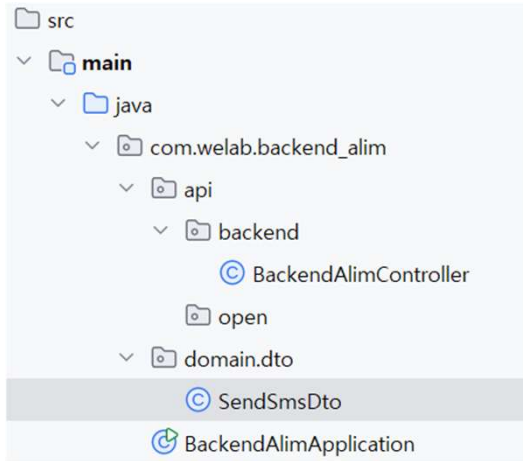


**Post Api 호출도
연습해 보자**

Alim 프로젝트 코드 작성

private
-> 가
-> Request & Response !



```
@NoArgsConstructor(access = AccessLevel.PRIVATE)  
public class SendSmsDto {
```

```
    @Getter @Setter  
    public static class Request {  
        private String userId;  
        private String phoneNumber;  
        private String title;  
        private String message;  
    }
```

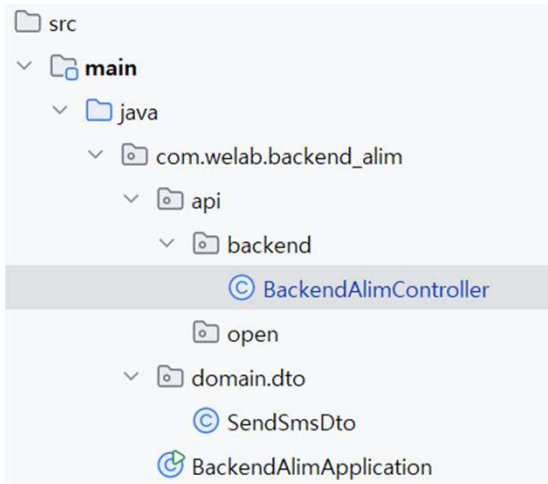
```
    @Getter @Setter  
    public static class Response {  
        private String result;  
    }
```

```
}
```

보기 수월한 코드

(굳이 두 클래스를 나눠서 구성할 필요는 없음)

Alim 프로젝트 코드 작성

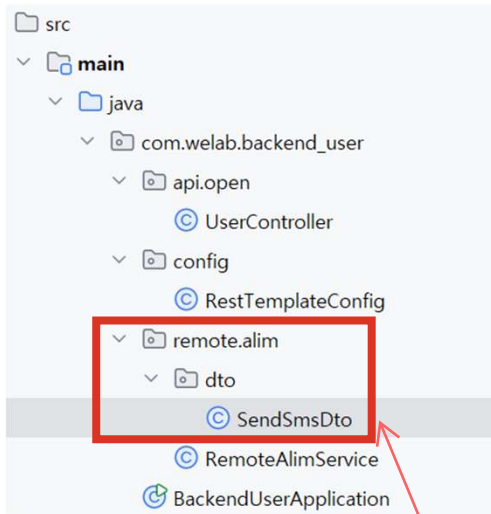


```
@Slf4j
@RestController
@RequestMapping(value = "/backend/alim/v1",
    produces = MediaType.APPLICATION_JSON_VALUE)
public class BackendAlimController {
    @GetMapping(value = "/hello")
    public String hello() {
        return "알림 백엔드 서비스가 호출되었습니다.";
    }

    @PostMapping(value = "/sms")
    public SendSmsDto.Response sendSms(@RequestBody SendSmsDto.Request request) {
        log.info("sendSms: userId={}", request.getUserId());
        SendSmsDto.Response response = new SendSmsDto.Response();
        response.setResult("OK");
        return response;
    }
}
```

