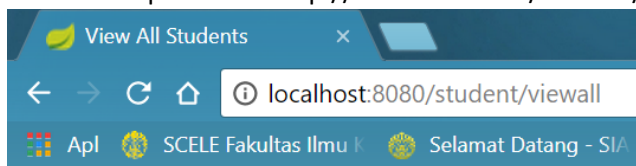


1. Ubah method `selectAllStudents` pada kelas `StudentMapper` agar halaman `viewall` menampilkan semua student beserta daftar kuliah yang diambil.

```
@Select("select npm, name, gpa from student")
@Results(value = {
    @Result(property="npm", column="npm"), // col
    @Result(property="name", column="name"),
    @Result(property="gpa", column="gpa"),
    @Result(property="courses", column="npm",
        javaType = List.class,
        many=@Many(select="selectCourses"))
})
List<StudentModel> selectAllStudents ();
```

Berikut tampilan hasil `http://localhost:8080/student/viewall`



All Students

No. 1

NPM = 123

Name = anjani dwi putri

GPA = 3.99

Kuliah yang diambil

- MPKT-6 sks

[Delete](#)

[Update](#)

No. 2

NPM = 124

Name = atikah

GPA = 4.0

Kuliah yang diambil

- PSP-4 sks
- SDA-3 sks

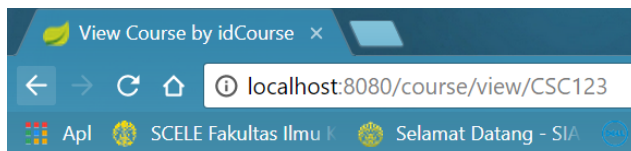
[Delete](#)

[Update](#)

2. **Buatlah view pada halaman `http://localhost:8080/course/view/{id}` untuk Course sehingga dapat menampilkan data course beserta Student yang mengambil.**

```
@RequestMapping("/course/view/{id}")
public String viewID (Model model, @PathVariable(value
    CourseModel course = studentDAO.viewID(id_course);

    if (course != null) {
        model.addAttribute ("course", course);
        return "view-id";
    } else {
        model.addAttribute ("id_course", id_course);
        return "not-found-id";
    }
}
```



ID = CSC123

Name = PSP

SKS = 4

Mahasiswa yang mengambil

- 124-atikah

3. **Jelaskan apa saja hal yang Anda pelajari dari tutorial ini.**

Pada tutorial ini sama seperti tutorial 4, namun pada tutorial ini di tambahkan bagaimana cara untuk mengambil data dari database untuk di tampilkan dengan relasi – relasi yang terbentuk. Seperti menampilkan daftar mata kuliah yang diambil oleh mahasiswa tertentu. Hal tersebut menggunakan 3 table dalam database.

4. **Cantumkan juga penjelasan Anda terhadap hal-hal berikut:**

- Method yang Anda ubah pada Latihan Merubah SelectAllStudents, jelaskan

Pada method ini saya menambahkan beberapa @Result yang bertujuan pada variable property diisi dengan nama dari class StudentModel, sedangkan variable column diisi dengan nama kolom hasil query dari database sesuai parameter yang dibutuhkan. Property diset dengan variable courses karena untuk mengisi variable courses di class StudentModel.

- Method yang Anda buat pada Latihan Menambahkan View pada Course, jelaskan

Pada method tersebut saya membuat dimana parameter yang dibutuhkan ada id_course, jika course yang diminta tidak ada pada database maka tidak akan menampilkan apapun untuk mata kuliahnya dan mahasiswa yang mengambil, sedangkan jika ada akan menampilkan. Sebelum dapat menampilkan pada html, saya membuat method selectCursesStudent pada studentMapper untuk mengisi data dari database ke variable property, lalu menambahkan interface baru CourseModel serta menambahkan interface pada StudentService juga.