##`account-service`

server:

port: 8081

spring:

application:

name: account-service

eureka:

client:

service-url:

defaultZone: http://localhost:8761/eureka/

```

### `AccountServiceApplication.java`

```java

@SpringBootApplication

@EnableDiscoveryClient

public class AccountServiceApplication {

public static void main(String[] args) {

SpringApplication.run(AccountServiceApplication.class, args);

}

}

```

### `AccountController.java`

@RestController

@RequestMapping("/api/account")

public class AccountController {

@GetMapping("/status")

public String getStatus() {

return "Account Service is up";

}

}

```

---

## `loan-service`

server:

port: 8082

spring:

application:

name: loan-service

eureka:

client:

service-url:

defaultZone: http://localhost:8761/eureka/

```

### `LoanServiceApplication.java`

@SpringBootApplication

@EnableDiscoveryClient

public class LoanServiceApplication {

public static void main(String[] args) {

SpringApplication.run(LoanServiceApplication.class, args);

}

}

```

### `LoanController.java`

@RestController

@RequestMapping("/api/loan")

public class LoanController {

@GetMapping("/status")

public String getStatus() {

return "Loan Service is up";

}

}

```

---

## `eureka-server`

server:

port: 8761

spring:

application:

name: eureka-server

eureka:

client:

register-with-eureka: false

fetch-registry: false

```

### `EurekaServerApplication.java`

```java

@SpringBootApplication

@EnableEurekaServer

public class EurekaServerApplication {

public static void main(String[] args) {

SpringApplication.run(EurekaServerApplication.class, args);

}

}

```

---

## Output

### From Browser or Postman:

#### `http://localhost:8081/api/account/status`

```

Account Service is up

```

#### `http://localhost:8082/api/loan/status`

```

Loan Service is up

```

#### `http://localhost:8761/`

```

Eureka Dashboard showing:

- account-service

- loan-service

Registered and available

```