Openfeign 적용하기

RestTemplate을 간편하고 간단하게 사용하기 위함

User, Alim 프로젝트 의존성 추가

backend-user C:\Workspace\spring-cloud-4\backend

- .gradle
- .idea
- build
- gradle
- src
- .gitattributes
- .gitignore
- build.gradle
- gradlew
- gradlew.bat
- HELP.md
- settings.gradle
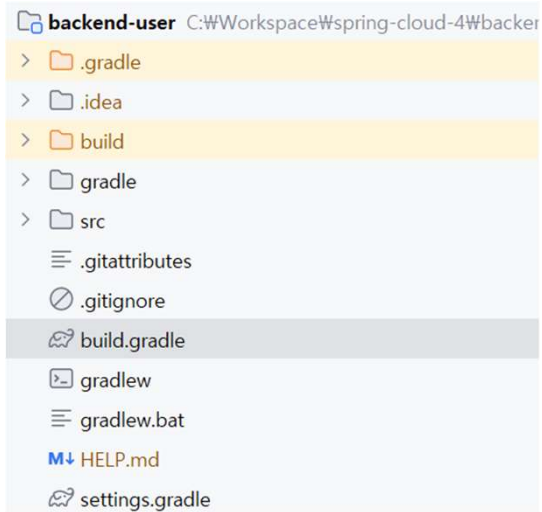
```
dependencies {
    implementation 'org.springframework.boot:spring-boot-starter-web'
    implementation 'org.springframework.cloud:spring-cloud-starter-loadbalancer'
    implementation 'org.springframework.cloud:spring-cloud-starter-netflix-eureka-client'
    implementation 'org.springframework.cloud:spring-cloud-starter-openfeign'
    compileOnly 'org.projectlombok:lombok'
    annotationProcessor 'org.projectlombok:lombok'

    testImplementation 'org.springframework.boot:spring-boot-starter-test'
    testRuntimeOnly 'org.junit.platform:junit-platform-launcher'
}
```

User, Alim Application에 어노테이션 추가

```
src
main
  java
    com.welab.backend_user
      api.open
      config
      remote.alim
      BackendUserApplication
  resources
    static
    templates
    application.yml
    application-local.yml
```
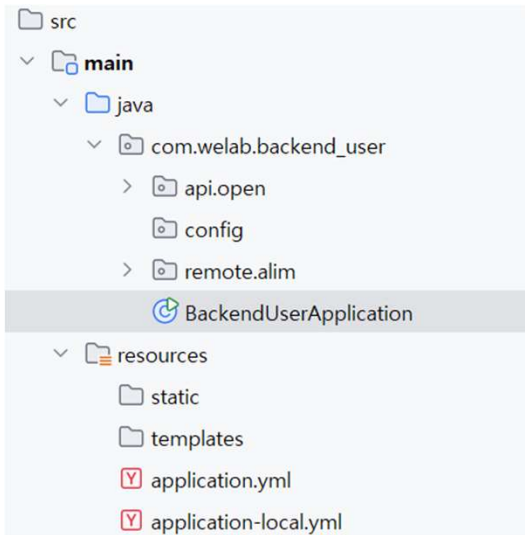
```java
@EnableDiscoveryClient
@EnableFeignClients
@SpringBootApplication
public class BackendUserApplication {

    public static void main(String[] args) {
        SpringApplication.run(BackendUserApplication.class, args);
    }

}
```
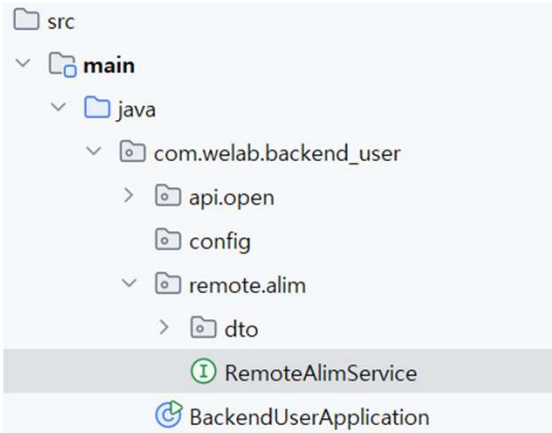
User 프로젝트 코드 작성

RestTemplate -> FeignClient 이용으로 변경

```
[API       ]
+  "backend- alim"
+  "/backend/alim/v1/**"    API
```

```
src
  main
    java
      com.welab.backend_user
        api.open
        config
        remote.alim
          dto
          RemoteAlimService
        BackendUserApplication
```

```java
@FeignClient(name = "backend-alim", path = "/backend/alim/v1")
public interface RemoteAlimService {
    @GetMapping(value = "/hello")
    public String sms();

    @PostMapping(value = "/sms")
    public SendSmsDto.Response sendSms(@RequestBody SendSmsDto.Request request);
}
```