

**TUGAS TUTORIAL 3**  
**ARSITEKTUR DAN PEMROGRAMAN APLIKASI**  
**ENTERPRISE**



Disusun oleh :

ANDESTA PUTRA  
NPM. 1706106633

**PROGRAM SARJANA 1 EKSTENSI SISTEM INFORMASI**  
**FAKULTAS ILMU KOMPUTER**  
**UNIVERSITAS INDONESIA**  
**2018**

## A. Ringkasan Materi

Tutorial ini berfokus pada materi komponen model dan service pada penggunaan Spring Boot. Model merupakan sebuah objek yang bertugas untuk menyimpan informasi, sebagai contoh model mahasiswa adalah sesuatu yang berisi informasi mengenai mahasiswa seperti nama, npm dan lainnya.

Service merupakan jembatan antara controller dan database untuk saling berkomunikasi, service bertugas mengolah data yang terdapat pada database dengan menggunakan bussines logic.

## B. Latihan Soal

### Membuat Controller dan Fungsi Add

localhost:8080/student/add?npm=12345&name=chanek&gpa=3.43

**Pertanyaan 1:** apakah hasilnya? Jika error, tuliskan penjelasan Anda

**Jawab :**

Hasil dari eksekusi program dari url tersebut menampilkan halaman add.html dan tidak terdapat error. Hal ini dikarenakan semua parameter yang ada pada controller method add yaitu npm, nama dan gpa telah terpenuhi.



localhost:8080/student/add?npm=12345&name=chanek

**Pertanyaan 2:** apakah hasilnya? Jika error, tuliskan penjelasan Anda.

**Jawab :**

Terjadi error karena parameter yang disertakan untuk dapat memanggil method add bersifat required sehingga wajib untuk diisi.

## Method View By NPM

localhost:8080/student/add?npm=12345&name=chanek&gpa=3.43 lalu buka  
localhost:8080/student/view?npm=12345

**Pertanyaan 3:** apakah data Student tersebut muncul? Jika tidak, mengapa?

Data muncul dikarenakan sudah melakukan add student pada url sebelumnya.



Restart program kemudian jalankan localhost:8080/student/view?npm=12345

**Pertanyaan 4:** apakah data Student tersebut muncul? Jika tidak, mengapa?

Program tidak menampilkan data yang diinginkan karena ketika kita melakukan restart program maka data akan direset juga.



## Method View All

localhost:8080/student/add?npm=12345&name=chanek&gpa=3.43 lalu buka  
localhost:8080/student/viewall,

**Pertanyaan 5:** apakah data Student tersebut muncul?

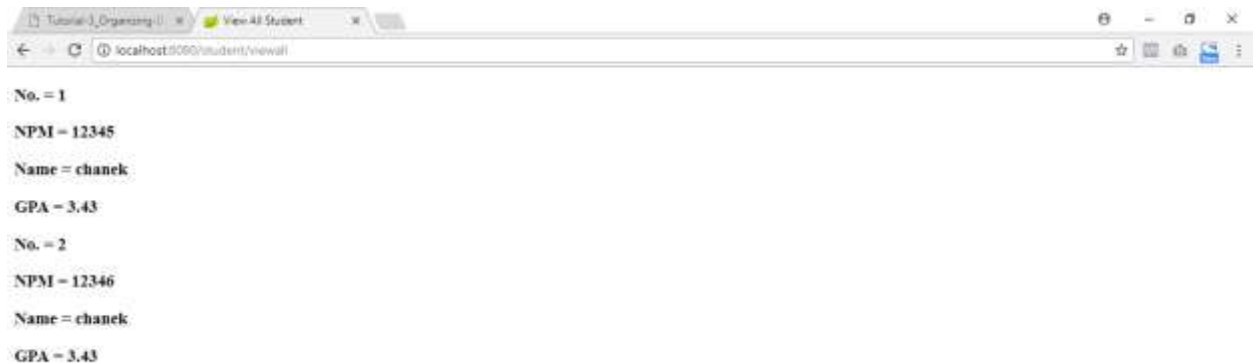
Data tersebut muncul.



Coba tambahkan data Student lainnya dengan NPM yang berbeda, lalu buka localhost:8080/student/viewall,

**Pertanyaan 6:** Apakah semua data Student muncul?

Semua data student berhasil dimunculkan.



### C. Method selectStudent

```
@Override
public StudentModel selectStudent(String npm) {
    for(StudentModel student :studentList){
        if(student.getNpm().equals(npm)) return student;
    }
    return null;
}
```

Method ini menerima parameter npm kemudian melakukan pengecekan dengan menggunakan forloop dan pengkondisian yaitu apabila npm mahasiswa sama dengan npm yang diinputkan maka program akan melakukan return objek mahasiswa tersebut dan apabila tidak maka program melakukan return null.

### D. Fitur Delete

Untuk pembuatan fitur delete hal yang pertama dilakukan adalah pembuatan mapping serta method deletePath pada StudentController sebagai berikut.

```
@RequestMapping("/student/delete/{npm}")
public String deletePath(@PathVariable Optional<String> npm, Model model){
    if (npm.isPresent()){

        StudentModel student = studentService.selectStudent(npm.get());

        if(student==null)
            return "errorNPM";
    }
}
```

```

        studentService.deleteStudent(student);
        return "delete";
    }
    return "errorNPM";
}

```

Pada source diatas ketika url localhost:8080/student/delete/{npm} maka method deletePath akan dieksekusi. Kemudian didalam method akan melakukan pengecekan apabila npm diinputkan pada url maka akan dilakukan pembuatan objek student berdasarkan data npm yang diinputkan. Setelah itu kembali dilakukan pengecekan apakah data student yang dicari terdapat pada database atau tidak, jika iya maka akan dilakukan eksekusi dari service deleteStudent yaitu untuk menghapus objek student yang dipilih, apabila tidak terdapat student berdasarkan npm tersebut maka akan ditampilkan view errorNPM.html yang berisi notifikasi bahwa npm kosong atau tidak ditemukan.

Kita juga perlu menambahkan method pada interface studentService dan class InMemoryStudentService yang bertugas untuk menghapus data pada database.

Source pada studentService

```

void deleteStudent(StudentModel student);

```

Source pada InMemoryStudentService

```

@Override
public void deleteStudent(StudentModel student) {
    studentList.remove(student); //menghapus list berdasarkan objek yang
dituju
}

```

