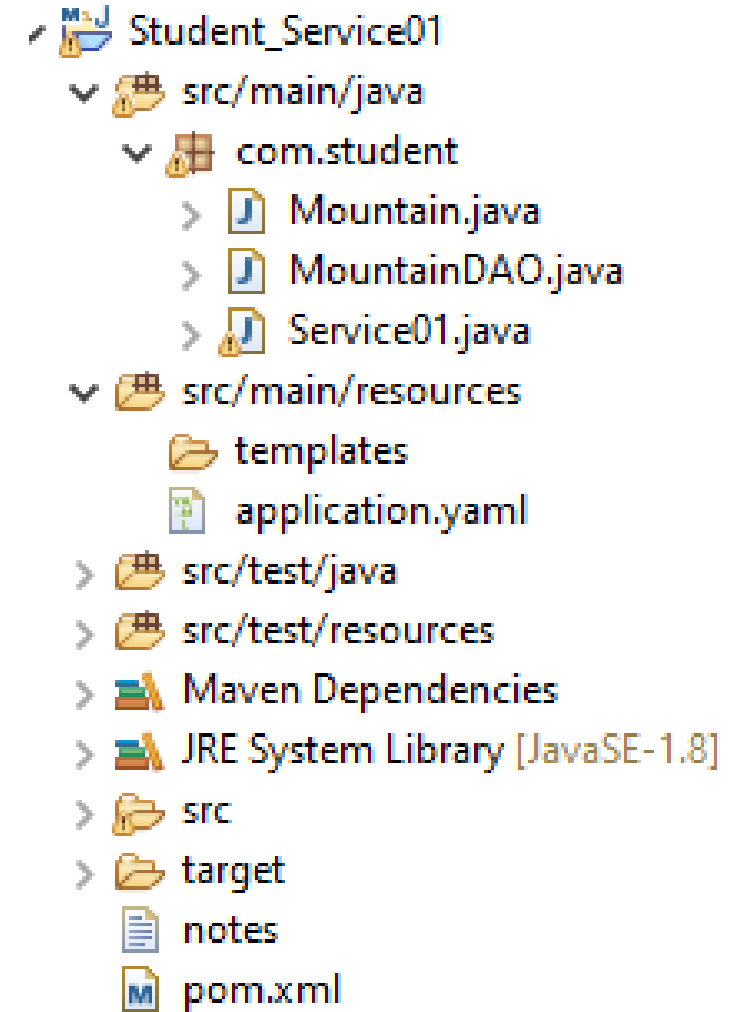# Lab 2.2

In this lab we will register a service, that retrieves the world mountains with our running Eureka Service.

In your eclipse IDE open up the starter project for this lab.

File->Open Projects from File System i.e. developer/Desktop/Studentwork/setup/Lab02.2/Student_Service01

Let the project build

- Student_Service01
  - src/main/java
    - com.student
      - Mountain.java
      - MountainDAO.java
      - Service01.java
  - src/main/resources
    - templates
    - application.yaml
  - src/test/java
  - src/test/resources
  - Maven Dependencies
  - JRE System Library [JavaSE-1.8]
  - src
  - target
  - notes
  - pom.xml

Lab 2.2

Now to make some changes

Open up the class com.student.Service01

Replace the //<TODO> with @EnableDiscoveryClient

In src/main/resources/application.yaml replace the <TODO> with;
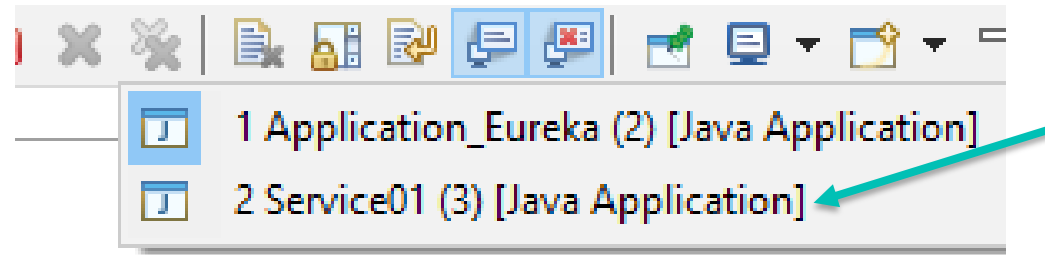
spring:application:name: service01

eureka:client:serviceUrl:defaultZone: http://localhost:1111/eureka/

Lab 2.2

Run the class Service01

Open up the console for this service

Look for the log;

1 Application_Eureka (2) [Java Application]
2 Service01 (3) [Java Application]

```
Registered instance SERVICE01/host.docker.internal:service01:8082 with status UP
```

Refresh your browser http://locaalhost:1111/eureka. You may see a message "EMERGENCY! EUREKA MAY BE INCORRECTLY CLAIMING INSTANCES ARE UP WHEN THEY'RE NOT. RENEWALS ARE LESSER THAN THRESHOLD AND HENCE THE INSTANCES ARE NOT BEING EXPIRED JUST TO BE SAFE". This is latency at work as the service is registered and there is nothing to worry about.

DS Replicas

Instances currently registered with Eureka

Application	AMIs	Availability Zones	Status
SERVICE01	n/a (1)	(1)	UP (1) - host.docker.internal:service01:8082

Click on your service url

Change http://hostname:8082/info to http://hostname:8082/1

```
// http://192.168.1.141:8082/1

{
    "id": 1,
    "name": "Annapurna I",
    "range": "Annapurna Himalaya",
    "country": "Nepal"
}
```