====================

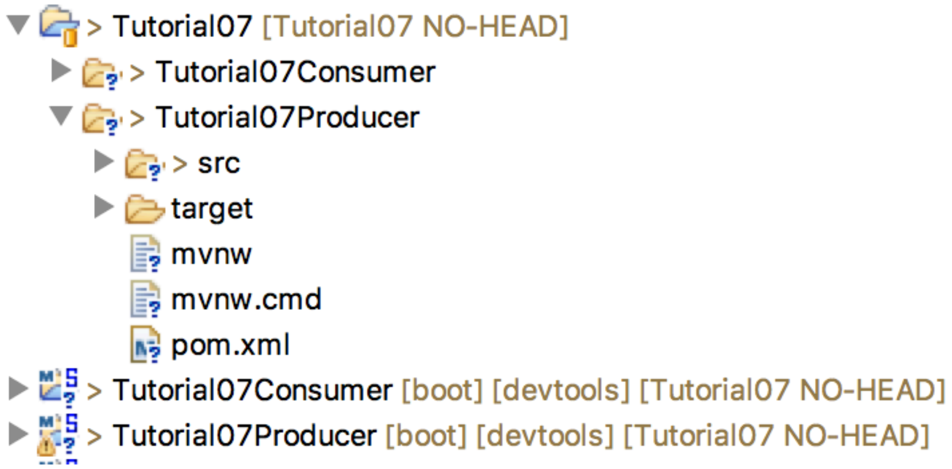
Tutorial 7
Emma Sharahwati
1406557466
APAP-B

====================

1. **Buat folder dengan nama ‘Tutorial07’**

Pada folder root ‘Tutorial07’ nantinya anda akan membuat dua buah *project* , yaitu Tutorial 07 – Membuat Web Service Menggunakan Spring Boot Framework | 1

‘Tutorial07Producer’ dan ‘Tutorial07Consumer’.

