Date : 29-May-21
Spring Boot 9AM
Mr. RAGHU

Workspace:
https://www.mediafire.com/file/yemvi2glnws8opf/SpringCloud9AM_29052021_RAGHU.zip/file

API Gateway : Netflix Zuul

API Gateway: Used for one common entry/exit point
a. Supports Dynamic routing by generating client code[LoadBalancerClient] using Proxy
b. Security at entrance(SSO-Single Sign On --JWT/OAuth2.x)
c. Execute Filters at multiple levels.
  (Error Filter , Routing Filter...etc)

Why API Gateway?
a. In our Application, there can be multiple MS# and having multiple instances.
b. Every Instance may run in different system. So, IP/PORT combinations are
   even muliple.
c. You must give single(one) IP/PORT(or HOST) to FrontEnd App(or Clients).
d. For that we should create one entry and exit point that even excutes
   out MS# based on Request URL.
e. Even it has to choose an instance having less load factor(Load Balancing)

*** So, we use API Gatway.

*) Routing: Executing one MS# instance based on given URL by Client.

*) Dynamic Routing: Chooing MS# IP/PORT(URL) from Eureka at runtime
                  based on LoadFactor.

==========

=> API Gateway also one MS# which is even registered with Eureka.
=> API Gateway calls our one of MS# based on URL
   using internally LoadBalancerClient code which is geneated.
   (Proxy)

=> API Gatway internally uses Proxy concept to make Client calls.

Q) What is difference between feign client and api getway sir
A) If one MS# wants to communicate with another MS# use FeignClient
   interface in MS#1/MS#2 (Create inteface manually).

  => For all MS# we need one common IP/PORT for making call from
    outside of our app. So, this time APIGateway (Generates code)

==========TRUE/FALSE==========

-> API Gateway also one MS# -- True

-> API Gateway supports dynamic routing/LoadBalancing -- TRUE
-> API Gateway internally uses Eureka Server -- TRUE
-> API Gateway define code manually for MS#
   Communication ---FALSE [Proxy Generated using LoadBalancerClient]

-> API Gateway provide Single IP/PORT to FrontEnd Apps -- TRUE
-> only API Gateway call our MS# --- TURE
-> There can be multiple API Gateway in one Project -- FALSE
======(coding steps)=======================================
1. Eureka Server
2. one MS# with 3 instances
3. API Gatway (Netflix Zuul)

===(code)===
1. Eureka Server
Name : SpringCloudEurekaServer
Dep : Eureka Server

=> At starter : @EnableEurekaServer
=> at properties
server.port=8761

eureka.client.register-with-eureka=false
eureka.client.fetch-registry=false
----
2. MS# app :
name : SpringCloudVendorService
Dep : Eureka Discovery Client, Web

=> At starter : @EnableEurekaClient
=> properties
server.port=9096
#serviceId
spring.application.name=VENDOR-SERVICE
#Eureka Location
eureka.client.service-url.defaultZone=http://localhost:8761/eureka
#InstanceId
eureka.instance.instance-id=${spring.application.name}:${random.value}

=> RestController
```
package in.nareshit.raghu.rest;

import org.springframework.beans.factory.annotation.Value;
import org.springframework.http.ResponseEntity;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RestController;

@RestController
@RequestMapping("/vendor")
public class VendorRestController {

	@Value("${server.port}")
	private String port;

	@GetMapping("/msg")
	public ResponseEntity<String> showVenMsg() {
```

```
                return ResponseEntity.ok("FROM VENDOR " + port);
        }
}
---------------------------------
3. API Gatway (Netflix Zuul)
Name : SpringCloudApiGatewayZuul
Dep  : Zuul, Eureka Discovery Client , web

=> At starter class : @EnableEurekaClient, @EnableZuulProxy
=> at properties    :
server.port=80
#serviceId
spring.application.name=ZUUL-PROXY
#Eureka Location
eureka.client.service-url.defaultZone=http://localhost:8761/eureka

# Path -- ServiceId
zuul.routes.vendor.path=/vendor-api/**
zuul.routes.vendor.service-id=VENDOR-SERVICE

--Execution order---------
1. Eureka Server
2. MS#
3. Zuul App
4. Goto eureka http://localhost:8761/
5. click zuul link
   http://192.168.0.8/actuator/info

   Modify as:
   http://192.168.0.8/vendor-api/vendor/msg

-----------------------------------------------------------------
Q) AWS upscaling and downscaling ?
A)

Q) AWS Request timeout?
A)

Q) AWS Lambda ?
A)

Q)
A)
```