LAB- Using FEIGN

In this lab, you will be working on projects under working set 06-spring-cloud-feign

We dont want to write code to make restful calls to the otherservices.

We just want to write the interfaces and want the implementation which makes rest calls to be generated automatically.

If calls to restful services fail, we want fallbacks to be executed.

Now you should work on 06-02-portfolio-service-start

Open pom.xml and complete TODO-1

Open QuoteService interface and observe that the interface methods are annotated with @RequestMapping just like in Spring@MVC Controller.

Complete the TODO-2

Can you tell what is the purpose of fallback configured in @FeignClient ?

Open QuoteServiceCallback and observe that it is implementing QuoteService. This call contains the hystrix fallback methods which should be executed incase the rest calls fail.

Also in Feign client we gave the name registered in Eureka. So, Feign will automatically lookup for the service in Eureka and internally uses Ribbon also for doing load balancing.

Open PortfolioApplication.java in **06-02-portfolio-service-start** and complete TODO-3

After Enabling Feign Clients, proxies will be generated implemeting the interfaces

Similar to QuoteService interface, we have created AccountService also. Go through the annotations in it and also observe the Fallback implementations.

Start EurekaServer in 03-01-eureka-server-common

start 06-01-quotes-service-solution and 06-02-portfolio-service-start

Go to Browser and check eureka console http://localhost:5001/ . You Should observe that all the services are registered.

Now Make a request to base url of portfolioservice +/portfolio/{your user id which you created}

You should observe your portfolio details as json.

So, Feign Clients are able to make restful calls.

Now kill QuoteService application.

Again make   a request to get portfolio details. You should see that you get the details without exceptions. So, hystrix fallbacks are executed when the rest calls fail

How do you configure other hystrix related configuration?

Did you observe how we configure hystrix.command.default.execution.timeout.enabled?

Congratulations !! you know how to use Spring-Feign to create Declarative restful clients