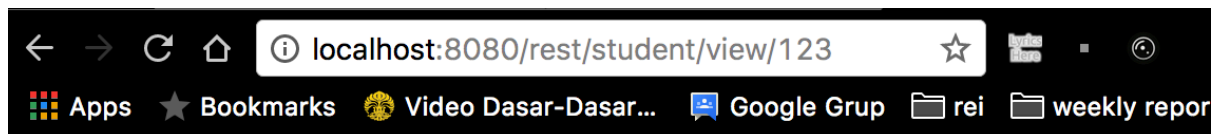


## Membuat *Service producer*



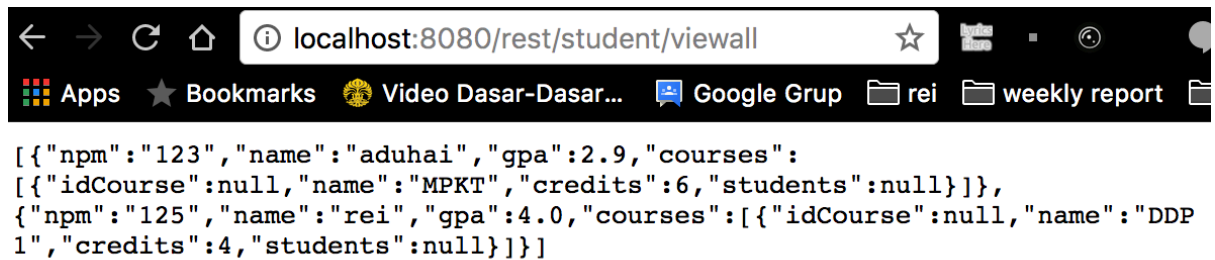
```
{"npm": "123", "name": "aduhai", "gpa": 2.9, "courses": [{"idCourse": null, "name": "MPKT", "credits": 6, "students": null}]}
```

| String parse  | JS eval   |
|---|---|
| <pre>{   "npm": "123",   "name": "aduhai",   "gpa": 2.9,   "courses": [     {       "idCourse":         null,       "name": "MPK T",       "credits": 6     },     {       "students":         null     }   ] }</pre> | <pre>{   "npm": "123",   "name": "aduhai",   "gpa": 2.9,   "courses": [     {       "idCourse":         null,       "name": "MPK T",       "credits": 6     },     {       "students":         null     }   ] }</pre> |

### Latihan:

1. Jawaban: penambahan pada kelas **StudentRestController.java**

```
@RequestMapping("/student/viewall")
public List<StudentModel> view ()
{
    List<StudentModel> students = studentService.selectAllStudents ();
    return students;
}
```



```
[{"npm": "123", "name": "aduhai", "gpa": 2.9, "courses": [{"idCourse": null, "name": "MPKT", "credits": 6, "students": null}]}, {"npm": "125", "name": "rei", "gpa": 4.0, "courses": [{"idCourse": null, "name": "DDP 1", "credits": 4, "students": null}]}]
```

