

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);

    }

}
```

AccountController.java:

```
package com.cognizant.account;

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> getAccountDetails(@PathVariable String number) {

        return Map.of(

            "number", number,

            "type", "savings",

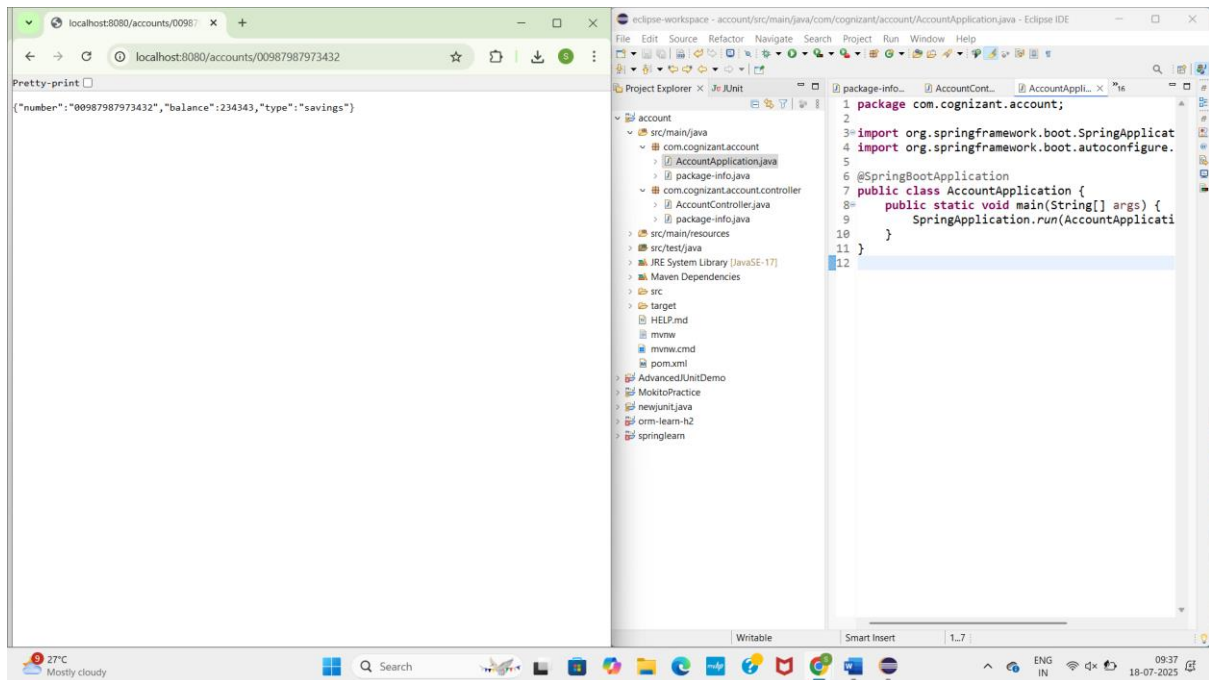
            "balance", 234343

        );

    }

}
```

OUTPUT:



LoanController.java:

```
package com.cognizant.loan;

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;
import java.util.HashMap;

@RestController

public class LoanController {

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

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

        Map<String, Object> loanDetails = new HashMap<>();

        loanDetails.put("number", number);

        loanDetails.put("type", "car");

        loanDetails.put("loan", 400000);

        loanDetails.put("emi", 3258);

        loanDetails.put("tenure", 18);

        return loanDetails;

    }

}
```

```
}
```

LoanApplication.java:

```
package com.cognizant.loan;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication

