDiscoveryClient;

import org.springframework.cloud.gateway.route.RouteLocator;

import org.springframework.cloud.gateway.route.builder.RouteLocatorBuilder;

import org.springframework.context.annotation.Bean;

@EnableDiscoveryClient

@SpringBootApplication

public class ApiGatewayApplication {

public static void main(String[] args) {

SpringApplication.run(ApiGatewayApplication.class, args);

}

@Bean

public RouteLocator customRouteLocator(RouteLocatorBuilder builder) {

return builder.routes()

.route("greet-service", r -> r.path("/greet-service/\*\*").uri("lb://greet-service"))

.route("account-service", r -> r.path("/account-service/\*\*").uri("lb://account-service"))

.route("loan-service", r -> r.path("/loan-service/\*\*").uri("lb://loan-service"))

.build();

}

}

**File Name : LogFilter.java**

package com.cognizant.apigateway.filter;

import org.springframework.cloud.gateway.filter.GlobalFilter;

import org.springframework.core.annotation.Order;

import org.springframework.stereotype.Component;

import org.springframework.web.server.ServerWebExchange;

import reactor.core.publisher.Mono;

import org.springframework.cloud.gateway.filter.GatewayFilterChain;

@Component

@Order(1)

public class LogFilter implements GlobalFilter {

@Override

public Mono<Void> filter(ServerWebExchange exchange, GatewayFilterChain chain) {

System.out.println("Request received for: " + exchange.getRequest().getURI());

return chain.filter(exchange);

