Helpful: https://howtodoinjava.com/spring-cloud/spring-boot-ribbon-eureka/

1st:

----

start ribbon-eureka-server application. (application.properties boilsdown to the following. This app only has one class annotated with RibbonEurekaServerApplication and will start Eureka server.

spring.application.name=eureka-serviceregistry

server.port=8761

eureka.instance.hostname=eureka-serviceregistry

eureka.client.registerWithEureka=false

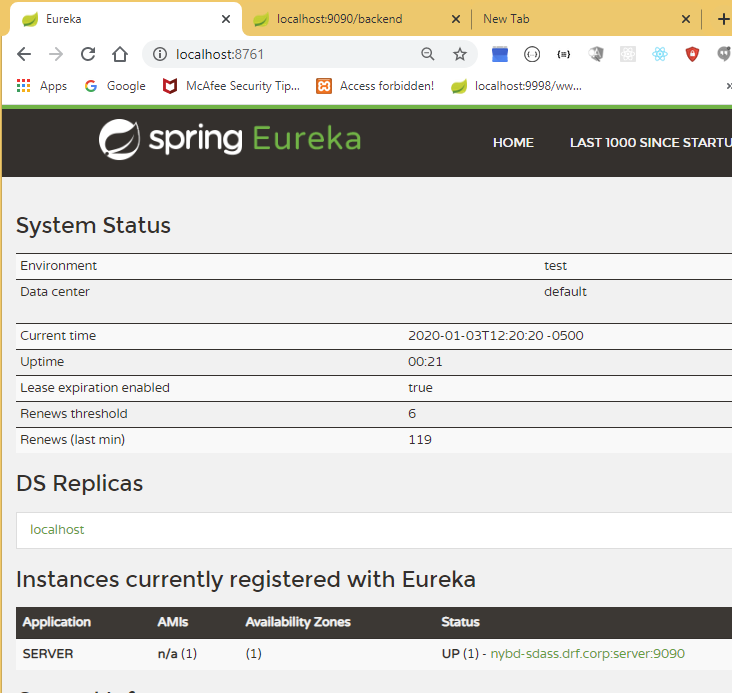
eureka.client.fetchRegistry=false

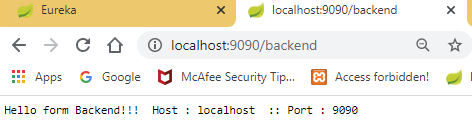
eureka.client.serviceUrl.defaultZone= http://localhost:8761/eureka/

1. After starting ribbon-eureka-server, chrome to <http://localhost:8761>

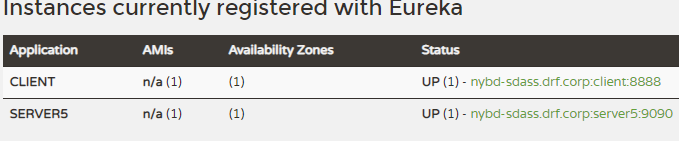
Instances currently registered with Eureka (NO entry under).

1. Now start ribbon-server (it has 2 rest endpoints “/”, “/backend”).Refresh chrome to see entry



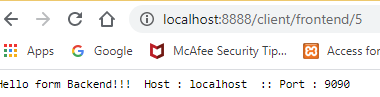


Start ribbon-client. Refresh chrome (I changed rest-server name from server to server5).



Now invoke (<http://localhost:8888/client/frontend/5>)

Get result



Currently server5 rest service running on port 9090. Now start from cmdline 2 more service (port 9092, 9097). Total 3 instances of backend service are running. Original one port 9090

# Caveat debugging

STS compile auto springbootapplication run giving trouble. So do mvn clean build from cmdline

Not in STS but from cmdline do: mvn clean. Next mvn build. Next start 3 servers as below

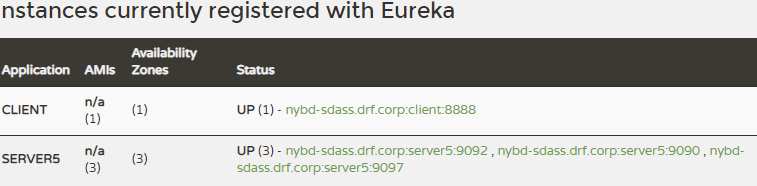
java -jar -Dserver.port=9092 -Dspring.application.name=server5 target\ribbon-server-0.0.1-SNAPSHOT.jar

java -jar -Dserver.port=9097 -Dspring.application.name=server5 target\ribbon-server-0.0.1-SNAPSHOT.jar

#tatking port 9090 from application.properties file

java -jar -Dspring.application.name=server5 target\ribbon-server-0.0.1-SNAPSHOT.jar

see on eureka server



Now hit 3 times <http://localhost:8888/client/frontend/5> url

result:

Hello form Backend!!! Host : localhost :: Port : 9090

Hello form Backend!!! Host : localhost :: Port : 9097

Hello form Backend!!! Host : localhost :: Port : 9092

Meaning this endpoint code calls server5

String randomString = this.restTemplate.getForObject("http://server5/backend", String.class);

The ribbon library helps round robin call to 3 different backend server (server5: 9090, 9092, 9097)