

Hal yang telah dipelajari pada tutorial 7 :

- Membuat layer backend (Service Producer) dan Frontend (Service Consumer) secara terpisah.
- Service Producer menyediakan web service yang dapat digunakan oleh Service Consumer.

Membuat Service Producer

1. Diimport tutorial 4 untuk menggunakan fungsi dan kelas yang diperlukan untuk membuat Service Producer ini.
2. Dibuat package baru yaitu com.fpamungkas.tutorial07producer.rest
3. Dibuat class baru yaitu StudentRestController.java pada package yang baru saja dibuat.

Pada class header untuk controller menggunakan REST web service maka perlu diberikan anotasi `@RestController`.

Pada class header juga ditambahkan anotasi `@RequestMapping("rest")` yang berarti semua request mapping di dalam kelas tersebut harus diawali dengan `"/rest/"`. Seperti pemanggilan service view Student dengan npm 123 dapat diakses melalui `"/rest/student/view/123"`.

Berikut contoh hasil tampilan dalam pemanggilan halaman `localhost:8080/rest/student/view/15066737823`

```
{
  "npm": "1506737823",
  "name": "Kermie",
  "gpa": 3.0
}
```

LATIHAN

Latihan 1

Proses dalam membuat service untuk mengembalikan seluruh student yang ada di dalam basis data. Ditambahkan function seperti berikut pada class StudentRestController yang mengembalikan object berupa List<StudentModel>.

```
@RequestMapping("/student/viewall")
public List<StudentModel> viewall() {
    List<StudentModel> students = studentService.selectAllStudents();
    return students;
}
```

Berikut adalah hasil dari pemanggilan halaman **"localhost:8080/rest/student/viewall/"**

```
String parse JS
[
  {
    "npm": "1506737823",
    "name": "Kermie",
    "gpa": 3.0
  },
  {
    "npm": "1506737824",
    "name": "Ellene",
    "gpa": 3.0
  },
  {
    "npm": "1506737825",
    "name": "Delmer",
    "gpa": 3.0
  },
  {
    "npm": "1506737826",
    "name": "Kristian",
    "gpa": 2.0
  },
  {
    "npm": "1506737827",
    "name": "Marilyn",
    "gpa": 3.0
  },
  {
    "npm": "1506737828",
    "name": "Katti",
    "gpa": 4.0
  }
]
```

Membuat Service Consumer Dan Latihan 2

1. Diimport tutorial 4 untuk menggunakan fungsi dan kelas yang diperlukan untuk membuat Service Consumer ini.
2. Ditambahkan pada application.properties baris berikut

```
server.port = 9090
```

3. Pada package DAO ditambahkan interface dengan nama StudentDAO.
4. Berikut adalah isi dari interface StudentDAO

```
public interface StudentDAO {
    StudentModel selectStudent (String npm);
    List<StudentModel> selectAllStudents();
}
```

5. Dibuat implementasi kelas StudentDAO yang sebelumnya telah dibuat dengan nama kelas StudentDAOImpl.java. Berikut isi dari pada kelas StudentDAOImpl.java

```
@Service
public class StudentDAOImpl implements StudentDAO{
    @Autowired
    private RestTemplate restTemplate;

    @Override
    public StudentModel selectStudent(String npm)
    {
        StudentModel student =
            restTemplate.getForObject("http://localhost:8080/re
st/student/view/"+npm, StudentModel.class);
        return student;
    }

    @Override
    public List<StudentModel> selectAllStudents()
    {
        @SuppressWarnings("unchecked")
        List<StudentModel> students =
            restTemplate.getForObject("http://localhost:8080/re
st/student/viewall", List.class);
        return students;
    }
}
```

Pada class tersebut terdapat 2 function yaitu StudentModel selectStudent dan selectAllStudent. Pada implementasinya digunakan kelas RestTemplate untuk meng-consume object REST web service. Data yang digunakan adalah data-data yang diberikan oleh service producer yang telah dibuat sebelumnya.

Penjelasan Method selectAllStudents(), fungsi akan mengembalikan object List yang berisi object-object student yang didapatkan dari service producer pada halaman localhost:8080/rest/student/viewall.

6. Dibuat kelas baru yaitu StudentServiceRest untuk mengalihkan StudentService agar mengakses data dari web service bukan dari database secara langsung. Berikut adalah isi dari kelas StudentServiceRest.

```
@Slf4j
@Service
@Primary
public class StudentServiceRest implements StudentService {
    @Autowired
    private StudentDAO studentDAO;

    @Override
    public StudentModel selectStudent(String npm)
    {
        log.info("REST - select student with npm{}", npm);
        return studentDAO.selectStudent(npm);
    }

    @Override
    public List<StudentModel> selectAllStudents()
    {
        log.info("REST - select all student");
        return studentDAO.selectAllStudents();
    }

    @Override
    public void addStudent(StudentModel student) {}

    @Override
    public void deleteStudent(String npm) {}

    @Override
    public void updateStudent(StudentModel student) {}

    // @Override
    // public void addCourse(String npm, String idCourse) {}
}
```

Ditambahkan anotasi @Primary agar StudentController mengakses data dari web service.

Penjelasan selectAllStudents() :

Ditambahkan baris return studentDAO.selectAllStudents() agar fungsi mengembalikan List<StudentModel> yang berisi object-object keseluruhan student;

Pada kelas Tutorial07ConsumerApplication.java ditambahkan baris kode berikut untuk menghindari error.

```
@Bean
public RestTemplate restTemplate() {
    return new RestTemplate();
}
```

localhost

https://scele.cs.ui.ac.id/pluginfile.php/35027/mod_r... View All Students +

All Students

Show 10 entries

Search:

No	NPM	Name	GPA	Predikat	Aksi
No. 1	NPM = 1506737823	Name = Kermie	GPA = 3.0	Sangat Memuaskan	Delete Data Update Data
No. 10	NPM = 1506737832	Name = Henri	GPA = 2.0	Sangat Memuaskan	Delete Data Update Data
No. 100	NPM = 1506737922	Name = Vonnice	GPA = 4.0	Cum Laude	Delete Data Update Data
No. 1000	NPM = 1506738822	Name = Nicola	GPA = 2.0	Sangat Memuaskan	Delete Data Update Data
No. 1001	NPM = 1506738823	Name = Kermie	GPA = 3.0	Sangat Memuaskan	Delete Data Update Data
No. 1002	NPM = 1506738824	Name = Ellene	GPA = 3.0	Sangat Memuaskan	Delete Data Update Data
No. 1003	NPM = 1506738825	Name = Delmer	GPA = 3.0	Sangat Memuaskan	Delete Data Update Data
No. 1004	NPM = 1506738826	Name = Kristian	GPA = 2.0	Sangat Memuaskan	Delete Data Update Data
No.	NPM =	Name =	GPA =	Sangat	Delete Data

Console Progress Problems

Tutorial07Consumer - Tutorial07ConsumerApplication [Spring Boot App] /Library/Java/JavaVirtualMachines/jdk1.8.0_131.jdk
 itil.threads.TaskThread\$WrappingRunnable.run(TaskThread.java:61) [tomcat-embed-core-8.5.29.jar:8.5.29]
 in(Thread.java:748) [na:1.8.0_131]

5364 --- [nio-9090-exec-2] c.f.t.service.StudentServiceRest : REST - select all student
 5364 --- [nio-9090-exec-4] c.f.t.service.StudentServiceRest : REST - select all student