public class LoanApplication {

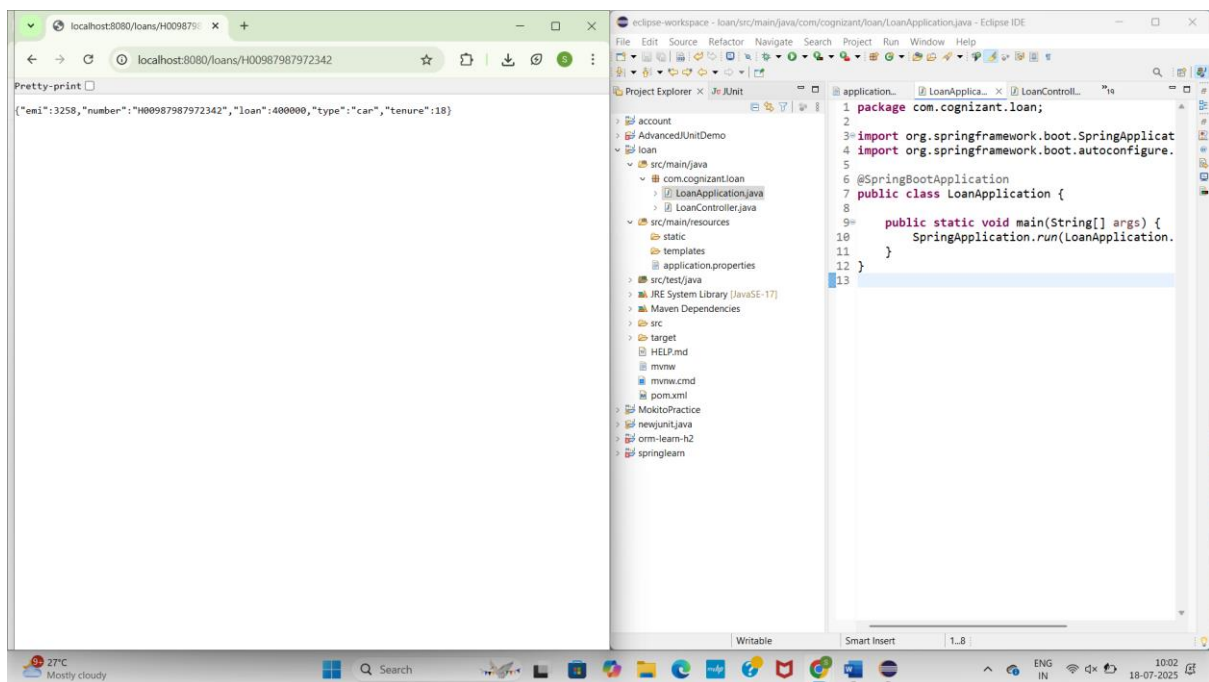
    public static void main(String[] args) {

        SpringApplication.run(LoanApplication.class, args);

    }

}
```

OUTPUT:



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);

    }

}

```

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>eureka-discovery-server</artifactId>
    <version>0.0.1-SNAPSHOT</version>
    <packaging>jar</packaging>
    <name>eureka-discovery-server</name>
    <description>Eureka Discovery Server</description>
    <parent>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-starter-parent</artifactId>
        <version>3.1.5</version>
        <relativePath/> <!-- lookup parent from repository -->
    </parent>
    <properties>
        <java.version>17</java.version>
        <spring-cloud.version>2022.0.4</spring-cloud.version>
    </properties>
    <dependencies>
        <dependency>
            <groupId>org.springframework.cloud</groupId>
            <artifactId>spring-cloud-starter-netflix-eureka-server</artifactId>

```

```
</dependency>
<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>
  <optional>true</optional>
</dependency>
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-test</artifactId>
  <scope>test</scope>
</dependency>
</dependencies>
<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>
<build>
  <plugins>
    <plugin>
```

