**WEEK 5**

**Microservices with API gateway**

**Creating Microservices for account and loan**

1.Created controller file:

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

);

}

}

**Loan Microservice**

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

);

}

}

Account Microservice Output:

{

"number": "00987987973432",

"type": "savings",

"balance": 234343

}

Loan Microservice Output:

{

"number": "H00987987972342",

"type": "car",

"loan": 400000,

"emi": 3258,

"tenure": 18

}

**Microservices with API gateway**

Create Eureka Discovery Server and register

**Eureka Discovery Server and Microservices Registration**

1. Created Eureka Discovery Server

application.properties:

server.port=8761

eureka.client.register-with-eureka=false

eureka.client.fetch-registry=false

logging.level.com.netflix.eureka=OFF

logging.level.com.netflix.discovery=OFF

* Ran the application.
* Accessed the Eureka dashboard at:  
  http://localhost:8761  
  Initially, no services were registered.

**2. Registered Account Microservice**

application.properties:

server.port=8080

spring.application.name=account-service

eureka.client.service-url.defaultZone=http://localhost:8761/eureka

* Restarted all services:
  + First started eureka-discovery-server
  + Then started account-service
* Verified registration at:  
  http://localhost:8761  
  → **account-service** was listed under registered instances.

**3. Registered Loan Microservice**

* application.properties:

server.port=8081

spring.application.name=loan-service

eureka.client.service-url.defaultZone=http://localhost:8761/eureka

**Spring Cloud API Gateway Setup**

**4. Created Greet Microservice**

* application.properties:

server.port=8082

spring.application.name=greet-service

eureka.client.service-url.defaultZone=http://localhost:8761/eureka

* Created GreetController.java:

package com.cognizant.greet.controller;

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

@RestController

@RequestMapping("/greet")

public class GreetController {

@GetMapping

public String greet() {

return "Hello World";

}

}

Output: "Hello World"

**5. Created Eureka Discovery Server (Again for Greet-Service)**

* Same setup as step 1.
* Verified Eureka running at:
* http://localhost:8761

**6. Configured Greet-Service POM**

Pom.xml

<dependencyManagement>

<dependencies>

<dependency>

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

<artifactId>spring-cloud-dependencies</artifactId>

<version>2023.0.0</version>

<type>pom</type>

<scope>import</scope>

</dependency>

</dependencies>

</dependencyManagement>

**7. Created API Gateway**

* Created new Spring Boot project:
  + **Group**: com.cognizant
  + **Artifact**: api-gateway
* Added dependencies:
  + Spring Cloud Gateway
  + Eureka Discovery Client
* Added to application.properties:

server.port=9090

spring.application.name=api-gateway

eureka.client.service-url.defaultZone=http://localhost:8761/eureka

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

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

**8. Accessed Greet-Service Through API Gateway**

* Accessed:
* http://localhost:9090/greet-service/greet
* Output:  
  "Hello World"

**9. Created Global Logging Filter in API Gateway**

* Created a new class LogFilter.java in api-gateway:

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 java.util.logging.Logger;

@Component

@Order(-1)

public class LogFilter implements GlobalFilter {

private final Logger logger = Logger.getLogger(LogFilter.class.getName());

@Override

public Mono<Void> filter(ServerWebExchange exchange, org.springframework.cloud.gateway.filter.GatewayFilterChain chain) {

logger.info("Request received: " + exchange.getRequest().getURI());

return chain.filter(exchange);

}

}

* Accessed:
* http://localhost:9090/greet-service/greet
* Verified the **request was logged** in API Gateway console.