"sms" API 요청 시, Dto의 요청, 응답 클래스 사용

User 프로젝트 코드 작성

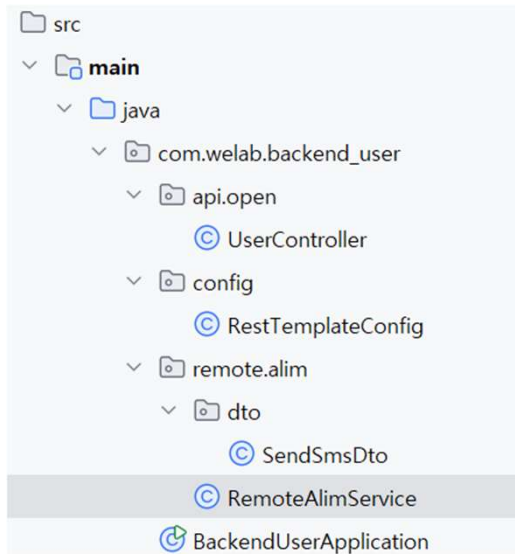


remote
DTO

```
@NoArgsConstructor(access = AccessLevel.PRIVATE)
public class SendSmsDto {
    @Getter
    @Setter
    public static class Request {
        private String userId;
        private String phoneNumber;
        private String title;
        private String message;
    }

    @Getter @Setter
    public static class Response {
        private String result;
    }
}
```

User 프로젝트 코드 작성



```
@Service
@RequiredArgsConstructor
public class RemoteAlimService {
    private final RestTemplate restTemplate;

    public String callAlimSms() {
        return restTemplate.getForObject(
            "http://backend-alim/backend/alim/v1/hello",
            String.class
        );
    }
}
```

```
public SendSmsDto.Response sendSms(SendSmsDto.Request request) {
    HttpHeaders headers = new HttpHeaders();
    headers.setContentType(MediaType.APPLICATION_JSON);

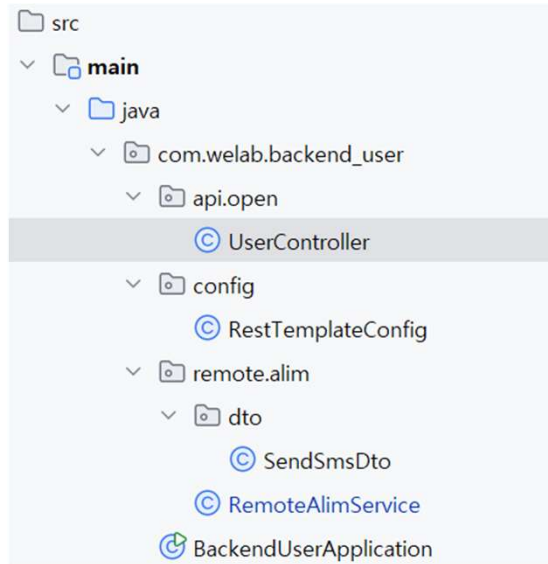
    HttpEntity<SendSmsDto.Request> httpRequest = new HttpEntity<>(request, headers);

    ResponseEntity<SendSmsDto.Response> httpResponse = restTemplate.exchange(
        "http://backend-alim/backend/alim/v1/sms",
        HttpMethod.POST,
        httpRequest,
        SendSmsDto.Response.class
    );

    return httpResponse.getBody();
}
```

이 부분을 간단히 작성할 방법은 없을까?

User 프로젝트 코드 작성



```
@Slf4j
@RestController
@RequestMapping(value = "/api/user/v1", produces = MediaType.APPLICATION_JSON_VALUE)
@RequiredArgsConstructor
public class UserController {
    private final RemoteAlimService remoteAlimService;

    @GetMapping(value = "/test")
    public String test() {
        return remoteAlimService.callAlimSms();
    }

    @PostMapping(value = "/sms")
    public String sms(@RequestBody SendSmsDto.Request request) {
        var response = remoteAlimService.sendSms(request);
        return "OK";
    }
}
```

[]

,

()