

**TUTORIAL 7**  
**ARSITEKTUR**  
**DAN PEMROGRAMAN APLIKASI PERUSAHAAN**



**Nama : Julio Muhammad Pranadano**

**NPM : 1706106785**

**FAKULTAS ILMU KOMPUTER**

**UNIVERSITAS INDONESIA**

**DEPOK**

**2018**

## A. Hal Yang Dipelajari

Pada tutorial ini mempelajari *web service* yang digunakan untuk pertukaran data antar satu aplikasi dengan aplikasi lainnya baik itu dengan bahasa pemrograman yang sama maupun tidak, hal ini dapat dilakukan antar platform yang sama maupun yang berbeda. Pada tutorial ini menggunakan dua project yaitu **producer** yang bertindak sebagai penyedia data dan **consumer** sebagai pengguna data yang mengambil data melalui *resttemplate* dari producer. Pada sisi consumer akan mengolah data tersebut yang akan digunakan oleh user. Pada *StudentServiceRest* di package service pada sisi consumer, terdapat `@Primary` yang berfungsi sebagai default service agar Service yang digunakan adalah service rest tersebut.

## B. Latihan

1. Membuat service agar mengembalikan seluruh data siswa pada database dan di mapping ke `rest/student/viewall` pada project producer yang mengambil data melalui mapper dan mengembalikannya berupa list.

```
@RequestMapping("/student/viewall")
public List<StudentModel> viewall (Model model) {
    List<StudentModel> students = studentService.selectAllStudents ();
    model.addAttribute ("students", students);

    return students;
}
```

Berikut hasil tampilan JSON pada browser dan JSON parser



The screenshot shows a web browser displaying a JSON response from a REST API. The URL is `localhost:8080/rest/student/viewall`. The response is a JSON array containing four student objects, each with `npm`, `name`, and `gpa` fields.

Below the browser window, a JSON parser interface is shown with two tabs: "String parse" and "JS eval". Both tabs display the same JSON structure, which is a list of four student objects.

```
[{"npm": "15097213162", "name": "Karrie", "gpa": 3.45}, {"npm": "16089412181", "name": "Johnson", "gpa": 3.42}, {"npm": "17061032423", "name": "Layla", "gpa": 3.63}, {"npm": "1706106785", "name": "Julio Muhammad Pranadano", "gpa": 3.65}]
```

2. Implementasikan service **consumer** untuk view all Students dengan melengkapi method **selectAllStudents** yang ada di kelas **StudentServiceRest** .

- Membuat SelectStudentAll() pada StudentServiceRest

```
@Override
public List<StudentModel> selectAllStudents() {
    // TODO Auto-generated method stub
    log.info("REST - select all students");
    return studentDAO.selectAllStudent();
}
```

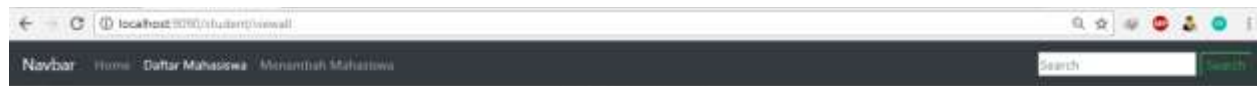
- Membuat method SelectAllStudent() pada class StudentDAOImpl

```
@Override
public List<StudentModel> selectAllStudent() {
    ResponseEntity<StudentModel[]> studentList = restTemplate.getForEntity("http://localhost:8080/rest/student/viewall", StudentModel[]);
    return Arrays.asList(studentList.getBody());
}
```

Membuat variable bertipe responseEntity dengan mengambil objek menggunakan restTemplate pada url yang digunakan untuk menampilkan data json pada project producer dan mengembalikannya dalam bentuk array.

Berikut hasil log dan tampilan dari StudentServiceRest

INFO 9312 --- [nio-9090-exec-4] c.e.t.service.StudentServiceRest : REST - select all students



## All Students

No	NPM	Name	GPA	Cumlaude		
1	15097213162	Kame	3.45	Sangat Memuaskan	Delete	Update
2	16089412181	Johnson	3.42	Sangat Memuaskan	Delete	Update
3	17061032423	Layla	3.63	Cumlaude	Delete	Update
4	1706106785	Julio Muhammad Pranadano	3.65	Cumlaude	Delete	Update

Mata Kuliah APAP