}

}

**File Name : pom.xml**

**eureka-discovery-server**

<dependency>

<groupId>org.springframework.cloud</groupId>

<artifactId>spring-cloud-starter-netflix-eureka-server</artifactId>

</dependency>

**greet-service/pom.xml, account-service, loan-service**

<dependency>

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

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

</dependency>

<dependency>

<groupId>org.springframework.cloud</groupId>

<artifactId>spring-cloud-starter-netflix-eureka-client</artifactId>

</dependency>

**api-gateway/pom.xml**

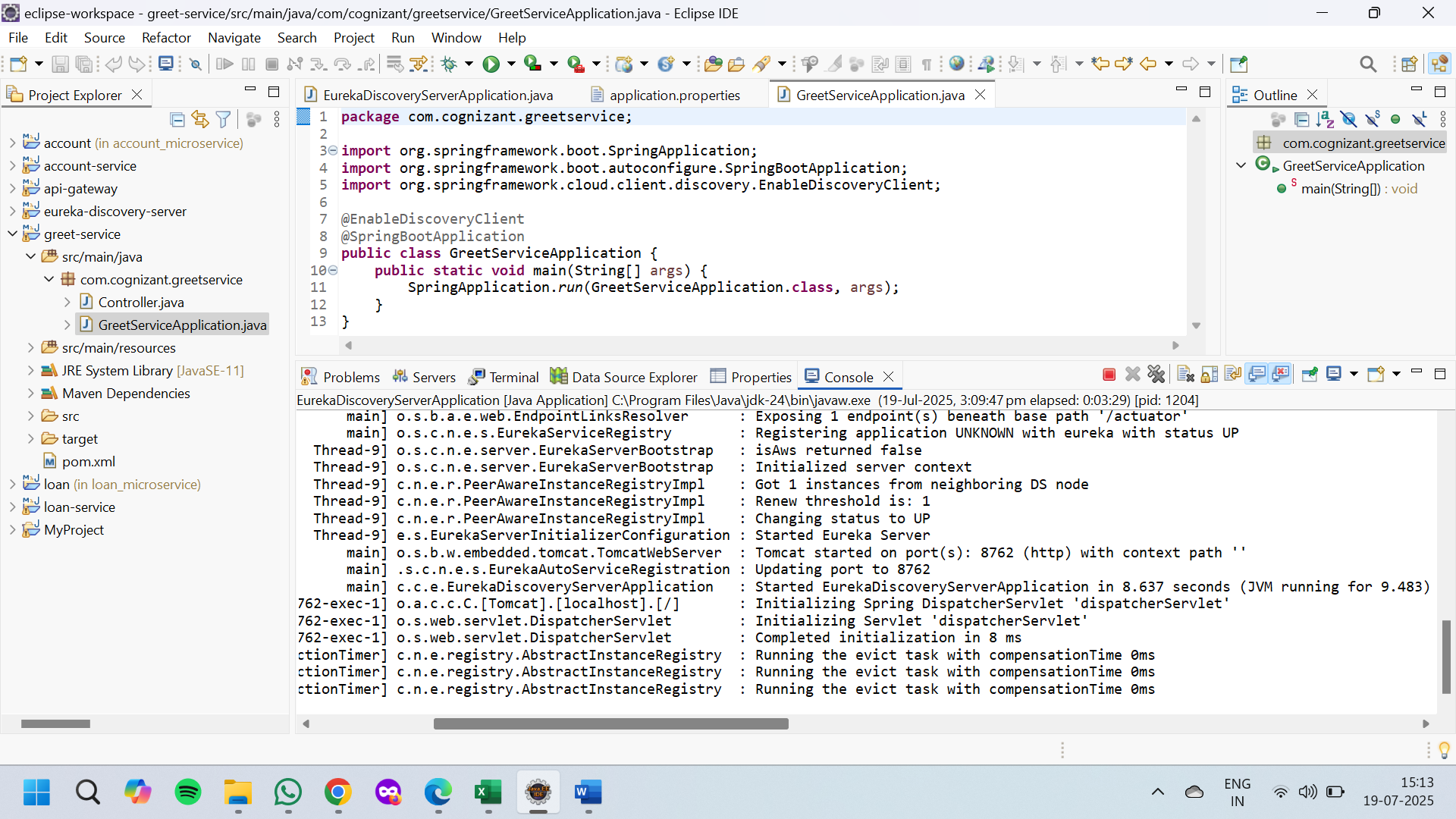
<dependency>

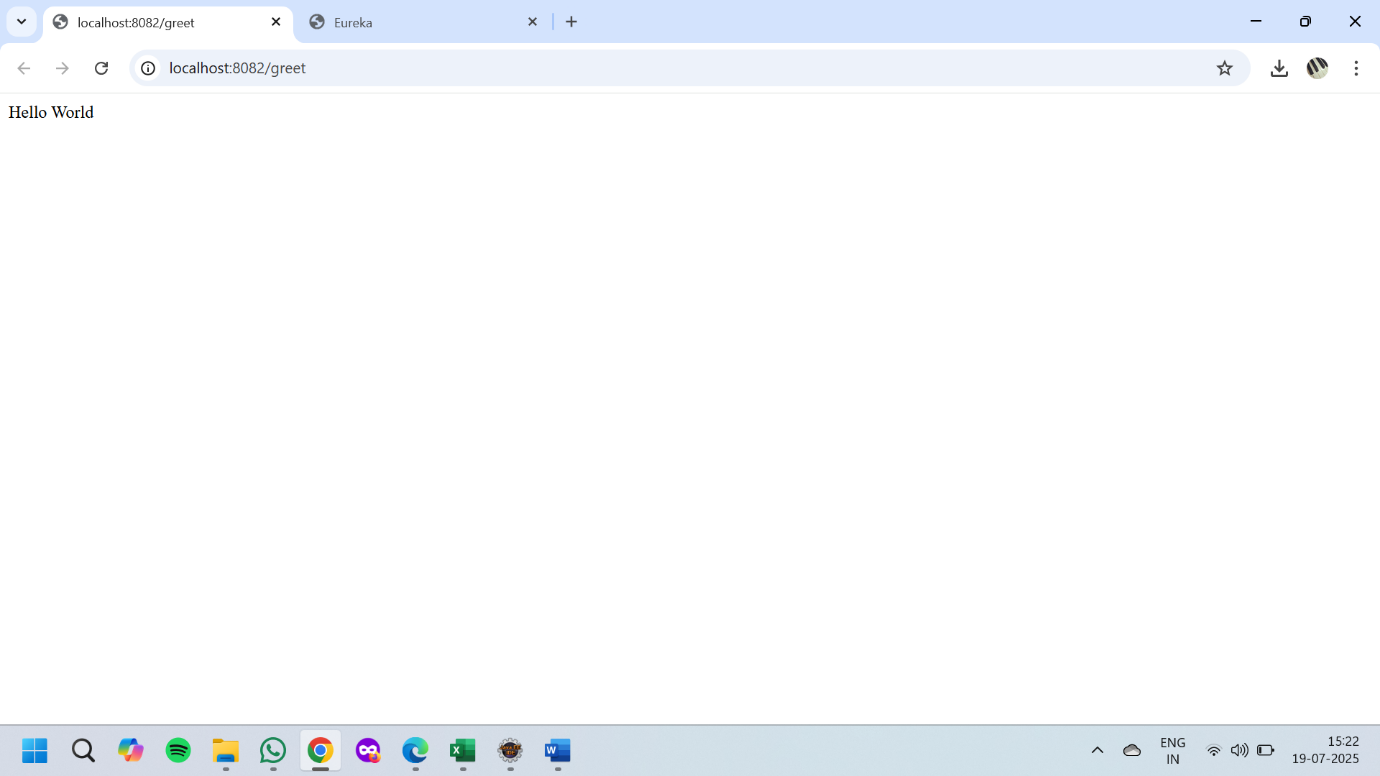
<groupId>org.springframework.cloud</groupId>