```

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

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

</plugin>

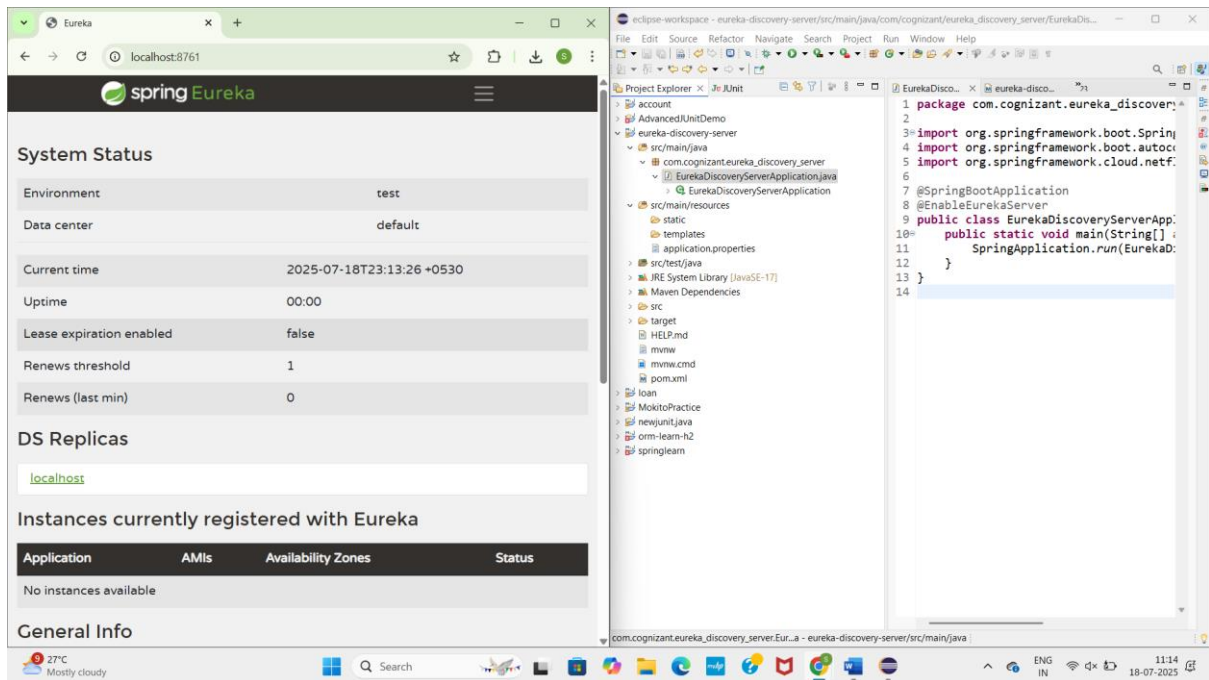
</plugins>

</build>

</project>

```

OUTPUT:



loan/pom.xml:

```

<?xml version="1.0" encoding="UTF-8"?>

<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
    https://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>3.1.5</version>

  <relativePath/> <!-- lookup parent from repository -->

</parent>

<properties>

  <java.version>17</java.version>

  <spring-cloud.version>2022.0.4</spring-cloud.version>

</properties>

<dependencies>

  <!-- Spring Boot Web -->

  <dependency>

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

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

  </dependency>

  <!-- Eureka Client -->

  <dependency>

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

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

  </dependency>

  <!-- DevTools (for auto-restart) -->

  <dependency>

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

    <artifactId>spring-boot-devtools</artifactId>

    <scope>runtime</scope>

    <optional>true</optional>

  </dependency>

  <!-- Test -->

  <dependency>
```

```

        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-starter-test</artifactId>
        <scope>test</scope>
    </dependency>
</dependencies>
<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>
<build>
    <plugins>
        <!-- Spring Boot Maven Plugin -->
        <plugin>
            <groupId>org.springframework.boot</groupId>
            <artifactId>spring-boot-maven-plugin</artifactId>
        </plugin>
    </plugins>
</build>
</project>

```

LoanApplication.java:

```

package com.cognizant.loan;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;
import org.springframework.cloud.client.discovery.EnableDiscoveryClient;

```


@SpringBootApplication

@EnableDiscoveryClient

```
public class LoanApplication {  
  
    public static void main(String[] args) {  
  
        SpringApplication.run(LoanApplication.class, args);  
  