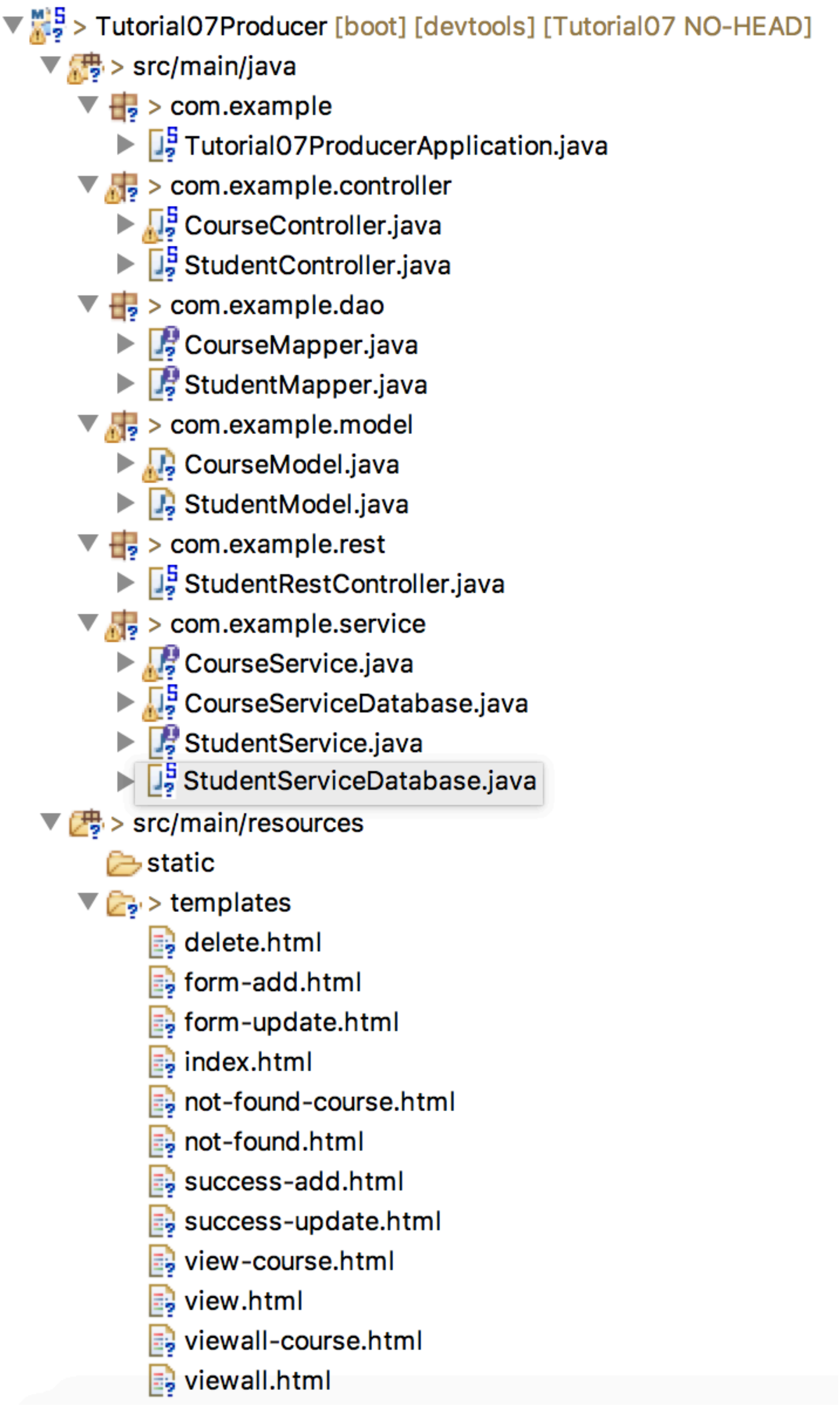


```
▼ > Tutorial07 [Tutorial07 NO-HEAD]
    ▶ > Tutorial07Consumer
    ▼ > Tutorial07Producer
        ▶ > src
        ▶ target
          mvnw
          mvnw.cmd
          pom.xml
▶ > Tutorial07Consumer [boot] [devtools] [Tutorial07 NO-HEAD]
▶ > Tutorial07Producer [boot] [devtools] [Tutorial07 NO-HEAD]
```

2. **Inisialisasi folder Tutorial07 menjadi git project**
Setelah inisiasi dengan *git init,* jangan lupa untuk *set remote link* ke *repository* Tutorial07 dengan *git remote set origin.*