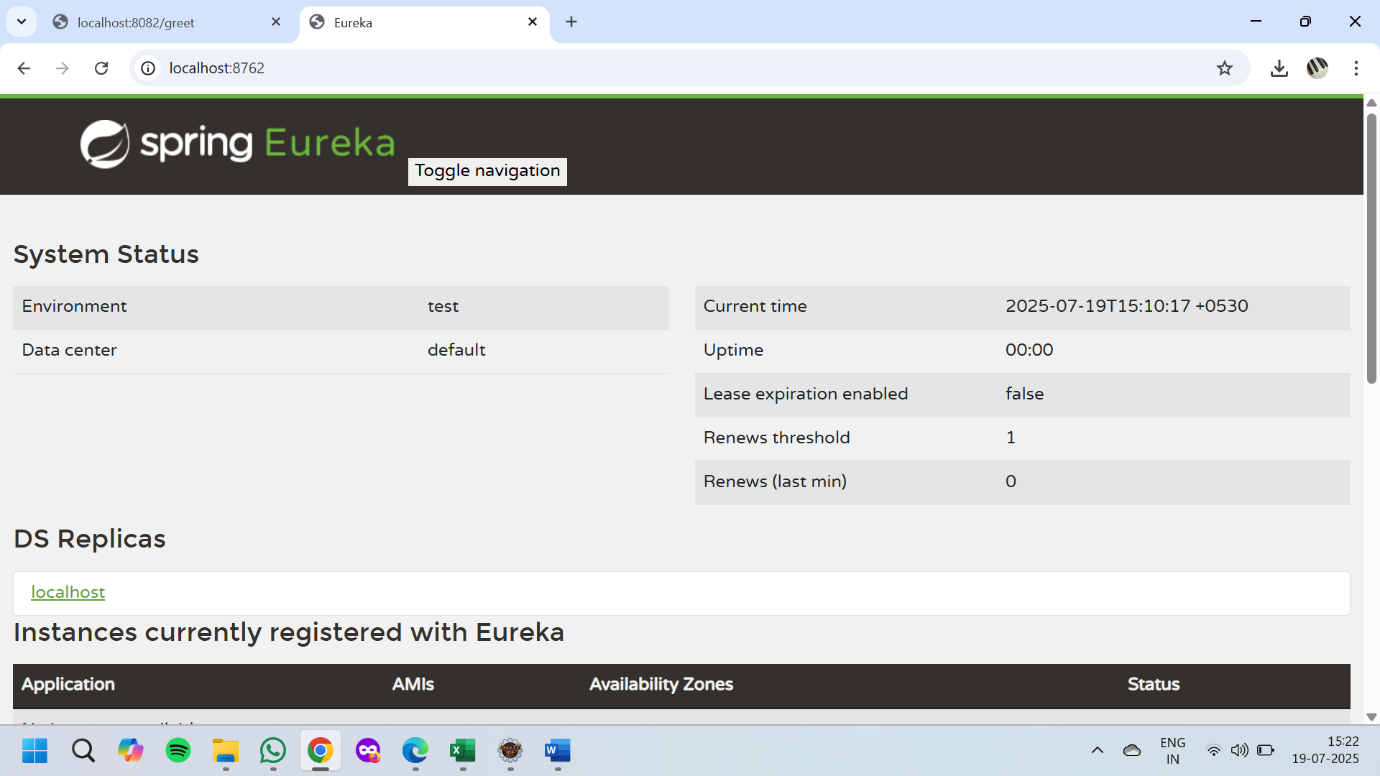
<artifactId>spring-cloud-starter-gateway</artifactId>

