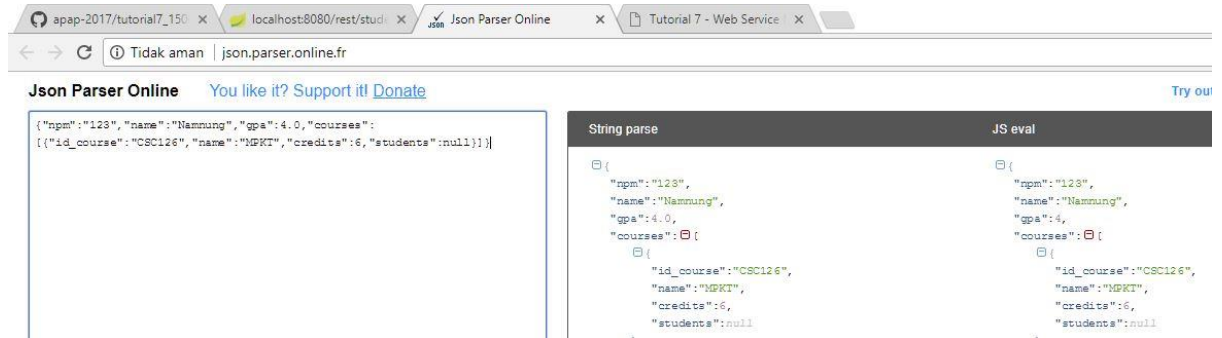


Ammar Shidqi
1506729185
APAP-A
Producer

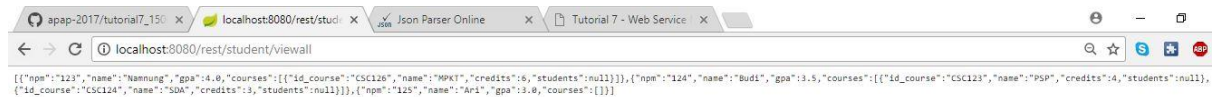
- localhost:8080/rest/student/view/123



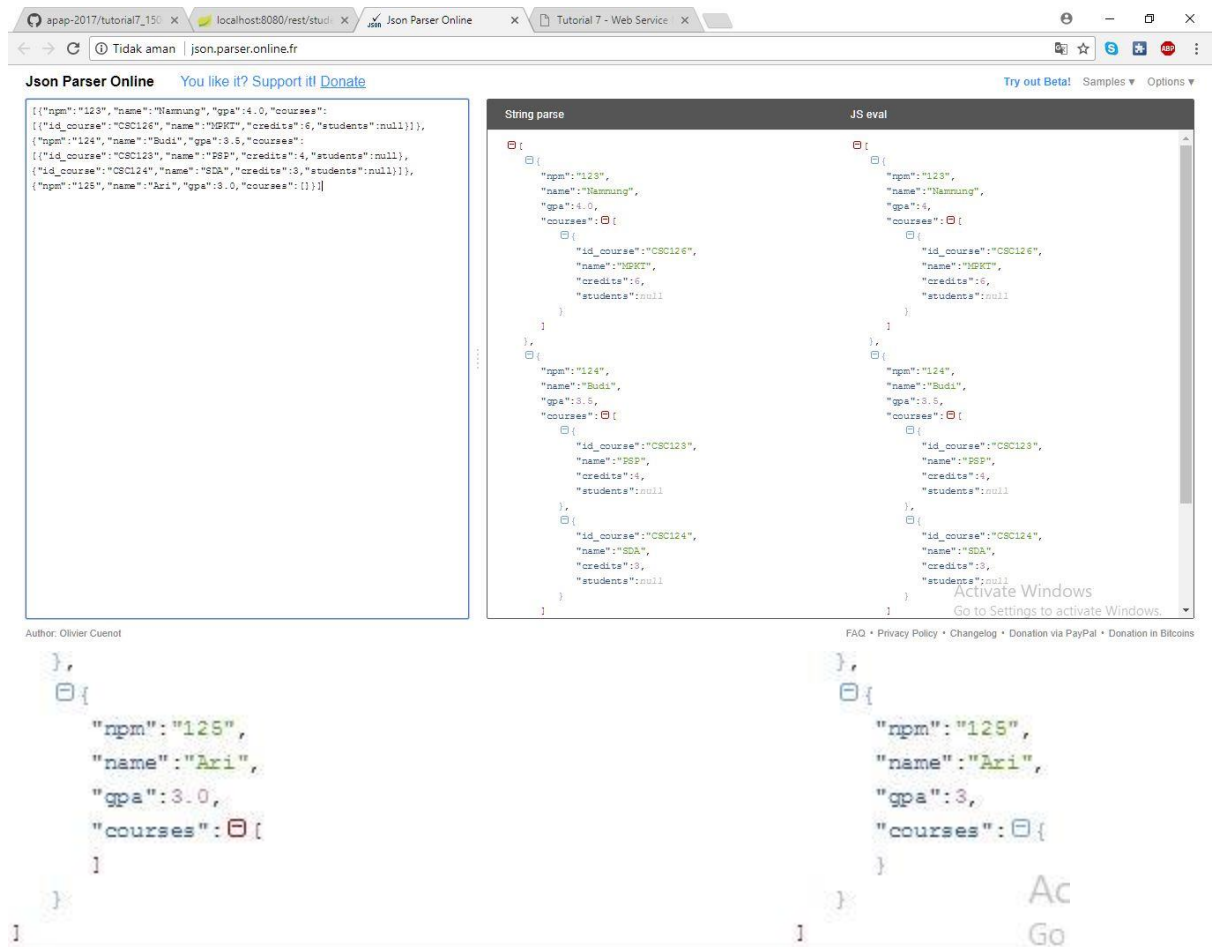
- Parse online



- Latihan 1: localhost:8080/rest/student/viewall



- Parse online



Code:

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

    return students;
}
```

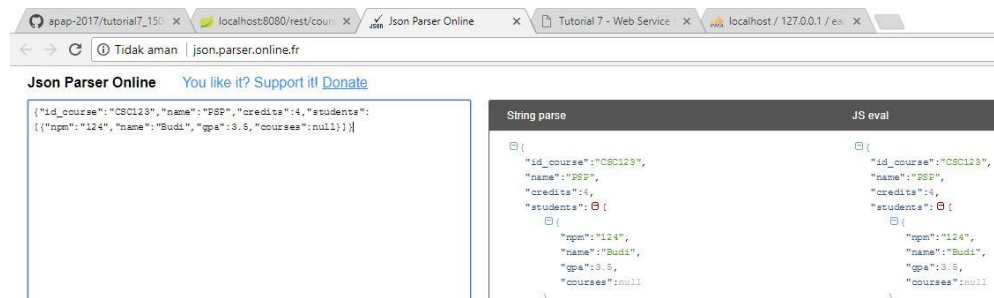
Penjelasan latihan 1:

Method tersebut agar mengembalikan List of StudentModel dalam bentuk objek dengan format JSON yang kemudian akan diparse agar kita dapat melihat outputnya (seperti API).

- Latihan 2: localhost:8080/rest/course/view/CSC123



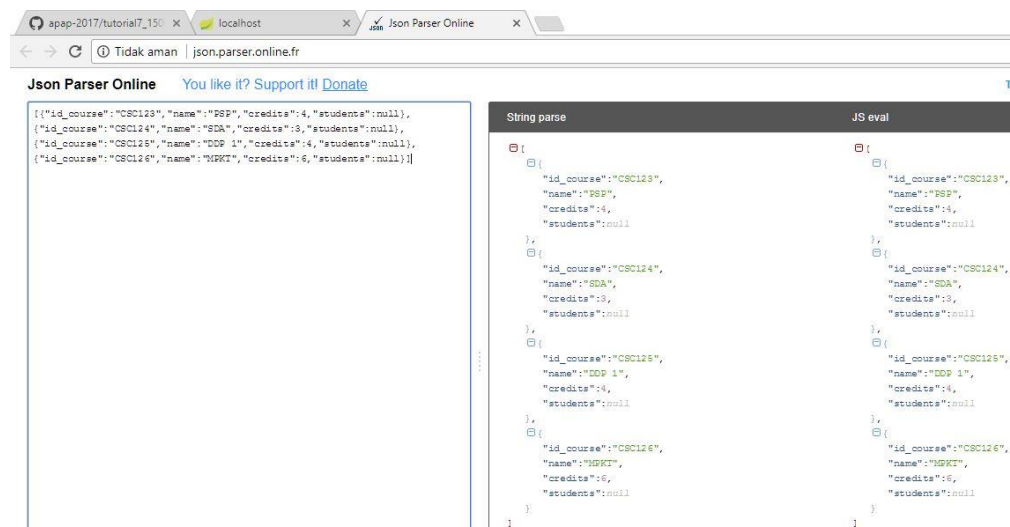
- Parse online



- Latihan 2: localhost:8080/rest/course/viewall



- Parse online



Ammar Shidqi
1506729185
APAP-A

Code:

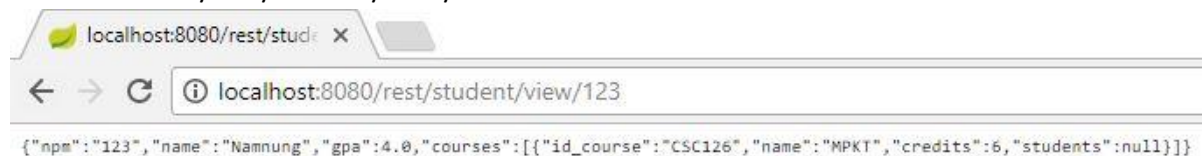
```
1 package com.example.rest;
2
3 import java.util.List;
4
14
15 @RestController
16 @RequestMapping("/rest")
17 public class CourseRestController {
18     @Autowired
19     CourseService courseService;
20
21     @RequestMapping("/course/view/{id_course}")
22     public CourseModel view (Model model,
23         @PathVariable(value = "id_course") String id_course)
24     {
25         CourseModel course = courseService.selectCourse (id_course);
26
27         return course;
28     }
29
30     @RequestMapping("course/viewall")
31     public List<CourseModel> viewall (Model model) {
32         List<CourseModel> courses = courseService.selectAllCourse();
33
34         return courses;
35     }
36 }
```

Penjelasan latihan 2:

Terdapat `@RequestMapping("/rest")` pada *class header* artinya semua *request mapping* pada kelas tersebut diawali dengan `"/rest"`. Kedua method tersebut agar mengembalikan `CourseModel` dan `List of CourseModel` dalam bentuk objek dengan format JSON yang kemudian akan diparse agar kita dapat melihat outputnya (seperti API).

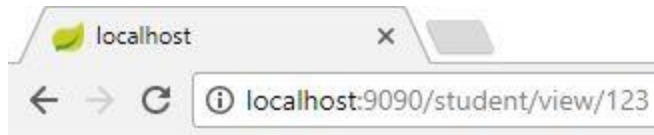
Consumer

- Uji coba running
 - localhost:8080/rest/student/view/123



Ammar Shidqi
1506729185
APAP-A

- localhost:9090/student/view/123



NPM = 123

Name = Namnung

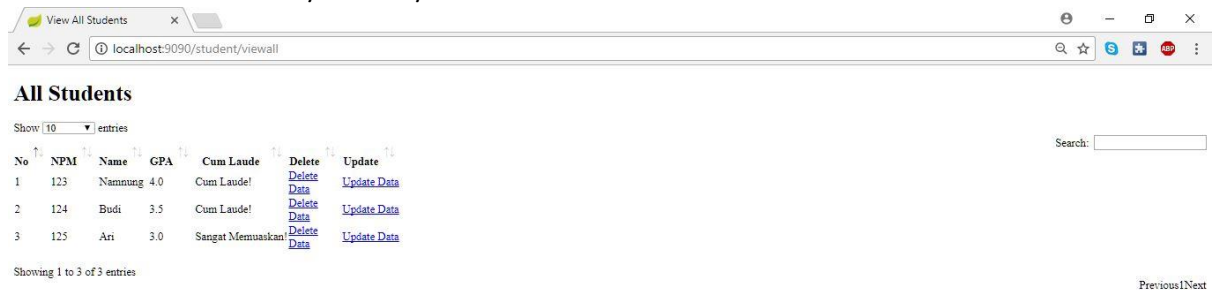
GPA = 4.0

Kuliah yang diambil

- MPKI-6 sks

2017-11-02 00:21:13.049 INFO 7444 --- [nio-9090-exec-1] com.example.service.StudentServiceRest : REST - select student with npm123

- Latihan 3: localhost:9090/student/viewall



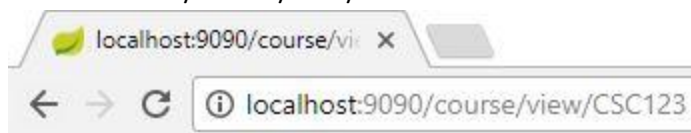
Code :

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

Penjelasan latihan 3:

Method tersebut diimplementasikan pada kelas StudentServiceRest, jadi kelas ini menggunakan *web service* bukan lagi dari *database*. Mungkin kelas ini fungsinya sama seperti service database dimana nanti method akan digunakan pada StudentController.

- Latihan 4:
 - localhost:9090/course/view/CSC123



ID = CSC123

Nama = PSP

SKS = 4

Mahasiswa yang mengambil

- 124-Budi

Ammar Shidqi
1506729185
APAP-A

- localhost:9090/course/viewall

No	ID	Nama	SKS
1	CSC123PSP		4
2	CSC124SDA		3
3	CSC125DDP 1		4
4	CSC126MPKT		6

Code:

Class CourseServiceRest