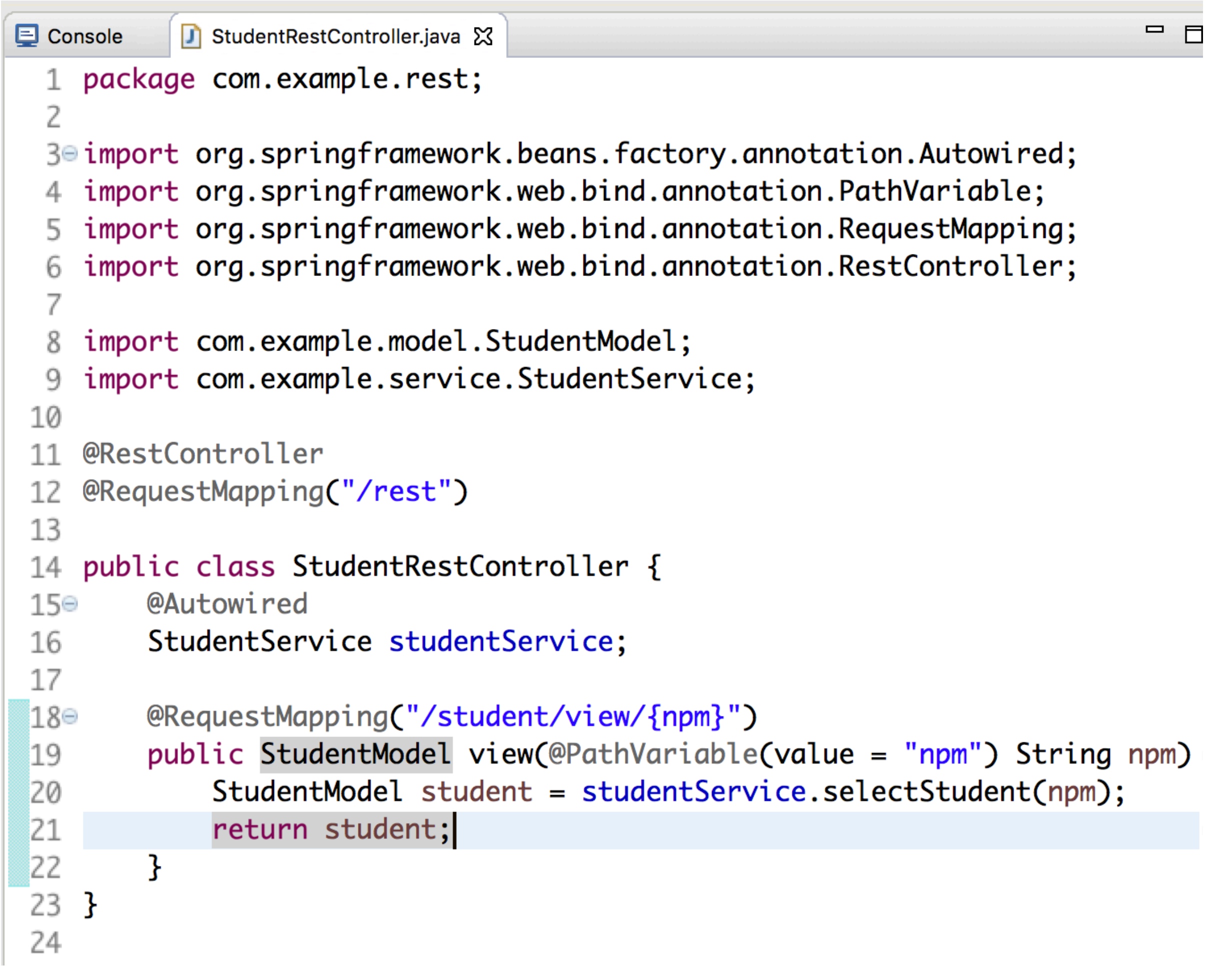
Membuat Service Provider

1. **Import berkas-berkas yang ada di tutorial 5 untuk Service producer ini.**
2. **Buat *package* baru com.example.rest, atau sesuaikan dengan *package* Anda masing-masing**



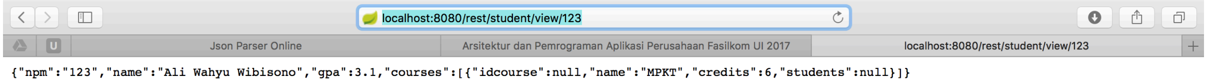
```
> Tutorial07Producer [boot] [devtools] [Tutorial07 NO-HEAD]
  > src/main/java
    > com.example
      Tutorial07ProducerApplication.java
    > com.example.controller
      CourseController.java
      StudentController.java
    > com.example.dao
      CourseMapper.java
      StudentMapper.java
    > com.example.model
      CourseModel.java
      StudentModel.java
    > com.example.rest
      StudentRestController.java
    > com.example.service
      CourseService.java
      CourseServiceDatabase.java
      StudentService.java
      StudentServiceDatabase.java
  > src/main/resources
    static
    > templates
      delete.html
      form-add.html
      form-update.html
      index.html
      not-found-course.html
      not-found.html
      success-add.html
      success-update.html
      view-course.html
      view.html
      viewall-course.html
      viewall.html
```

