Date : 12-May-21
Spring Boot 9AM
Mr. RAGHU

Link:
https://www.mediafire.com/file/x1e3o8v5e2v7e75/SpringCloudOpenFeignCode_12052021_9AM_RAGHU.zip/file

Microservices : Feign Client(FC)

Monolithic App: Publisher---<>Book

---

1. Eureka Server
Name : SpringCloudFCEurekaServer
Dep : Eureka Server

=> At starter class: @EnableEurekaServer
=> properties

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

---

2. Producer Application
Name : SpringCloudFCBookServices
Dep : Lombok, Web, Eureka Discovery Client

=> At starter class: @EnableEurekaClient
=> Properties

```
server.port=9900
spring.application.name=BOOK-SERVICE
eureka.client.service-url.defaultZone=http://localhost:8761/eureka
eureka.instance.instance-id=${spring.application.name}:${random.value}
```

=> Model class

```
package in.nareshit.raghu.model;

import lombok.AllArgsConstructor;
import lombok.Data;
import lombok.NoArgsConstructor;

@Data
@NoArgsConstructor
@AllArgsConstructor
public class Book {

        private Integer bookId;
        private String bookName;
        private String bookAuthor;
}
```

=> RestController

```
package in.nareshit.raghu.rest;

import java.util.Arrays;
import java.util.List;
import org.springframework.http.ResponseEntity;
import org.springframework.web.bind.annotation.DeleteMapping;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.PathVariable;
import org.springframework.web.bind.annotation.PostMapping;
import org.springframework.web.bind.annotation.RequestBody;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RestController;

import in.nareshit.raghu.model.Book;

@RestController
@RequestMapping("/books")
public class BookRestController {

        @GetMapping("/all")
        public ResponseEntity<List<Book>> getAllBooks(){
                return
                                ResponseEntity.ok(
                                                Arrays.asList(
                                                                new
Book(10, "Core Java", "SAM"),
                                                                new
Book(11, "Adv Java", "RAM"),
                                                                new
Book(12, "Spring Boot", "SAM"),
                                                                new
Book(13, "Microservices", "SYED")
                                                                )
                                                );
        }

        @GetMapping("/find/{id}")
        public ResponseEntity<Book> getOneBook(
                        @PathVariable Integer id
                        )
        {
                return ResponseEntity.ok(
                                new Book(id, "Dummy Book", "Dummy
Author"));
        }

        @PostMapping("/create")
        public ResponseEntity<String> createNewBook(
                        @RequestBody Book book
                        )
        {
                return ResponseEntity.ok(book.toString());
        }

        @DeleteMapping("/remove/{id}")
        public ResponseEntity<String> deleteBook(
                        @PathVariable Integer id
                        )
        {
                return ResponseEntity.ok("Book deleted! " + id);
        }
}
```

==

3. Consumer Application
Name : SpringCloudFCPublisherService
Dep : Web, lombok, Eureka Discovery Client, open feign

=> At starter class: @EnableEurekaClient, @EnableFeignClients
=> Properties

```
server.port=9880
spring.application.name=PUBLISHER-SERVICE
eureka.client.service-url.defaultZone=http://localhost:8761/eureka
```

=> Consumer interface
*** Copy Model class also from Producer to Consumer

*) Model

```
package in.nareshit.raghu.model;

import lombok.AllArgsConstructor;
import lombok.Data;
import lombok.NoArgsConstructor;

@Data
@NoArgsConstructor
@AllArgsConstructor
public class Book {

        private Integer bookId;
        private String bookName;
        private String bookAuthor;
}
```

*) Consumer

```
package in.nareshit.raghu.consumer;

import java.util.List;

import org.springframework.cloud.openfeign.FeignClient;
import org.springframework.http.ResponseEntity;
import org.springframework.web.bind.annotation.DeleteMapping;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.PathVariable;
import org.springframework.web.bind.annotation.PostMapping;
import org.springframework.web.bind.annotation.RequestBody;

import in.nareshit.raghu.model.Book;

@FeignClient("BOOK-SERVICE")
public interface BookRestConsumer {

        @GetMapping("/books/all")
        public ResponseEntity<List<Book>> getAllBooks();
        @GetMapping("/books/find/{id}")
        public ResponseEntity<Book> getOneBook(
                        @PathVariable Integer id);

        @PostMapping("/books/create")
        public ResponseEntity<String> createNewBook(
                        @RequestBody Book book);


        @DeleteMapping("/books/remove/{id}")
        public ResponseEntity<String> deleteBook(
                        @PathVariable Integer id);
}
```

*) RestController

```
package in.nareshit.raghu.rest;

import java.util.List;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.http.ResponseEntity;
import org.springframework.web.bind.annotation.DeleteMapping;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.PathVariable;
import org.springframework.web.bind.annotation.PostMapping;
import org.springframework.web.bind.annotation.RequestBody;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RestController;

import in.nareshit.raghu.consumer.BookRestConsumer;
import in.nareshit.raghu.model.Book;

@RestController
@RequestMapping("/publisher")
public class PublisherRestController {

        @Autowired
        private BookRestConsumer consumer;

        @GetMapping("/view")
        public String viewPublisherBooks() {
                ResponseEntity<List<Book>> resp =
consumer.getAllBooks();
                return "FROM PUBLISHER =>" + resp.getBody();
        }

        @GetMapping("/getOne")
        public String viewOnePublisherBook() {
                ResponseEntity<Book> resp = consumer.getOneBook(550);
                return "FROM PUBLISHER ONE BOOK=>" + resp.getBody();
        }

        @PostMapping("/save")
        public String createOnePublisherBook(
                        @RequestBody Book book
                        )
        {
                ResponseEntity<String> resp =
consumer.createNewBook(book);
                return "FROM PUBLISHER SAVE BOOK=>" + resp.getBody();
        }

        @DeleteMapping("/del/{id}")
        public String removeOnePublisherBook(
                        @PathVariable Integer id
                        )
        {
                ResponseEntity<String> resp = consumer.deleteBook(id);
                return "FROM PUBLISHER DELETE BOOK=>" +
resp.getBody();
        }
}
```

```
==============postman screen================================
1.
GET  http://192.168.0.3:9880/publisher/view   SEND

2.
GET  http://192.168.0.3:9880/publisher/getOne  SEND

3.
POST  http://192.168.0.3:9880/publisher/save   SEND
          Body
               raw(*)   JSON
{
    "bookId" : 99,
    "bookName" : "Test",
    "bookAuthor" : "SAMPLE"
}

4.
DELETE  http://192.168.0.3:9880/publisher/del/1234  SEND


1-MS----1 Module  --- 1 RestController
1--MS --* Module Integration --- n RestConsumers
```