

Di tutorial kali ini saya mempelajari tentang bagaimana menggunakan relasi database dengan Spring. Latihan yang diberikan ialah menampilkan daftar course yang diambil tiap student dan menampilkan daftar student yang mengambil setiap course.

Pada course yang diambil setiap student, saya menambahkan anotasi results yang sama dengan yang ada di tutorial ke method selectAllStudents()

```
@Select("select npm, name, gpa from student")
@Results(value = {
    @Result(property="npm", column="npm"),
    @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 ();
```

Setelah itu saya menambahkan list di laman viewall.html untuk menampilkan nama course yang diambilnya.

```
<div th:each="student, iterationStatus: ${students}">
    <h3 th:text="'No. ' + ${iterationStatus.count}">No. 1</h3>
    <h3 th:text="'NPM = ' + ${student.npm}">Student NPM</h3>
    <h3 th:text="'Name = ' + ${student.name}">Student Name</h3>
    <h3 th:text="'GPA = ' + ${student.gpa}">Student GPA</h3>
    <h3>Kuliah yang diambil</h3>
    <ul th:each="course, iterationStatus: ${student.courses}">
        <li th:text="${course.name} + ' - ' + ${course.credits} + ' sks'"
            Nama Kuliah-X sks
        </li>
    </ul>

    <a th:href="'/student/delete/' + ${student.npm}">Delete Data</a><br/>
    <a th:href="'/student/update/' + ${student.npm}">Update Data</a><br/>
    <hr/>
</div>
```

Pada latihan kedua, saya membuat method selectCourse(id\_course) pada Mapper dengan anotasi result yang disesuaikan (dibalik dengan yang sebelumnya). Setelah itu saya membuat method selectStudents(id\_course) dengan anotasi select yang mengabungkan tabel studentcourse dengan tabel student

```



```

Kemudian saya membuat mapping baru di controller

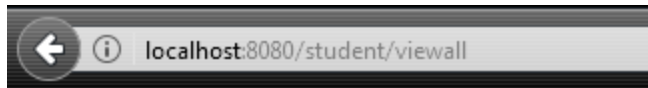
```

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

    if(course !=null) {
        model.addAttribute("course", course);
        return "view-course";
    } else {
        model.addAttribute("id_course", id_course);
        return "not-found2";
    }
}

```

Setelah itu hasil sudah sesuai dengan requirement yang diminta



**NPM = 12346**

**Name = rizurinrinrin**

**GPA = 2.1**

**Kuliah yang diambil**

- MPKT-6 sks

[Delete Data](#)

[Update Data](#)

---

**No. 2**

**NPM = 98123**

**Name = oaskdosd**

**GPA = 3.54**

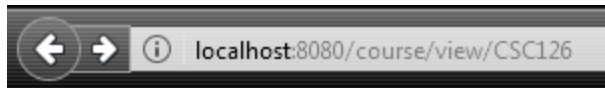
**Kuliah yang diambil**

- PSP-4 sks
- SDA-3 sks
- MPKT-6 sks

[Delete Data](#)

[Update Data](#)

---



**ID = CSC126**

**Nama = MPKT**

**SKS = 6**

**Mahasiswa yang mengambil**

- 12346-rizurinrinrin
- 98123-oaskdosd