

Nama : Radhitya Rahman

Kelas : 1406623335

Hal yang telah saya pelajari dari tutorial kali ini adalah pemanggilan objek (atau banyak objek) dari sebuah objek. Jika dilihat dari kodingan saya adalah pada viewCourse.html terdapat `${course.students}` di mana course merupakan sebuah objek dan students adalah banyak objek yang telah disimpan di database.

1. Pada method `selectAllStudents` saya hanya membuatnya tepat sama seperti method `selectStudent` yang dibuat pada latihan sebelumnya dengan hanya merubah pemanggilan query yang sebelumnya `"select npm, name, gpa from student where npm = #{npm}"` menjadi `"select npm, name, gpa from student"` di mana perbedaan keduanya terletak pada dihilangkannya decision 'memiliki npm' tertentu atau tidak.
2. Pada class `StudentController` saya menambahkan method `viewCourse` di mana isinya sama seperti method `view` dengan RequestMapping-nya `"/student/view"` dengan mengganti variable `StudentModel` dengan `CourseModel` di mana variable ini memiliki value instance dari interface `StudentService` dengan method `selectCourse()` yang dijalankan pada class `StudentServiceDatabase`. Method `selectCourse()` pada class `StudentServiceDatabase` me-return object `CourseModel` dari method `selectCourseStudent()` pada class `StudentMapper`. Pada method `selectCourseStudent()` terdapat parameter yang berasal dari requestMapping berupa string `id` di mana `id` tersebut menjadi variable untuk memilih `id_course`, `name`, `credits` di table `course` di database. `id_course`, `name`, `credits` ini kemudian dimasukkan ke dalam property yang terhubung ke attribute di `CourseModel` di mana semua attribute ini diisikan value yang sesuai dengan value property-nya yang telah diambil dari database sebelumnya. Kemudian untuk property `"students"` diisi dengan objek `List<StudentModel>` yang dipanggil dari `@Many(select=selectStudentCourse)` dengan parameter `id_course` yang diambil dari column `id_course` yang telah diambil dari database. Kemudian method `selectStudentCourse` akan menjalankan statement `@Select` yang menggabungkan(join) table `student` dan `studentcourse` dengan searching data menggunakan parameter `id_course`. Akhirnya property `"courses"` dengan tipe data `List<StudentModel>` yang berada di `CourseModel` diisi dengan hasil pemanggilan `@Select` sebelumnya. Property `"students"` sekarang memiliki data `"name"` dan `"student.npm"`.