Tutorial 7

Pada kesempatan kali ini kita mempelajari bagaimana cara penggunaan sebuah Web Service dengan menggunakan 2 Project yang berbeda. Project yang dimaksud adalah Producer sebagai penyedia data & Customer yang akan mengambil data menggunakan Rest Template pada ServiceRest yang diajarkan pada tutorial tersebut. Hal yang harus diperhatikan bahwa anotasi @Primary yang digunakan pada Service Rest akan me autowired otomatis menginstansiasi StudentService menggunakan StudentServiceRest karena dia primary.

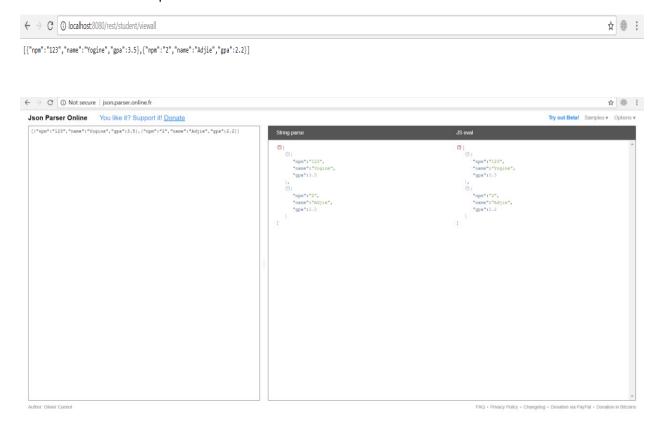
Latihan 1

Buatlah service untuk mengembalikan seluruh student yang ada di basis data. Service ini mirip seperti method viewAll di Web Controller. Service tersebut di-mapping ke "/rest/student/viewall"

Tambahkan Method dibawah ini pada service yang akan mapping pada /student/viewall:

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

Maka akan muncul Output sbb:



Latihan 2

Implementasikan service consumer untuk view all Students dengan melengkapi method selectAllStudents yang ada di kelas StudentServiceRest.

 $Tambahkan\ method\ dibawah\ pada\ Student Service Rest:$

```
@Override
public List<StudentModel> selectAllStudents() {
    // TODO Auto-generated method stub
    Log.info("REST - select all students");
    return studentDAO.selectAllStudent();
}
```

Dimana method tersebut akan mengembalikan kepada StudentDAO.SelectAllStudent();

Buat lah method selectAllStudent() pada StudentDaoImpl.java:

```
@Override
public List<StudentModel> selectAllStudent() {
    ResponseEntity<StudentModel[]> studentList = restTemplate.getForEntity("http://localhost:8080/rest/student/viewall",StudentModel[].class);
    return Arrays.asList(studentList.getBody());
}
```

Berikut hasil output menggunakan Port:9090



Mata Kuliah APAP