

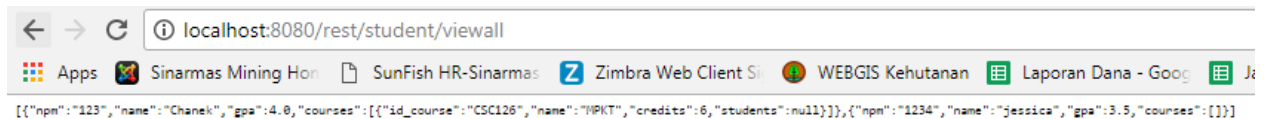
## Ringkasan

Pada tutorial ini, saya mempelajari cara menggunakan web service dengan melakukan pemisahan layer back end dan front end dari web. Front end merupakan aplikasi yang berinteraksi langsung dengan pengguna, sedangkan back end merupakan aplikasi pengolahan data dari database

## Latihan

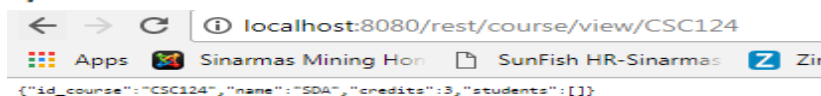
1. Pada method viewall() pada StudentRestController dilakukan mapping ke rest/student/viewall dan method akan memanggil method selectAllStudents() yang ada pada StudentService dan akan mengembalikan list dari object StudentModel dengan format JSON.

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



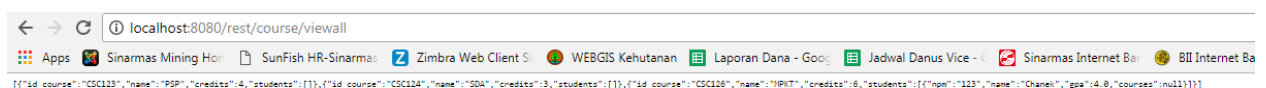
2. Membuat CourseRestController yang akan menggunakan REST web service untuk course. Untuk melihat suatu course, dilakukan mapping ke rest/course/view/id\_course. id\_course merupakan parameter untuk melihat course kemudian akan dipanggil method selectCourse dengan parameter id\_course yang diinginkan dari CourseService. Method ini akan mengembalikan object CourseModel berformat JSON.

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



Pada method viewall pada CourseRestController, akan dilakukan mapping ke "rest/course/viewall" dan method memanggil method selectAllCourses() pada CourseService dan akan mengembalikan list dari object CourseModel berformat JSON.

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



3. Method ini menggunakan `ResponseEntity<T>` yang akan menyimpan hasil yang diinginkan dari dalam bentuk JSON. Selain itu pada method ini digunakan juga `ParameterizedTypeReference` untuk melakukan passing data dengan tipe generic. Kemudian akan menggunakan `getBody()` untuk mengambil data dari JSON yang dihasilkan pada response yang dihasilkan.

```
@Override
public List<StudentModel> selectAllStudents ()
{
    ResponseEntity<List<StudentModel>> response = restTemplate.exchange("http://localhost:8080/rest/student/viewall",
        HttpMethod.GET, null, new ParameterizedTypeReference<List<StudentModel>>() {});
    List<StudentModel> students = response.getBody();

    return students;
}
```

4. Method view course menggunakan `getForObject` untuk mendapatkan data sebuah object dari url dengan tipe class yang telah ditentukan. Setelah membuat method pada `CourseDAOImpl`, panggil method `selectCourse` dengan parameter `id_course` dari `CourseDAO` pada `CourseServiceRest` untuk menjalankan `CourseService`

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

Method view all course menggunakan `ResponseEntity<T>` yang akan menyimpan hasil yang diinginkan dalam bentuk JSON. Pada method ini digunakan juga `ParameterizedTypeReference` untuk melakukan passing data dengan tipe generic. Selanjutnya akan menggunakan `getBody()` untuk mengambil data dari JSON yang dihasilkan pada response yang dihasilkan. Setelah membuat method pada `CourseDAOImpl`, panggil method `dselectAllCourses` dengan parameter `id_course` dari `CourseDAO` pada `CourseServiceRest` untuk menjalankan `CourseService`

```
@Override
public List<CourseModel> selectAllCourses ()
{
    ResponseEntity<List<CourseModel>> response = restTemplate.exchange("http://localhost:8080/rest/course/viewall",
        HttpMethod.GET, null, new ParameterizedTypeReference<List<CourseModel>>() {});
    List<CourseModel> courses = response.getBody();

    return courses;
}
```



## All Courses

No. 1

ID = CSC123

Name = PSP

Credit = 4

Mahasiswa yang mengambil

---

No. 2

ID = CSC124

Name = SDA

Credit = 3

Mahasiswa yang mengambil

---

No. 3

ID = CSC126

Name = MPKT

Credit = 6

Mahasiswa yang mengambil

- 123 - Chanek