    }  
}
```

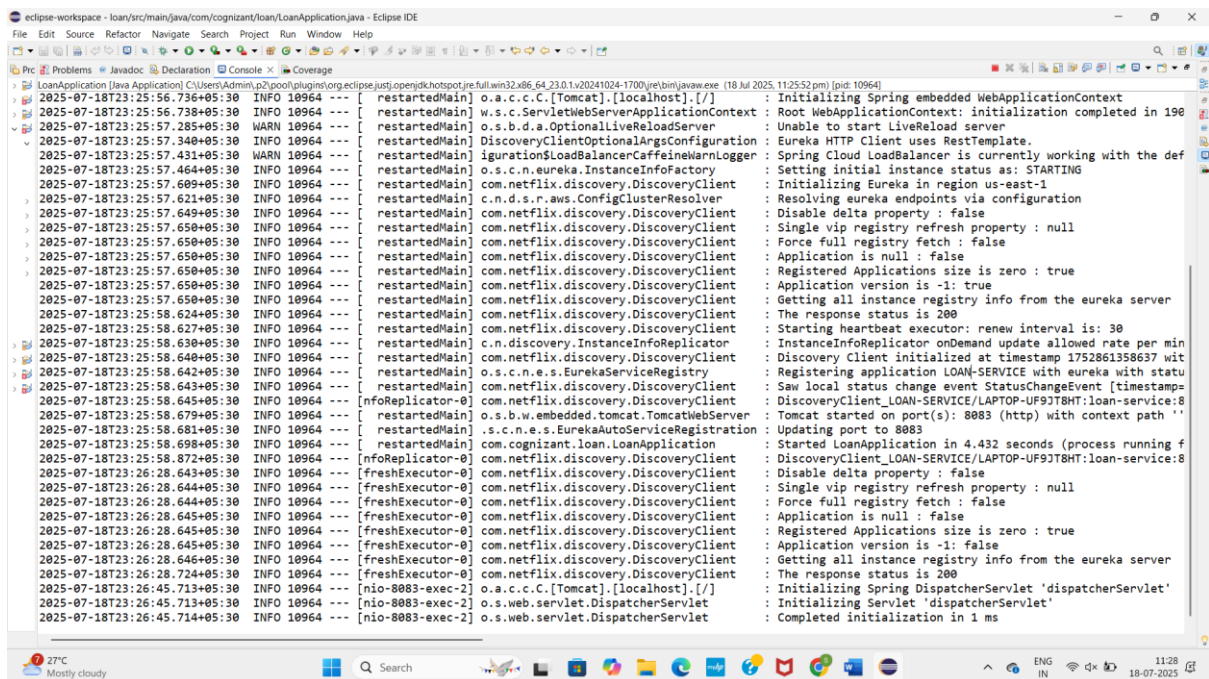
application.properties of loan service:

spring.application.name=loan-service

server.port=8083

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

OUTPUT:



```
2025-07-18T23:25:56.736+05:30 INFO 10964 --- [ restartedMain] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring embedded WebApplicationContext  
2025-07-18T23:25:56.738+05:30 INFO 10964 --- [ restartedMain] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization completed in 190  
2025-07-18T23:25:57.285+05:30 WARN 10964 --- [ restartedMain] o.s.b.d.a.OptionalLiveReloadServer : Unable to start LiveReload server  
2025-07-18T23:25:57.348+05:30 INFO 10964 --- [ restartedMain] DiscoveryClientOptionalArgsConfiguration : Eureka HTTP Client uses RestTemplate.  
2025-07-18T23:25:57.431+05:30 WARN 10964 --- [ restartedMain] igation$LoadBalancerCaffeineWarnLogger : Spring Cloud LoadBalancer is currently working with the def  
2025-07-18T23:25:57.464+05:30 INFO 10964 --- [ restartedMain] o.s.c.n.eureka.InstanceInfoFactory : Setting initial instance status as: STARTING  
2025-07-18T23:25:57.609+05:30 INFO 10964 --- [ restartedMain] com.netflix.discovery.DiscoveryClient : Initializing Eureka in region us-east-1  
2025-07-18T23:25:57.621+05:30 INFO 10964 --- [ restartedMain] c.n.d.s.r.aws.ConfigClusterResolver : Resolving eureka endpoints via configuration  
2025-07-18T23:25:57.649+05:30 INFO 10964 --- [ restartedMain] com.netflix.discovery.DiscoveryClient : Disable delta property : false  
2025-07-18T23:25:57.658+05:30 INFO 10964 --- [ restartedMain] com.netflix.discovery.DiscoveryClient : Single vip registry refresh property : null  
2025-07-18T23:25:57.658+05:30 INFO 10964 --- [ restartedMain] com.netflix.discovery.DiscoveryClient : Force full registry fetch : false  
2025-07-18T23:25:57.658+05:30 INFO 10964 --- [ restartedMain] com.netflix.discovery.DiscoveryClient : Application is null : false  
2025-07-18T23:25:57.658+05:30 INFO 10964 --- [ restartedMain] com.netflix.discovery.DiscoveryClient : Registered Applications size is zero : true  
2025-07-18T23:25:57.658+05:30 INFO 10964 --- [ restartedMain] com.netflix.discovery.DiscoveryClient : Application version is -1: true  
2025-07-18T23:25:57.658+05:30 INFO 10964 --- [ restartedMain] com.netflix.discovery.DiscoveryClient : Getting all instance registry info from the eureka server  
2025-07-18T23:25:58.624+05:30 INFO 10964 --- [ restartedMain] com.netflix.discovery.DiscoveryClient : The response status is 200  
2025-07-18T23:25:58.627+05:30 INFO 10964 --- [ restartedMain] com.netflix.discovery.DiscoveryClient : Starting heartbeat executor: renew interval is: 30  
2025-07-18T23:25:58.630+05:30 INFO 10964 --- [ restartedMain] c.n.discovery.InstanceInfoReplicator : InstanceInfoReplicator onDemand update allowed rate per min  
2025-07-18T23:25:58.648+05:30 INFO 10964 --- [ restartedMain] com.netflix.discovery.DiscoveryClient : Discovery Client initialized at timestamp 1752861358637 wit  
2025-07-18T23:25:58.642+05:30 INFO 10964 --- [ restartedMain] o.s.c.n.e.s.EurekaServiceRegistry : Registering application LOAN-SERVICE with eureka with statu  
2025-07-18T23:25:58.643+05:30 INFO 10964 --- [ restartedMain] com.netflix.discovery.DiscoveryClient : Saw local status change event StatusChangeEvent [timestamp=  
2025-07-18T23:25:58.645+05:30 INFO 10964 --- [ nfoReplicator-0] com.netflix.discovery.DiscoveryClient : DiscoveryClient_LOAN-SERVICE/LAPTOP-UF9J78HT:loan-service:8  
2025-07-18T23:25:58.679+05:30 INFO 10964 --- [ restartedMain] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port(s): 8083 (http) with context path ''  
2025-07-18T23:25:58.681+05:30 INFO 10964 --- [ restartedMain] o.s.c.n.e.s.EurekaAutoServiceRegistration : Updating port to 8083  
2025-07-18T23:25:58.698+05:30 INFO 10964 --- [ restartedMain] com.cognizant.loan.LoanApplication : Started LoanApplication in 4.432 seconds (process running f  
2025-07-18T23:25:58.872+05:30 INFO 10964 --- [nfoReplicator-0] com.netflix.discovery.DiscoveryClient : DiscoveryClient_LOAN-SERVICE/LAPTOP-UF9J78HT:loan-service:8  
2025-07-18T23:26:28.643+05:30 INFO 10964 --- [freshExecutor-0] com.netflix.discovery.DiscoveryClient : Disable delta property : false  
2025-07-18T23:26:28.644+05:30 INFO 10964 --- [freshExecutor-0] com.netflix.discovery.DiscoveryClient : Single vip registry refresh property : null  
2025-07-18T23:26:28.644+05:30 INFO 10964 --- [freshExecutor-0] com.netflix.discovery.DiscoveryClient : Force full registry fetch : false  
2025-07-18T23:26:28.645+05:30 INFO 10964 --- [freshExecutor-0] com.netflix.discovery.DiscoveryClient : Application is null : false  
2025-07-18T23:26:28.645+05:30 INFO 10964 --- [freshExecutor-0] com.netflix.discovery.DiscoveryClient : Registered Applications size is zero : true  
2025-07-18T23:26:28.645+05:30 INFO 10964 --- [freshExecutor-0] com.netflix.discovery.DiscoveryClient : Application version is -1: false  
2025-07-18T23:26:28.645+05:30 INFO 10964 --- [freshExecutor-0] com.netflix.discovery.DiscoveryClient : Getting all instance registry info from the eureka server  
2025-07-18T23:26:28.724+05:30 INFO 10964 --- [freshExecutor-0] com.netflix.discovery.DiscoveryClient : The response status is 200  
2025-07-18T23:26:45.713+05:30 INFO 10964 --- [nio-8083-exec-2] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring DispatcherServlet 'dispatcherServlet'  
2025-07-18T23:26:45.713+05:30 INFO 10964 --- [nio-8083-exec-2] o.s.web.servlet.DispatcherServlet : Initializing Servlet 'dispatcherServlet'  
2025-07-18T23:26:45.714+05:30 INFO 10964 --- [nio-8083-exec-2] o.s.web.servlet.DispatcherServlet : Completed initialization in 1 ms
```

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 greet() {
    return "Hello World";
}

```

application.properties:

spring.application.name=greet-service

server.port=8080

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

GreetServiceApplication.java:

```

package com.cognizant.greet_service;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;
import org.springframework.cloud.client.discovery.EnableDiscoveryClient;

@SpringBootApplication
@EnableDiscoveryClient

public class GreetServiceApplication {

    public static void main(String[] args) {

        SpringApplication.run(GreetServiceApplication.class, args);

    }
}

```

OUTPUT:

