LAB- Load Balancing with Ribbon

Load Balancing - using Ribbon's LoadBalancerClient

In this Lab you will understand how to do client side load balancing with Ribbon.

You will be working on the projects present in 04-**spring-cloud-ribbon** working set

Problem with existing application :

- We are always getting the first instance of service (there could be many instances of service registered)

- In PortfolioService : getting first Server Instance of AccountService
- In QuoteRemoteCallService, getting first Server Instance of QuoteService
-

- In WebApplication :
  MarketService : getting first service instance of QuoteService
  UserService : getting first service instance of Accounts Service

Now, you will be working on **04-02-portfolio-service-start** project
1) Open pom.xml and complete TODO-1
2) Open PortfolioService.java and complete TODO-2 and TODO-3
3) Open QuoteRemoteCallService.java and complete TODO-4 and TODO-5

Run Eureka Server in **03-01-eureka-server-common**

Run 2 instances of QuotesService in 04-01-quotes-service-solution

Run PortfolioApplication.java in 04-02-portfolio-service-start

Go to Browser and check http://localhost:5001/ . Did u observe 2 instances of QuotesService and one instance of PortfolioService

Open PortfolioController in 04-**02-portfolio-service-start**

Observe that there is a method mapped to /portfolio/{id} which returns the portfolio of a user by userid

Observe the code in it . You will understand that it is making a rest call to QuotesService.

Observe getQuotes() inside QuoteRemoteCallService.
It is printing the base uri of quotes service to which it is making the call after loadbalancing

In Browser, go to http://localhost:5001/ and find the base URI of PortfolioService.
Now Give a request to portfolioservice baseurl+/portfolio/{youruserid which you created}

You should get the portfolio details as json.

Now Observe the console of Portfolio Application. you should observe the URI of the quotes service to which request is sent.

Refresh the browser 3 times and observe that quote service uris in the console of PortfolioApplication.

Congratulations !! you know how to do client side load balancing using Ribbon.

Load Balancing - using @LoadBalanced RestTemplate

You have to work on 04-03-portfolio-service-loadbalanced-start

Complete TODO-1 in PortfolioApplication.java
Complete TODO-2, TODO-3 and TODO-4 in QuoteRemoteCallService.java

Run 2 instances of QuotesService in 04-01-quotes-service-solution

Run PortfolioApplication.java in **-03-portfolio-service-loadbalanced-start**

Go to Browser and check http://localhost:5001/ . Did u observe 2 instances of QuotesService and one instance of PortfolioService

In Browser, go to http://localhost:5001/ and find the base URI of PortfolioService.
Now Give a request to portfolioservice baseurl+/portfolio/{youruserid which you created}

You should get the portfolio details as json.

Goto Console and clear the consoles of both Quotes Application Instances.
Now Refresh the browser 4 times.

Observe the console of QuoteService. You should see the logs for every alternate request.

Now Kill once instance of quote service and refresh the browser 2 times. Did you get any exception?

You should get exception. So, RestTemplate is not retrying. If you want RestTemplate to retry, configure the below property in yml file

```
spring:
  cloud:
    loadbalancer:
      retry:
         enabled: true
```

How does this work? Do you remember? If you dont remember, ask me or see video if available to you.

Just observe LoadBalancerAutoConfiguration and RibbonAutoConfiguration and RetryLoadBalancerInterceptor

GoodReads :

https://groups.google.com/forum/#!topic/eureka_netflix/B3uJ0onU_Bo
http://techblog.netflix.com/2013/01/announcing-ribbon-tying-netflix-mid.html

http://stackoverflow.com/questions/29730310/why-client-side-load-balancers-like-ribbon

https://github.com/Oreste-Luci/netflix-oss-example+