

Nama: Muhammad Hasby Rosyadi
NPM: 1306386642
APAP – A

Tutorial07Producer

1. Buatlah *service* untuk mengembalikan seluruh student yang ada di basis data. *Service* ini miripseperti *method* `viewAll` di Web Controller. *Service* tersebut di- *mapping* ke `"/rest/student/viewall"`.

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

pada code diatas akan menampilkan list isi dari StudentModel yang disimpan pada students. Saat di jalankan maka akan menampilkan view seperti dibawah.



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).

• View by ID

```
@RequestMapping("/course/view/{id_course}")
public CourseModel viewCourse (Model model,
    @PathVariable(value = "id_course") String id_course)
{
    CourseModel course = studentService.selectCourse(id_course);
    return course;
}
```

pada code diatas akan menampilkan detail dari course yang pilih berdasarkan idCourse tersebut. saat dijalankan akan menampilkan view seperti dibawah



• View All Course

```
@Select("select id_course, name, credits from course")
@Results(value = {
    @Result(property="idCourse", column="id_course"),
    @Result(property="students", column="id_course", javaType=List.class,
        many=@Many(select="selectCourseByStudent"))
})
List<CourseModel> selectAllCourse();
```

pada code diatas merupakan mapper untuk menampilkan setiap course dan siapa saja mahasiswa yang mengambil course tersebut.

Nama: Muhammad Hasby Rosyadi
NPM: 1306386642
APAP – A

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

Pada code diatas memanggil mapper selectAllCourse yang dibuat tadi. Kode diatas akan menampilkan seluruh course dan mahasiswa yang mengambil.
Saat dijalankan akan menampilkan view seperti dibawah.



Tutorial07Customer

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

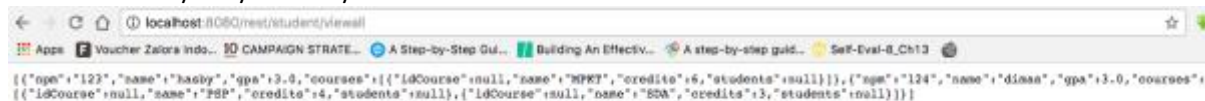
```
@Override
public List<StudentModel> selectAllStudents() {
    ResponseEntity<StudentModel> responseEntity = restTemplate.getForEntity("http://localhost:8080/rest/student/viewall",
    return Arrays.asList(responseEntity.getBody());
}
```

menambahkan code diatas pada StudentDAOImpl untuk mendapatkan list seluruh StudentModel dari url rest/student/viewall. Menggunakan ResponseEntity untuk objek List.

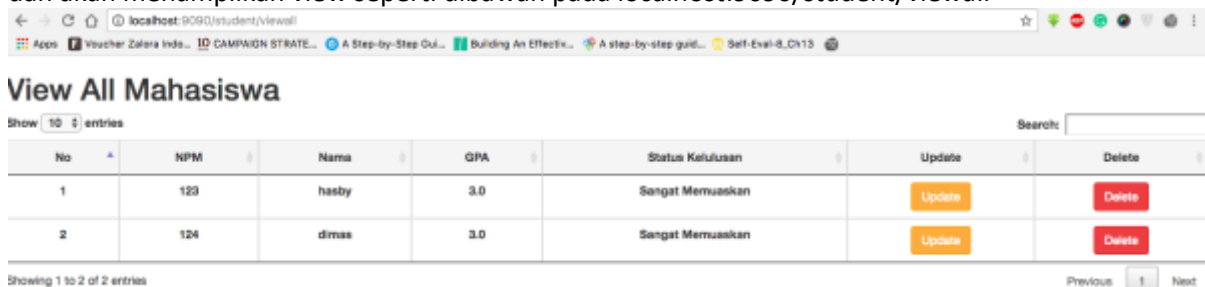
```
@Override
public List<StudentModel> selectAllStudents() {
    log.info("REST - select all students" );
    return studentDAO.selectAllStudents();
}
```

Selanjutnya tambahkan code diatas pada StudentServiceRest agar dapat mengambil data-data dari StudentDAOImpl.

Ketika dijalankan maka akan menampilkan view seperti dibawah pada localhost:8080/rest/student/viewall



dan akan menampilkan view seperti dibawah pada localhost:9090/student/viewall



Nama: Muhammad Hasby Rosyadi
NPM: 1306386642
APAP – A

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

• View Course by Id

1. membuat interface CourseDAO pada example.dao
2. membuat class CourseDAOImpl pada example.service yang meng implements CourseDAO. Kemudian isi dengan kode dibawah ini.

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

3. membuat interface CourseService

```
public interface CourseService {
    CourseModel selectCourse (String id_course);

    List<CourseModel> selectAllCourse();
}
```

dan CourseServiceRest pada example.service

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

4. kemudian jalankan. Maka akan menampilkan view seperti dibawah pada localhost:8080/rest/course/view/csc123



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

dan akan menampilkan view seperi dibawah pada localhost:9090/course/view/csc123



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

View Student by Course

ID = CSC123

Name = PSP

SKS = 4

Mahasiswa yang Mengambil

- 124-dimas

Nama: Muhammad Hasby Rosyadi
NPM: 1306386642
APAP – A

• View All Course

1. membuat interface CourseDAO pada example.dao
2. membuat class CourseDAOImpl pada example.service yang meng implements CourseDAO. Kemudian isi dengan kode dibawah ini.

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

3. membuat interface CourseService

```
public interface CourseService {
    CourseModel selectCourse (String id_course);

    List<CourseModel> selectAllCourse();
}
```

dan CourseServiceRest pada example.service

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

4. kemudian jalankan. Maka akan menampilkan view seperti dibawah pada localhost:8080/rest/course/viewall



```
[{"idCourse": "CSC123", "name": "PSP", "credits": 4, "students": [{"npe": "124", "name": "dimas", "gpa": 8.9, "courses": "all"}], {"idCourse": "CSC124", "name": "SDA", "credits": 3, "students": [{"npe": "124", "name": "dimas", "gpa": 6.8, "courses": "all"}], {"idCourse": "CSC125", "name": "DDP 1", "credits": 4, "students": [{"idCourse": "CSC126", "name": "MPRT", "credits": 4, "students": [{"npe": "123", "name": "kobby", "gpa": 6.9, "courses": "all"}]}]}
```

dan akan menampilkan view seperi dibawah pada localhost:9090/course/viewall



All Students

No. 1

ID Course = CSC123

Name = PSP

SKS = 4

Mahasiswa yang mengambil

- dimas-124

No. 2

ID Course = CSC124

Name = SDA

SKS = 3

Mahasiswa yang mengambil

- dimas-124

No. 3

ID Course = CSC125

Name = DDP 1

SKS = 4

Mahasiswa yang mengambil