3. Buat class baru yaitu **StudentRestController.java pada package com.example.rest**

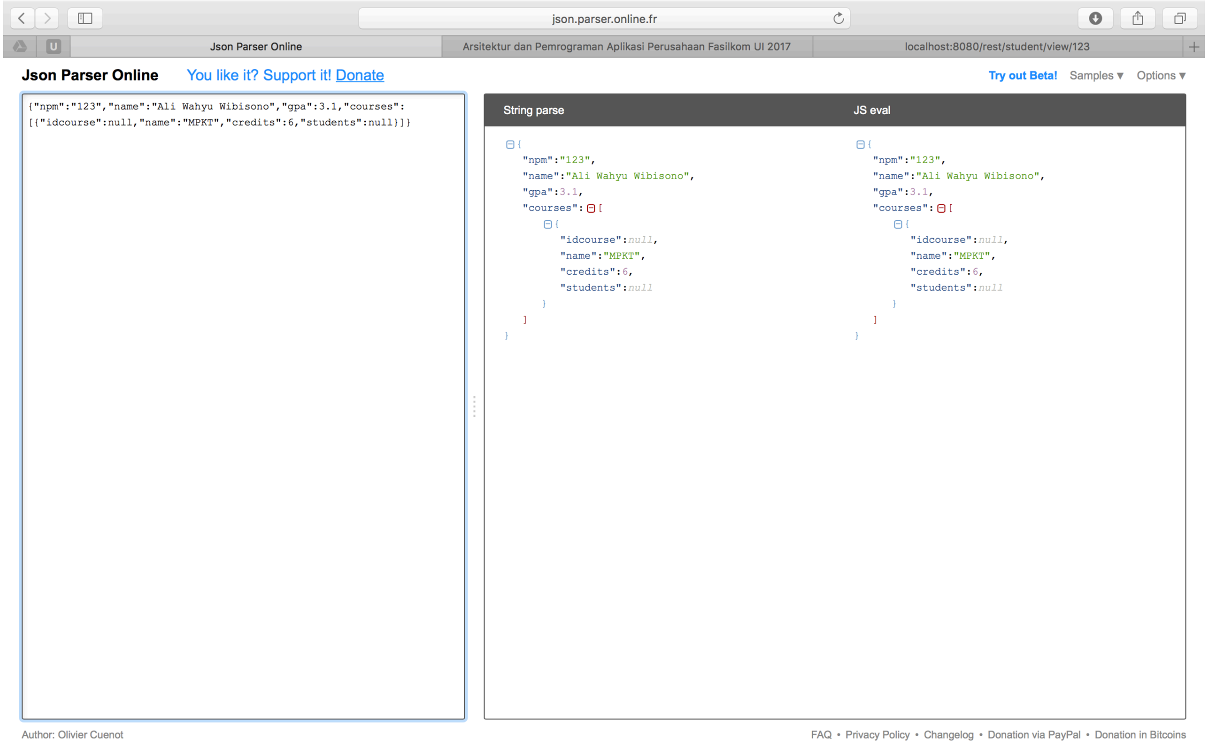


```java
package com.example.rest;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.web.bind.annotation.PathVariable;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RestController;

import com.example.model.StudentModel;
import com.example.service.StudentService;

@RestController
@RequestMapping("/rest")

public class StudentRestController {
    @Autowired
    StudentService studentService;

    @RequestMapping("/student/view/{npm}")
    public StudentModel view(@PathVariable(value = "npm") String npm) {
        StudentModel student = studentService.selectStudent(npm);
        return student;
    }
}

```

http://localhost:8080/rest/student/view/123


localhost:8080/rest/student/view/123
Json Parser Online
Arsitektur dan Pemrograman Aplikasi Perusahaan Fasilkom UI 2017
localhost:8080/rest/student/view/123

