

1. Method Delete yang dibuat adalah sebagai berikut

```
@Delete("delete from student where npm = #{npm}")
void deleteStudent(String npm);
```

Method ini ditambahkan pada class student mapper. Delete akan dilakukan jika memiliki kondisi npm sama sesuai dengan npm yang dipilih pada viewall.html

Kemudian menambahkan log pada class student service database

```
@Override
public void updateStudent (StudentModel student)
{
    log.info ("student " + student + "updated ");
    studentMapper.updateStudent (student);
}
```

Baru setelah itu menambahkan kondisi untuk menghandle npm not found

```
@RequestMapping("/student/update/{npm}")
public String update (Model model, @PathVariable(value = "npm") String npm)
{
    StudentModel student = studentDAO.selectStudent (npm);

    if (student != null) {
        model.addAttribute ("student", student);
        return "form-update";
    } else {
        model.addAttribute ("npm", npm);
        return "not-found";
    }
}
```

2. Method Update yang dibuat adalah sebagai berikut

```
@Update("update student SET name =#{name},gpa =#{gpa} where npm = #{npm}")
void updateStudent(StudentModel student);
```

Update akan mengganti nama gpa pada kondisi yang sama

Kemudian menambahkan log pada class student service database

```
@Override
public void updateStudent (StudentModel student)
{
    log.info ("student " + student + "updated ");
    studentMapper.updateStudent (student);
}
```

3. Untuk Object parameter hanya mengganti method update pada student controller sebagai berikut

```
@RequestMapping(value = "/student/update/submit", method = RequestMethod.POST)
public String updateSubmit (@ModelAttribute StudentModel student)
{
    studentDAO.updateStudent (student);
    return "success-update";
}
```

Dan mengganti form update menjadi seperti ini

```
<form action="/student/update/submit" method="post" th:object="${student}">
    <div>
        <label for="npm">NPM</label> <input type="text" name="npm" readonly="true" th:value="${student.npm}" th:field="*{npm}" />
    </div>
    <div>
        <label for="name">Name</label> <input type="text" name="name" th:value="${student.name}" th:field="*{name}" />
    </div>
    <div>
        <label for="gpa">GPA</label> <input type="text" name="gpa" th:value="${student.gpa}" th:field="*{gpa}" />
    </div>
    <div>
        <button type="submit" name="action" value="save">Update</button>
    </div>
</form>
```

Pertanyaan

- 1.
2. Ada perbedaan antara post method dengan get method.. post method menempatkan data yang dikirim pada proses submit di bagian body. Sementara get method menempatkannya di bagian header. Ya, ada penanganan berbeda di controller dalam mengambil data data tersebut.. untuk penanganan di controller sepertinya sama saja.
3. Ya post dan get bisa dilakukan bersamaan