- Lib Discovery-Eureka cung cấp cơ chế client đăng ký vào 1 app tập trung có vai trò giám sát, theo dõi dạng thái của microservice, có thể xem, ip, hostname,port.

- các service con gửi tin hiệu đến registry-server để cho nó viết service con còn hd k.

- giao tiếp, truyền tải, … tới các service con mà không cần sử dụng địa chỉ IP, chỉ cần thông qua host’s name (xem phần cuối **Finaly**)

- bên trong registry-server (discovery uereka) **đã được tích hợp loadbalance bên trong, thông qua openfeign, loadbalance sẽ tự động cân bằng tải tới các service con.**

Deps:

<!-- https://mvnrepository.com/artifact/org.springframework.cloud/spring-cloud-starter-netflix-eureka-server -->  
<dependency>  
 <groupId>org.springframework.cloud</groupId>  
 <artifactId>spring-cloud-starter-netflix-eureka-server</artifactId>  
 <version>${cloud-version}</version>  
</dependency>

Có thể dùng config server để load config từ 1 app config tập trung, thì thêm dev:

<!-- for config server at client (centralize)-->  
<dependency>  
 <groupId>org.springframework.cloud</groupId>  
 <artifactId>spring-cloud-config-client</artifactId>  
 <version>${cloud-version}</version>  
</dependency>

Config tại application.properties

server:  
 port: 9999

Tiếp theo, thêm annotation: **@EnableEurekaServer** vào class **Application.class. => spring tự động config ứng dụng thành 1 ứng dụng discovery**, cho phép các service khác đăng ký vào.

---**Tất cả** Các services muốn đăng ký vào app discovery để giám sát này cần thực hiện:

+ Taji Class maina: thêm **@EnableEurekaClient**

+ config application.properties: (nên để trong resource/shares/application.yml thì sẽ dùng đc cho tất cả các service con)

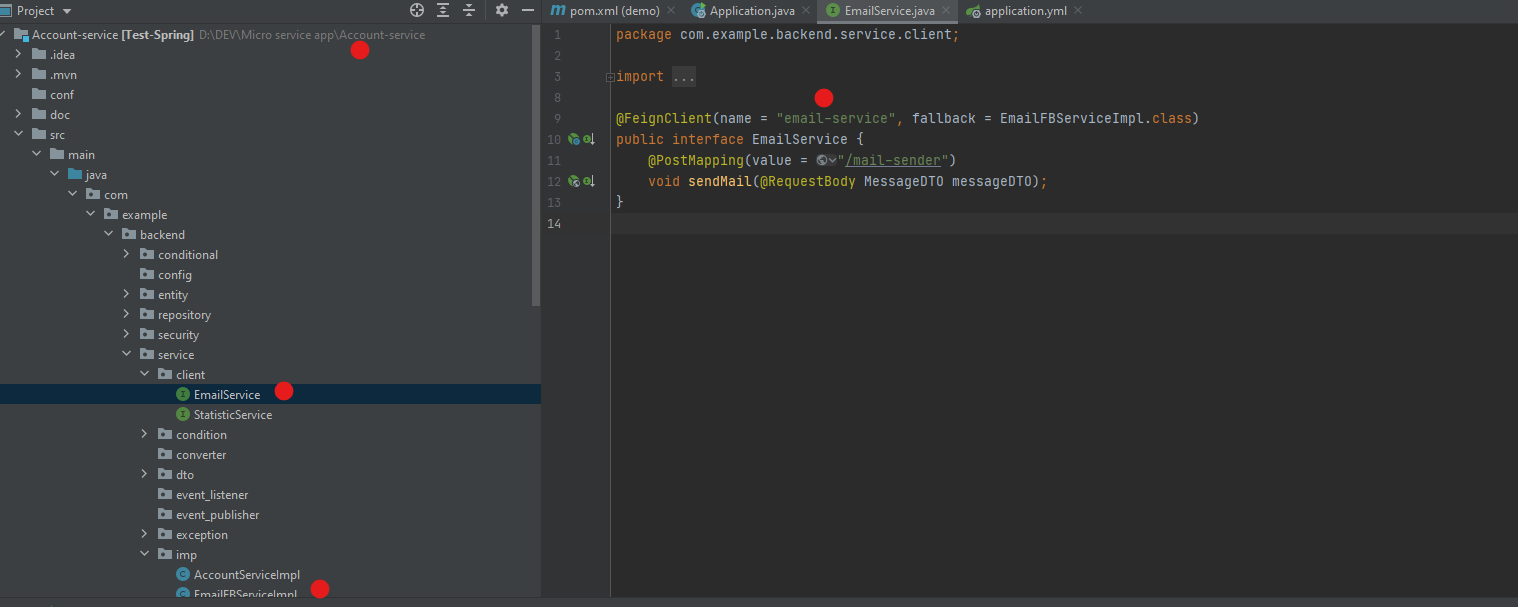
*# config for discoverty-client*

*# Where is our registry server is located*eureka:  
 client:  
 serviceUrl:  
 defaultZone: <http://localhost:9999/eureka>

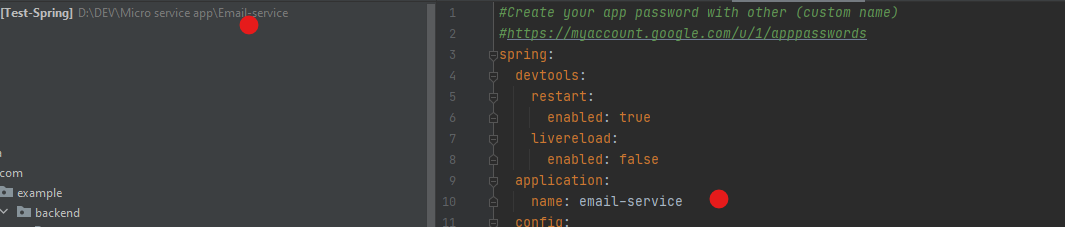
*# services register eureka, true-> they use IP address rather than host name* instance:  
 prefer-ip-address: true

--- test: registry server: <http://localhost:9999>

**Finaly**: Khi đã sử dụng **Registry server** rồi, ta không cần fix cứng địa chỉ url trong @FeignClient nữa, mà chỉ cần chỉ rõ name trùng với name trong application.property của service:



Trong project Email-service:



RUN.