{"npm":"123","name":"Ali Wahyu Wibisono","gpa":3.1,"courses":[{"idcourse":null,"name":"MPKT","credits":6,"students":null}]}


json.parser.online.fr
Json Parser Online
Arsitektur dan Pemrograman Aplikasi Perusahaan Fasilkom UI 2017
localhost:8080/rest/student/view/123
Json Parser Online
You like it? Support it! Donate
Try out Beta!
Samples ▾
Options ▾
String parse
JS eval
Author: Olivier Cuenot
FAQ • Privacy Policy • Changelog • Donation via PayPal • Donation in Bitcoins

```
{"npm":"123","name":"Ali Wahyu Wibisono","gpa":3.1,"courses":
[{"idcourse":null,"name":"MPKT","credits":6,"students":null}]}
```

```
{
   "npm":"123",
   "name":"Ali Wahyu Wibisono",
   "gpa":3.1,
   "courses":[
      {
         "idcourse":null,
         "name":"MPKT",
         "credits":6,
         "students":null
      }
   ]
}
```

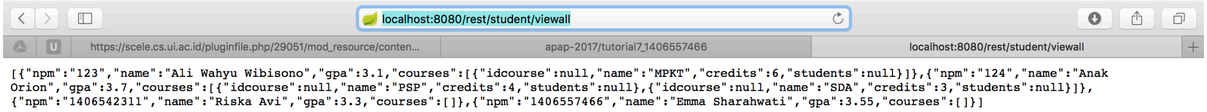
```
{
   "npm":"123",
   "name":"Ali Wahyu Wibisono",
   "gpa":3.1,
   "courses":[
      {
         "idcourse":null,
         "name":"MPKT",
         "credits":6,
         "students":null
      }
   ]
}
```

LATIHAN

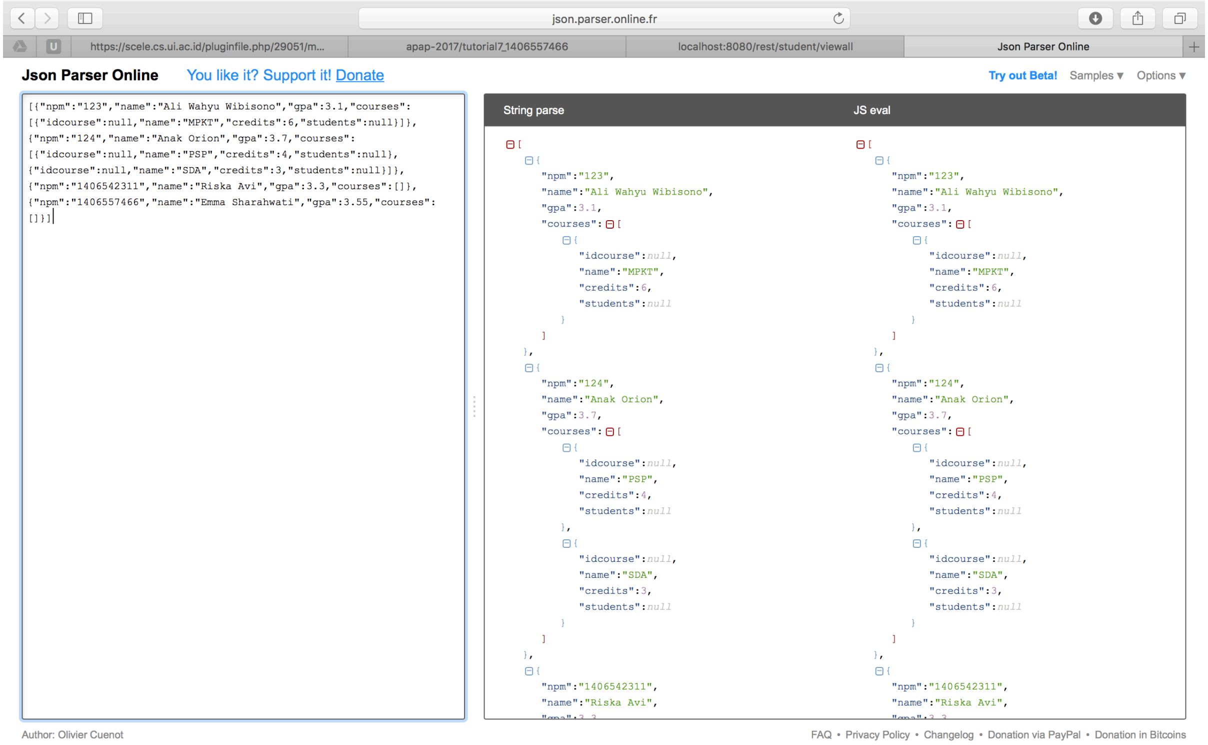
1. **Latihan 1:** Buatlah *service* untuk mengembalikan seluruh student yang ada di basis data. *Service* ini mirip seperti *method* viewAll di Web Controller. *Service* tersebut di-*mapping* ke **"/rest/student/viewall"**

