

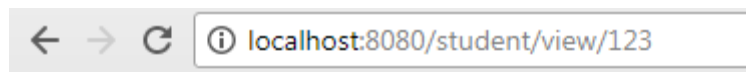
**Nama** : Adil Krisnadi Pradana  
**NPM** : 1406623631  
**Kelas** : APAP-C

## TUTORIAL 5

### Lesson Learned

Tutorial 5 kali ini merupakan lanjutan dari tutorial 4 sebelumnya, yakni masih menggunakan MyBatis dan Lombok sebagai dua library eksternal. Pada tutorial 5 kali ini, kita melengkapi database yang ada dengan menambahkan tabel Course, dan menghubungkannya dengan tabel Student yang telah ada sebelumnya di mana hubungan antara Student dan Course bersifat Many-to-Many karena student dapat mengambil beberapa course dan course dapat diambil oleh beberapa student. Adapun hal lain yang dapat dipelajari dari tutorial 5 kali ini adalah penggunaan anotasi @Results, @Result, dan @Many.

Jalankan program dan buka <http://localhost:8080/student/view/123> dan <http://localhost:8080/student/view/124>



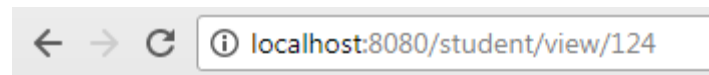
**NPM = 123**

**Name = Chanek**

**GPA = 4.0**

**Kuliah yang diambil**

- MPKT-6 sks



**NPM = 124**

**Name = Chanek Jr.**

**GPA = 3.0**

**Kuliah yang diambil**

- PSP-4 sks
- SDA-3 sks

### LATIHAN

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

#### All Students

No. 1

NPM = 1

Name = Adil

GPA = 3.5

[Delete Data](#)  
[Update Data](#)

---

No. 2

NPM = 123

Name = Chanek

GPA = 4.0

- MPKT-6 sks

[Delete Data](#)  
[Update Data](#)

---

No. 3

NPM = 124

Name = Chanek Jr.

GPA = 3.0

- PSP-4 sks
- SDA-3 sks

[Delete Data](#)  
[Update Data](#)

---

No. 4

NPM = 2

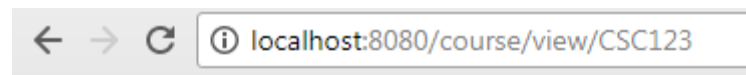
Name = Aminur

GPA = 3.8

[Delete Data](#)  
[Update Data](#)

---

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



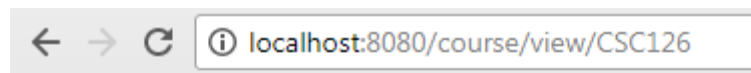
**ID = CSC123**

**Nama = PSP**

**SKS = 4**

**Mahasiswa yang mengambil kuliah ini**

- 124-Chanek Jr.



**ID = CSC126**

**Nama = MPKT**

**SKS = 6**

**Mahasiswa yang mengambil kuliah ini**

- 123-Chanek

## Merubah SelectAllStudents

```
1 package com.example.dao;
2
3 import java.util.List;
4
17 @Mapper
18 public interface StudentMapper
19 {
20
21
22 @Select("select npm, name, gpa from student where npm = #{npm}")
23 @Results(value = {
24     @Result(property="npm", column="npm"),
25     @Result(property="name", column="name"),
26     @Result(property="gpa", column="gpa"),
27     @Result(property="courses", column="npm",
28         javaType = List.class,
29         many=@Many(select="selectCourses"))
30 })
31 StudentModel selectStudent (@Param("npm") String npm);
32
33 @Select("select course.id_course, nama, sks " +
34     "from studentcourse join course " +
35     "on studentcourse.id_course = course.id_course " +
36     "where studentcourse.npm = #{npm}")
37 List<CourseModel> selectCourses (@Param("npm") String npm);
38
39 @Select("select npm, name, gpa from student ")
40 @Results(value = {
41     @Result(property="npm", column="npm"),
42     @Result(property="name", column="name"),
43     @Result(property="gpa", column="gpa"),
44     @Result(property = "courses", column = "npm", javaType = List.class, many = @Many(select = "selectCourses"))
45 })
46 List<StudentModel> selectAllStudents ();
47
```

Anotasi yang digunakan pada SelectAllStudents adalah @Results dan @Result untuk melakukan mapping dari database ke List<StudentModel>, serta anotasi @Many untuk menyimpan course yang diambil oleh setiap student. Anotasi @Many sendiri digunakan karena hubungan antara tabel Student dengan tabel studentcourse adalah One-to-Many, dan nantinya hasil query dari method selectCourses akan disimpan ke dalam model StudentModel dalam bentuk List<CourseModel>

## Menambahkan View pada Course

```
1 package com.example.dao;
2
3 import java.util.List;
4
14 @Mapper
15 public interface CourseMapper {
16
17 @Select("select id_course, nama, sks from course where id_course = #{id_course}")
18 @Results(value = {
19     @Result(property = "id_course", column = "id_course"),
20     @Result(property = "nama", column = "nama"), @Result(property = "sks", column = "sks"),
21     @Result(property = "students", column = "id_course", javaType = List.class, many = @Many(select = "selectStudents")) })
22 CourseModel selectCourse(@Param("id_course") String id_course);
23
24 @Select("select student.npm, student.name from studentcourse join student on studentcourse.npm = student.npm "
25     + "where studentcourse.id_course = #{id_course}")
26 List<StudentModel> selectStudents(@Param("id_course") String id_course);
27 }
28
```

Pertama kita membuat variable List<StudentModel> students di dalam model CourseModel. Lalu anotasi @Results dan @Result digunakan untuk melakukan mapping dari database ke objek CourseModel. Untuk mendapatkan list student yang mengambil suatu course kita menggunakan method mapper selectStudents yang nantinya akan disimpan ke dalam objek CourseModel. Anotasi @Many sendiri digunakan karena hubungan antara tabel Student dengan tabel studentcourse adalah One-To-Many.