## 2. Jawaban: penambahan padakelas **CourseRestController.java**

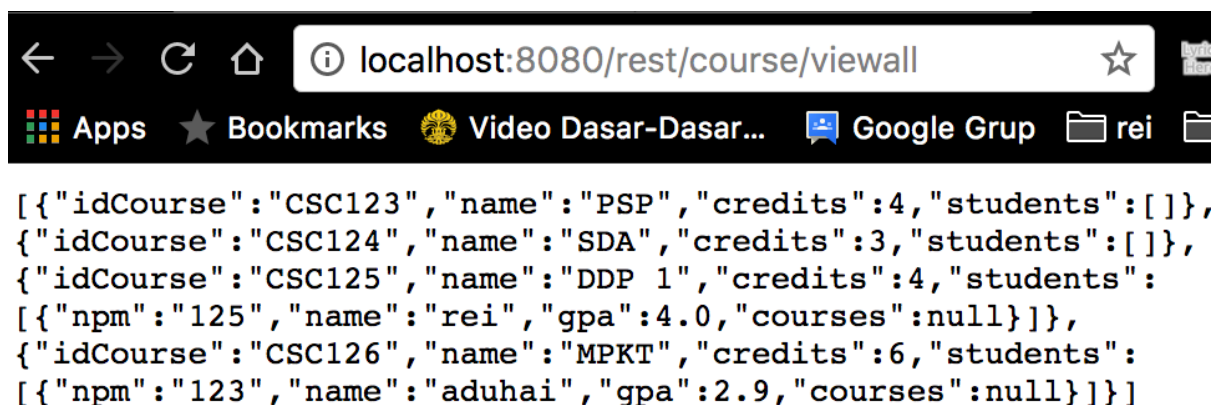
```
package com.example.rest;

import java.util.List;

@RestController
@RequestMapping("/rest")
public class CourseRestController {
    @Autowired
    CourseService courseService;

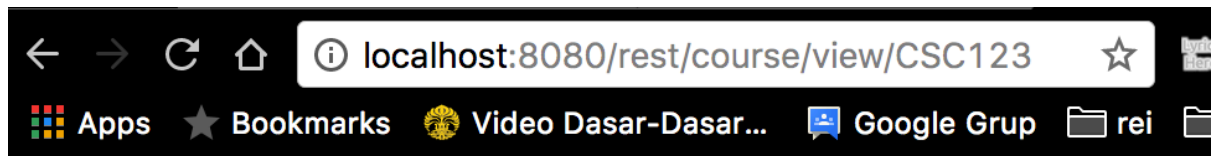
    @RequestMapping("/course/view/{id_course}")
    public CourseModel view(@PathVariable(value = "id_course") String id_course) {
        CourseModel course = courseService.selectCourse(id_course);
        return course;
    }

    @RequestMapping("/course/viewall")
    public List<CourseModel> view ()
    {
        List<CourseModel> courses = courseService.selectAllCourses ();
        return courses;
    }
}
```



```
[{"idCourse": "CSC123", "name": "PSP", "credits": 4, "students": []}, {"idCourse": "CSC124", "name": "SDA", "credits": 3, "students": []}, {"idCourse": "CSC125", "name": "DDP 1", "credits": 4, "students": [{"npm": "125", "name": "rei", "gpa": 4.0, "courses": null}]}, {"idCourse": "CSC126", "name": "MPKT", "credits": 6, "students": [{"npm": "123", "name": "aduhai", "gpa": 2.9, "courses": null}]}]
```

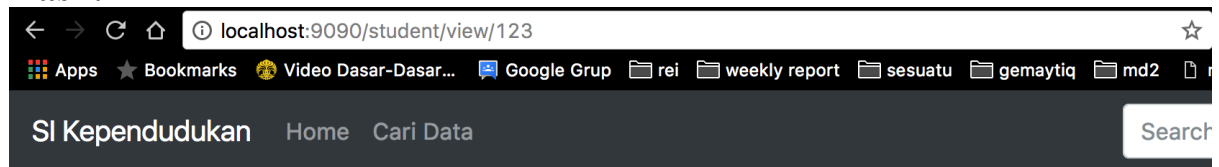
Reitasya Devina  
1406623663



```
{"idCourse": "CSC123", "name": "PSP", "credits": 4, "students": []}
```

## Membuat *Service Consumer*

### Hasil:



### Lihat Data Mahasiswa

NPM = 123

Name = aduhai

GPA = 2.9

Kuliah yang diambil

- MPKT-6 sks

by Reitasya Devina, 2017. Tugas 1 Arsitektur dan Perancangan Aplikasi.

## Latihan

1. **Latihan 3:** Implementasikan *service consumer* untuk view all Students dengan melengkapi *method* `selectAllStudents` yang ada di kelas **StudentServiceRest**.
  - a. Method yang saya ubah yaitu seperti dibawah:

Reitasya Devina  
1406623663

```
@Service
public class StudentDAOImpl implements StudentDAO {

    @Autowired
    private RestTemplate restTemplate;

    @Override
    public StudentModel selectStudent(String npm) {
        StudentModel student = restTemplate.getForObject("http://localhost:8080/rest/student/view/" + npm,
            StudentModel.class);
        return student;
    }

    @Override
    public List<StudentModel> selectAllStudents () {
        List<StudentModel> students = restTemplate.getForObject("http://localhost:8080/rest/student/viewall", List.class);
        return students;
    }
}
```

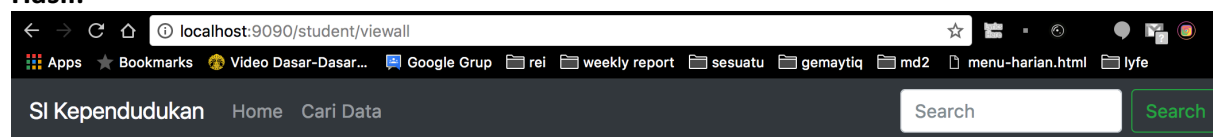
```
public class StudentServiceRest implements StudentService {

    @Autowired
    private StudentDAO studentDAO;

    @Override
    public StudentModel selectStudent (String npm)
    {
        log.info ("REST - select student with npm {}", npm);
        return studentDAO.selectStudent(npm);
    }

    @Override
    public List<StudentModel> selectAllStudents ()
    {
        log.info ("REST - select all students");
        List<StudentModel> students = studentDAO.selectAllStudents();
        return students;
    }
}
```

