Date : 18/03/2021
Spring Boot 6PM
Mr. RAGHU

Code (Config Server)
https://www.mediafire.com/file/pq3uidxqunv41ou/SpringCloudConfigServerExternal_18032021_RAGHU.zip/file
Git
https://www.youtube.com/watch?v=T2UHpsxJ-2o
https://www.youtube.com/watch?v=38UGVeXuj3Q

Spring Cloud - Config Server

*) In our Application, there can be multiple MS# and instances exist.
*) Every MS# may have few common key=value(both)
   in that case they are considered as duplicates,
   even modifications/maintanance becomes complex.

*) Solution given is : Config Server
  Common key=val placing outside of all MS# Projects.

===

=> Config Server behaves like mediator
=> It runs on default port 8888. We can modify even.
=> At MS# just add config client dependency only.
   ***No additional coding required (that has inernal code
     communicates to http://localhost:8888)
=> Config Types are 2.
  a. External Config (Used in realtime)
  b. Native Config (Used for Dev/Test Env only)

----

  a. External Config : In this case of one git account is used.
     and we place application.properties (or) application.yml

  => We need to create one Project "ConfigServer" along with MS.
  => Inside this configserver also provide one properties file
     that holds location of git account.

  =>** Finally at 3 places we have properties file
    #1 MS# level, #2 Config Server , #3 Git Account/Native Level
    (Specific keys)   [link location]   [commonkey=val]

Q) When we run MS# what will happen?
A)
   First MS# executes Config Client
   Client Communicates with ConfigServer(default 8888)
   Config Server gets common-key=vals from External/Native
   Given it back to Config Client
   Merge with MS# project
   Start MS# Application with all setup
   Finally Register with Eureka Server.

===========Full code of External Config Server(ECS)==================

1. Eureka Server (same as before)

Name: SpringCloudECSEurekaServer
Dep : Eureka Server

=> At starter class: @EnableEurekaServer

=> application.properties

```
server.port=8761
eureka.client.register-with-eureka=false
eureka.client.fetch-registry=false
```

```
GitLab Account
#a) Register
https://gitlab.com/users/sign_up
#b) Verify Email Address
#c) Login
https://gitlab.com/users/sign_in
#d) Create new Project (https://gitlab.com/projects/new)
  > select blank Project
  > Enter Project name : myconfigtestnew
  > Create Project
  > check URL as
   https://gitlab.com/<username>/<projectName>
   https://gitlab.com/javaraghu2018/myconfigtestnew

 > click on New File option (one more time even)
 > Enter name : application.properties > create button
 > Provide data (key=val) in Edit Section
    [ex : my.app.title=NIT-ONE-TEST ]
 > Click on Commit > Enter commit message (any dummy)
 > Click on Commit

 Final Location is :
https://gitlab.com/javaraghu2018/myconfigtestnew.git
```

----

2. Config Server

Name: SpringCloudECSConfigServer
Dep : Config Server

=> At starter class: @EnableConfigServer

=> application.properties

```
server.port=8888
spring.cloud.config.server.git.uri=https://gitlab.com/javaraghu2018/my
configtestnew.git
```

----

3. MS# Project

Name: SpringCloudECSEmployeeService
Dep : Config Client, Spring Web, Eureka Discovery Client

=> At starter class: @EnableEurekaClient

```
=> application.properties
server.port=8086
spring.application.name=EMPLOYEE-SERVICE
eureka.client.service-url.defaultZone=http://localhost:8761/eureka
```

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

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

@RestController
@RequestMapping("/emp")
public class EmployeeRestController {

        @Value("${my.app.title}")
        private String title;

        @GetMapping("/data")
        public String showKeyData() {
                return "FROM EMP-APP " + title;
        }
}
```

```
====================Execution Order=========================
1. Config Server
2. Eureka Server
3. MS# App
Check with URL:
http://localhost:8086/emp/data


*)Note:
=> Config Server never registred with Eureka
=> No need of adding code for Config Client at MS#
   not even annotation required like @ConfigClient
=> inside MS# App(Emp-Service) we did not provide
   any location of config server manually.
   Only added config client, that executes
   default communication URL as 'http://localhost:8888'

=> When we run MS# you check at console/log first line as
   ConfigServicePropertySourceLocator=>
      Fetching config from server at : http://localhost:8888
-----------------------------------------------------------
Task:
I gave GitLab Account Steps
You try github/bitbucket account steps

==========================================
HQL/JPQL Joins
SELECT <p>.<code>
FROM  <ParentModelClass>  <p>
       [JOIN TYPE]
   <p>.<HasAVariableName> as <C>
```

```
WHERE <c>.<id>=?
------------------------------------
clas A {
 id,code
}
class B{
  id, mode
  A oa; //HAS-A
}
----------------------------
SELECT b.mode, a.code
FROM  B  b
   INNER JOIN
 b.oa  as a
WHERE a.id=?
```