

Yang Saya dapatkan pada tutorial ini adalah saya belajar untuk menggunakan webservice untuk mengakses data data student dan juga course.

View student

```
localhost:8080/rest/student/view/124

{"npm":"124","name":"faiz","gpa":3.99,"courses":
[{"idCourse":null,"name":"PSP","credits":4,"students":null},
{"idCourse":null,"name":"SDA","credits":3,"students":null}]}
```

Hasilnya

String parse	JS eval
<pre>{ "npm": "124", "name": "faiz", "gpa": 3.99, "courses": [{ "idCourse": null, "name": "PSP", "credits": 4, "students": null }, { "idCourse": null, "name": "SDA", "credits": 3, "students": null }] }</pre>	<pre>{ "npm": "124", "name": "faiz", "gpa": 3.99, "courses": [{ "idCourse": null, "name": "PSP", "credits": 4, "students": null }, { "idCourse": null, "name": "SDA", "credits": 3, "students": null }] }</pre>

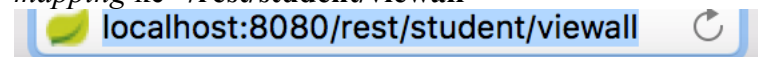
```
@RequestMapping("/student/view/{npm}")
public StudentModel view(@PathVariable(value="npm")String npm) {
    StudentModel student = studentService.selectStudent(npm);
    return student;
}
```

Karena return dari method tersebut adalah student maka method tersebut akan mengembalikan object dalam bentuk

Faiz Muhammad Refano
1506721895
APAP-A

json yang ada pada StudentModel yang terpilih dengan npm yang dituju.

Latihan 1: Buatlah *service* untuk mengembalikan seluruh student yang ada di basis data. *Service* ini mirip seperti *method* viewAll di Web Controller. *Service* tersebut di-*mapping* ke “/rest/student/viewall”



```
[{"npm": "124", "name": "faiz", "gpa": 3.99, "courses": [{"idCourse": null, "name": "PSP", "credits": 4, "students": null}, {"idCourse": null, "name": "SDA", "credits": 3, "students": null}]}, {"npm": "126", "name": "udin petot", "gpa": 3.77, "courses": [{"idCourse": null, "name": "PSP", "credits": 4, "students": null}]}]
```

Hasilnya

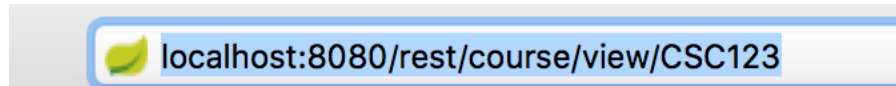
String parse	JS eval
<pre>[{ "npm": "124", "name": "faiz", "gpa": 3.99, "courses": [{ "idCourse": null, "name": "PSP", "credits": 4, "students": null }, { "idCourse": null, "name": "SDA", "credits": 3, "students": null }] }, { "npm": "126", "name": "udin petot", "gpa": 3.77, "courses": [{ "idCourse": null, "name": "PSP", "credits": 4, "students": null }] }]</pre>	<pre>[{ "npm": "124", "name": "faiz", "gpa": 3.99, "courses": [{ "idCourse": null, "name": "PSP", "credits": 4, "students": null }, { "idCourse": null, "name": "SDA", "credits": 3, "students": null }] }, { "npm": "126", "name": "udin petot", "gpa": 3.77, "courses": [{ "idCourse": null, "name": "PSP", "credits": 4, "students": null }] }]</pre>

```
@RequestMapping("/student/viewall")
public List<StudentModel> viewAll(Model model){
    List<StudentModel> kumpulanStudent = studentService.selectAllStudents();
    return kumpulanStudent;
}
```

Karena return dari method tersebut adalah kumpulanStudent yang berupa List maka method tersebut

akan mengembalikan object dalam bentuk json yang ada pada `List<StudentModel>` .

Latihan 2: Buatlah *service* untuk class `Course`. Buatlah *controller* baru yang terdapat *service* untuk melihat suatu course dengan masukan ID Course (view by ID) dan *service* untuk melihat semua course (view all).



```
{ "idCourse": "CSC123", "name": "PSP", "credits": 4, "students":  
  [ { "npm": "124", "name": "faiz", "gpa": 0.0, "courses": null }, { "npm": "126", "name": "udin  
petot", "gpa": 0.0, "courses": null } ] }
```