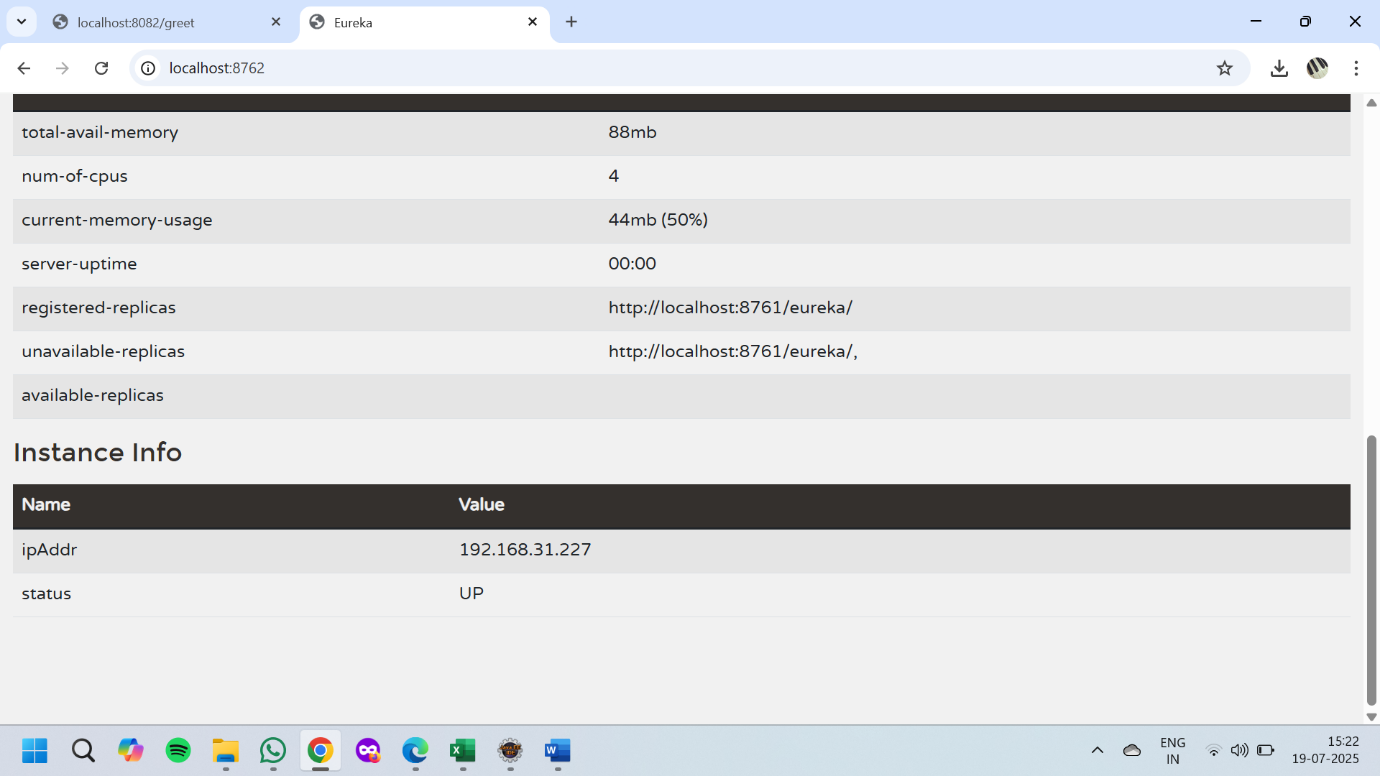
</dependency>

**OUTPUT:**

****

****

****

****