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

Output :


localhost:8080/rest/student/viewall
https://scele.cs.ui.ac.id/pluginfile.php/29051/mod_resource/conten...
apap-2017/tutorial7_1406557466
localhost:8080/rest/student/viewall

```
[{"npm":"123","name":"Ali Wahyu Wibisono","gpa":3.1,"courses":[{"idcourse":null,"name":"MPKT","credits":6,"students":null}]},{"npm":"124","name":"Anak
Orion","gpa":3.7,"courses":[{"idcourse":null,"name":"PSP","credits":4,"students":null},{"idcourse":null,"name":"SDA","credits":3,"students":null}]},
{"npm":"1406542311","name":"Riska Avi","gpa":3.3,"courses":[]},{"npm":"1406557466","name":"Emma Sharahwati","gpa":3.55,"courses":[]}]
```

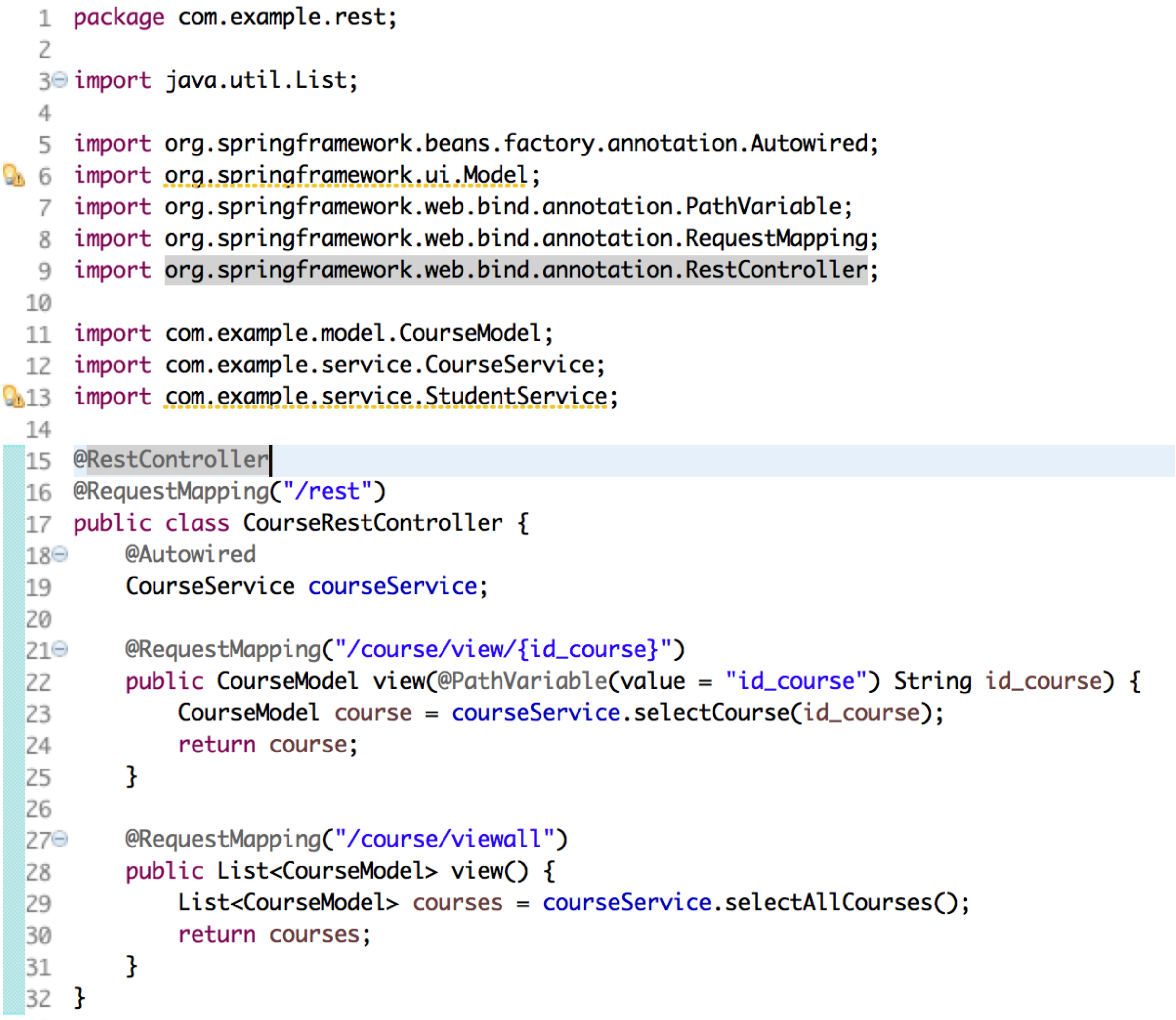

json.parser.online.fr
https://scele.cs.ui.ac.id/pluginfile.php/29051/m...
apap-2017/tutorial7_1406557466
localhost:8080/rest/student/viewall
Json Parser Online
Json Parser Online
You like it? Support it! Donate
Try out Beta!
Samples ▾
Options ▾
String parse
JS eval
Author: Olivier Cuenot
FAQ • Privacy Policy • Changelog • Donation via PayPal • Donation in Bitcoins

```
[{"npm":"123","name":"Ali Wahyu Wibisono","gpa":3.1,"courses":