Hasilnya adalah

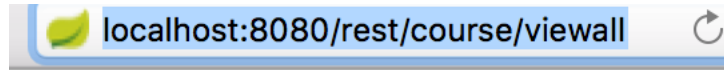
String parse	JS eval
<pre>{ "idCourse": "CSC123", "name": "PSP", "credits": 4, "students": [{ "npm": "124", "name": "faiz", "gpa": 0.0, "courses": null }, { "npm": "126", "name": "udin petot", "gpa": 0.0, "courses": null }] }</pre>	<pre>{ "idCourse": "CSC123", "name": "PSP", "credits": 4, "students": [{ "npm": "124", "name": "faiz", "gpa": 0, "courses": null }, { "npm": "126", "name": "udin petot", "gpa": 0, "courses": null }] }</pre>

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

Faiz Muhammad Refano
1506721895
APAP-A

Karena return dari method tersebut adalah course maka method tersebut akan mengembalikan object yang berupa json.

View All



```
[{"idCourse": "CSC123", "name": "PSP", "credits": 4, "students": null},  
{"idCourse": "CSC124", "name": "SDA", "credits": 3, "students": null},  
{"idCourse": "CSC125", "name": "DDP 1", "credits": 4, "students": null},  
{"idCourse": "CSC126", "name": "MPKT", "credits": 6, "students": null}]
```

Hasilnya adalah

String parse	JS eval
<pre>[{ "idCourse": "CSC123", "name": "PSP", "credits": 4, "students": null }, { "idCourse": "CSC124", "name": "SDA", "credits": 3, "students": null }, { "idCourse": "CSC125", "name": "DDP 1", "credits": 4, "students": null }, { "idCourse": "CSC126", "name": "MPKT", "credits": 6, "students": null }]</pre>	<pre>[{ "idCourse": "CSC123", "name": "PSP", "credits": 4, "students": null }, { "idCourse": "CSC124", "name": "SDA", "credits": 3, "students": null }, { "idCourse": "CSC125", "name": "DDP 1", "credits": 4, "students": null }, { "idCourse": "CSC126", "name": "MPKT", "credits": 6, "students": null }]</pre>

```
@RequestMapping("/course/viewall")
public List<CourseModel> viewAllCourse(Model model){
    List<CourseModel> kumpulanCourse = courseService.selectAllCourse();
    return kumpulanCourse;
}
```

Karena return dari method tersebut adalah kumpulanCourse maka method tersebut akan mengembalikan object yang berupa json yang berisi seluruh course yang ada didalam list tersebut.

Latihan 3: Implementasikan *service consumer* untuk view all Students dengan melengkapi *method* selectAllStudents yang ada di kelas **StudentServiceRest** .

Faiz Muhammad Refano
1506721895
APAP-A

localhost:9090/student/viewall

All Students

Navbar Home Daftar Mahasiswa Menambah Mahasiswa

Show 10 entries

No.	NPM	Nama	GPA	Status IPK	Delete	Update
No. 1	NPM = 124	Name = faiz	GPA = 3.99	CumLaude	Delete Data	Update Data
No. 2	NPM = 126	Name = udin petot	GPA = 3.77	CumLaude	Delete Data	Update Data

Showing 1 to 2 of 2 entries

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

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

localhost:9090/course/view/CSC123

ID = CSC123

Name = PSP

SKS = 4

Mahasiswa yang Mengambil

- 124-faiz
- 126-udin petot

Mata Kuliah APAP

Faiz Muhammad Refano
1506721895
APAP-A

```
    }

    @Override
    public CourseModel selectCourse (String id) {
        CourseModel course = restTemplate.getForObject
            ("http://localhost:8080/rest/course/view/"+id,
            CourseModel.class);


        return course;
    }

    @Override
    public List<CourseModel> selectAllCourse() {
        List<CourseModel> courses = restTemplate.getForObject("http://localhost:8080/rest/course/viewall", List.class);
        return courses;
    }
}
```

Method ini bertujuan untuk Memilih salah satu course yang di pilih berdasarkan id yang kita input.

Viewall

Faiz Muhammad Refano
1506721895
APAP-A

 localhost:9090/course/viewall

All Courses

Navbar

Home

Daftar Mahasiswa

Menambah Mahasiswa

Search

Search

No.	Id Course	Name	Credits
1	CSC123	PSP	4
2	CSC124	SDA	3
3	CSC125	DDP 1	4
4	CSC126	MPKT	6

Mata Kuliah APAP

Method ini bertujuan untuk menampilkan seluruh course yang ada pada database.

```
1 package com.example.dao;
2
3 import java.util.List;
4
5
6
7
8
9
10 public interface CourseDAO {
11     CourseModel selectCourse(String id);
12     List<CourseModel> selectAllCourse();
13 }
14
```