```
1 package com.example.service;
2
3 import java.util.List;
4
13
14 @Slf4j
15 @Service
16 @Primary
17 public class CourseServiceRest implements CourseService {
18     @Autowired
19     private CourseDAO courseDAO;
20
21     @Override
22     public CourseModel selectCourse(String id_course) {
23         Log.info("REST - select course with id_course{}", id_course);
24         return courseDAO.selectCourse(id_course);
25     }
26
27     @Override
28     public List<CourseModel> selectAllCourse() {
29         Log.info("REST - select all course");
30         return courseDAO.selectAllCourse();
31     }
32
33 }
```

Class CourseDAOImpl

```
1 package com.example.dao;
2
3 import java.util.List;
4
5 import org.springframework.beans.factory.annotation.Autowired;
6 import org.springframework.stereotype.Service;
7 import org.springframework.web.client.RestTemplate;
8
9 import com.example.model.CourseModel;
10
11 @Service
12 public class CourseDAOImpl implements CourseDAO {
13
14     @Autowired
15     private RestTemplate restTemplate;
16
17     @Override
18     public CourseModel selectCourse(String id_course) {
19         CourseModel course = restTemplate.getForObject("http://localhost:8080/rest/course/view/" + id_course, CourseModel.class);
20         return course;
21     }
22
23     @Override
24     public List<CourseModel> selectAllCourse() {
25         List<CourseModel> courses = restTemplate.getForObject("http://localhost:8080/rest/course/viewall", List.class);
26         return courses;
27     }
28 }
```


Interface CourseDAO

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

Penjelasan latihan 4:

Pada latihan 4 saya membuat 2 kelas baru dan 1 interface yaitu pada package DAO (Interface CourseDAO dan Class CourseDAOImpl yang mengimplements CourseDAO) dan pada package service (Class CourseServiceRest yang mengimplements CourseService). Pada interface CourseDAO saya mendefine method apa saja yang akan digunakan di CourseController dari CourseMapper, kemudian pada Class CourseDAOImpl akan dipanggil object url yang dibutuhkan, sementara CourseServiceRest adalah hal yang sama seperti CourseServiceDatabase (tetapi dalam hal ini menggunakan web service).