Hasil:



## All Students

Student GPA Student GPA  
Show 10 entries

Search:

| No. ▲ | Student NPM ▲ | Student Name ▲ | Student GPA ▲ | Status ▲          | Delete ▲ | Update ▲ |
|-------|---------------|----------------|---------------|-------------------|----------|----------|
| No. 1 | NPM = 123     | Name = aduhai  | GPA = 2.9     | Sangat Memuaskan! | Delete   | Update   |
| No. 2 | NPM = 125     | Name = rei     | GPA = 4.0     | Cum Laude!        | Delete   | Update   |

Showing 1 to 2 of 2 entries

Previous 1 Next

by Reitasya Devina, 2017. Tugas 1 Arsitektur dan Perancangan Aplikasi.

## 2. Latihan 4: Implementasikan *service consumer* untuk *class* CourseModel dengan membuat *class-class* DAO dan *service* baru.

### Code yang diubah dan buat:

```
1 package com.example.dao;
2
3 import java.util.List;
4
15 @Service
16 public class CourseDAOImpl implements CourseDAO {
17
18     @Autowired
19     private RestTemplate restTemplate;
20
21     @Override
22     public CourseModel selectCourse(String id_course) {
23         CourseModel course = restTemplate.getForObject("http://localhost:8080/rest/course/view/" + id_course,
24             CourseModel.class);
25         return course;
26     }
27
28     @Override
29     public List<CourseModel> selectAllCourse () {
30         List<CourseModel> courses = restTemplate.getForObject("http://localhost:8080/rest/course/viewall", List.class);
31         return courses;
32     }
33
34
35
36 }
37
@Slf4j
@Service
@Primary
public class CourseServiceRest implements CourseService {

    @Autowired
    private CourseDAO courseDAO;

    @Override
    public CourseModel selectCourse (String id_course)
    {
        log.info ("REST - select course with id_course {}", id_course);
        return courseDAO.selectCourse(id_course);
    }

    @Override
    public List < CourseModel > selectAllCourses ()
    {
        log.info ("REST - select all course");
        List<CourseModel> courses = courseDAO.selectAllCourse();
        return courses;
    }
}
```

Reitasya Devina  
1406623663

## Hasil:

| SI Kependudukan          |  | Home | Cari Data | Search | Search |
|--------------------------|--|------|-----------|--------|--------|
| All Course               |  |      |           |        |        |
| No. 1                    |  |      |           |        |        |
| idCourse = CSC123        |  |      |           |        |        |
| Name = PSP               |  |      |           |        |        |
| SKS = 4                  |  |      |           |        |        |
| Mahasiswa yang mengambil |  |      |           |        |        |
| No. 2                    |  |      |           |        |        |
| idCourse = CSC124        |  |      |           |        |        |
| Name = SDA               |  |      |           |        |        |
| SKS = 3                  |  |      |           |        |        |
| Mahasiswa yang mengambil |  |      |           |        |        |
| No. 3                    |  |      |           |        |        |
| idCourse = CSC125        |  |      |           |        |        |
| Name = DDP 1             |  |      |           |        |        |
| SKS = 4                  |  |      |           |        |        |
| Mahasiswa yang mengambil |  |      |           |        |        |
| • 125-rei                |  |      |           |        |        |
| No. 4                    |  |      |           |        |        |
| idCourse = CSC126        |  |      |           |        |        |
| Name = MPKT              |  |      |           |        |        |
| SKS = 6                  |  |      |           |        |        |
| Mahasiswa yang mengambil |  |      |           |        |        |
| • 123-aduhai             |  |      |           |        |        |

by Reitasya Devina, 2017. Tugas 1 Arsitektur dan Perancangan Aplikasi.

localhost:9090/course/view/CSC125

Apps

★ Bookmarks

🔔 Video Dasar-Dasar...

📄 Google Grup

📁 rei

📁 weekly report

📁 sesuatu

📁 gemaytiq

📁 md2

📄 menu-harian.html

📁 lyfe

📁 u kno

SI Kependudukan

Home

Cari Data

Search

ID = CSC125

Nama = DDP 1

SKS = 4

Mahasiswa yang mengambil

• 125-rei

by Reitasya Devina, 2017. Tugas 1 Arsitektur dan Perancangan Aplikasi.