

Week 5 : Microservices

Microservices with Spring Boot 3 and Spring Cloud

Microservices with API gateway:

Creating Microservices for account and loan:

Create REST Controller

AccountController.java:

```
package com.cognizant.account;
```

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

```
@RestController
```

```
@RequestMapping("/accounts")
```

```
public class AccountController {
```

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

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

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

```
    }
```

```
    static class Account {
```

```
        public String number;
```

```

    public String type;
    public double balance;

    public Account(String number, String type, double
balance) {
        this.number = number;
        this.type = type;
        this.balance = balance;
    }
}

```

Its shown:

```

{
    "number": "00987987973432",
    "type": "savings",
    "balance": 234343
}

```

LoanController.java:

```

package com.cognizant.loan;

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

@RestController

```

```

@RequestMapping("/loans")
public class LoanController {

    @GetMapping("/{number}")
    public Loan getLoan(@PathVariable String number) {
        return new Loan(number, "car", 400000, 3258, 18);
    }

    static class Loan {
        public String number;
        public String type;
        public int loan;
        public int emi;
        public int tenure;

        public Loan(String number, String type, int loan, int emi, int tenure) {
            this.number = number;
            this.type = type;
            this.loan = loan;
            this.emi = emi;
            this.tenure = tenure;
        }
    }
}

Its Shown
{
    "number": "H00987987972342",

```

```

"type": "car",

"loan": 400000,

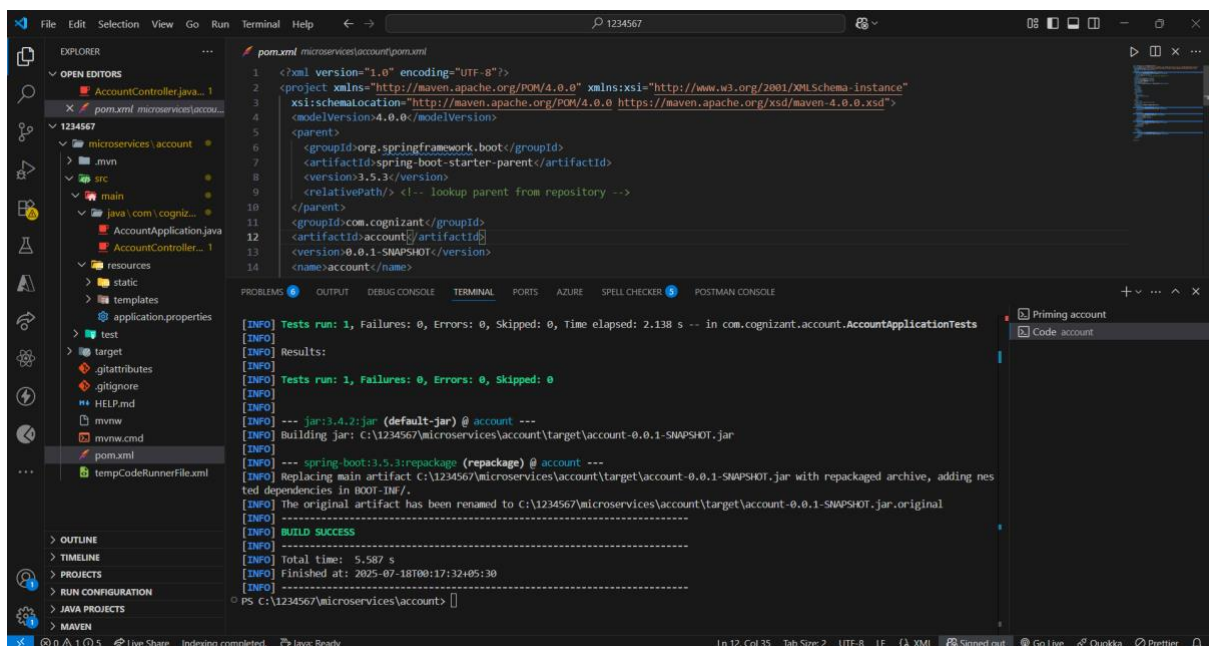
"emi": 3258,

"tenure": 18

}

```

Output:



Create and Launch Eureka Discovery Server

EurekaDiscoveryServerApplication.java

@SpringBootApplication

@EnableEurekaServer

```
public class EurekaDiscoveryServerApplication {
```

```
    public static void main(String[] args) {
```

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

```
}  
}
```

application.properties

server.port=8081

spring.application.name=loan-service

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

server.port=9090

spring.application.name=api-gateway

Eureka config

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

Route lowercase

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

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

LogFilter.java:

```
package com.cognizant.apigateway.filters;
```

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

```
import org.springframework.core.annotation.Order;
```

```
import org.springframework.stereotype.Component;
```

```
import reactor.core.publisher.Mono;
```

```
import org.springframework.web.server.ServerWebExchange;
```

```
import
org.springframework.cloud.gateway.filter.GatewayFilterChain;
import java.util.logging.Logger;
```

```
@Component
```

```
@Order(1)
```

```
public class LogFilter implements GlobalFilter {
```

```
    Logger logger = Logger.getLogger(LogFilter.class.getName());
```

```
    @Override
```

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

```
        logger.info("API Gateway - Request path: " +
exchange.getRequest().getPath());
```

```
        return chain.filter(exchange);
```

```
    }
```

```
}
```

pom.xml:

```
<dependencyManagement>
```

```
    <dependencies>
```

```
        <dependency>
```

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

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

<version>${spring-cloud.version}</version>

<type>pom</type>

<scope>import</scope>

</dependency>

</dependencies>

</dependencyManagement>
```

Output:

```
[INFO]
[INFO] Results:
[INFO]
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0
[INFO]
[INFO]
[INFO] --- jar:3.4.2:jar (default-jar) @ account ---
[INFO] Building jar: C:\1234567\microservices\account\target\account-0.0.1-SNAPSHOT.jar
[INFO]
[INFO] --- spring-boot:3.5.3:repackage (repackage) @ account ---
[INFO] Replacing main artifact C:\1234567\microservices\account\target\account-0.0.1-SNAPSHOT.jar with repackaged archive, adding nested dependencies in BOOT-INF/.
[INFO] The original artifact has been renamed to C:\1234567\microservices\account\target\account-0.0.1-SNAPSHOT.jar.original
[INFO]
[INFO] BUILD SUCCESS
[INFO]
[INFO] Total time: 5.587 s
[INFO] Finished at: 2025-07-18T00:17:32+05:30
[INFO]
```