[{"idcourse":null,"name":"MPKT","credits":6,"students":null}]},
{"npm":"124","name":"Anak Orion","gpa":3.7,"courses":
[{"idcourse":null,"name":"PSP","credits":4,"students":null},
{"idcourse":null,"name":"SDA","credits":3,"students":null}]},
{"npm":"1406542311","name":"Riska Avi","gpa":3.3,"courses":[]},
{"npm":"1406557466","name":"Emma Sharahwati","gpa":3.55,"courses":
[]}]
```

```
⊟[
   ⊟{
      "npm":"123",
      "name":"Ali Wahyu Wibisono",
      "gpa":3.1,
      "courses":⊟[
         ⊟{
            "idcourse":null,
            "name":"MPKT",
            "credits":6,
            "students":null
         }
      ]
   },
   ⊟{
      "npm":"124",
      "name":"Anak Orion",
      "gpa":3.7,
      "courses":⊟[
         ⊟{
            "idcourse":null,
            "name":"PSP",
            "credits":4,
            "students":null
         },
         ⊟{
            "idcourse":null,
            "name":"SDA",
            "credits":3,
            "students":null
         }
      ]
   },
   ⊟{
      "npm":"1406542311",
      "name":"Riska Avi",
      "gpa":3.3
```

```
⊟[
   ⊟{
      "npm":"123",
      "name":"Ali Wahyu Wibisono",
      "gpa":3.1,
      "courses":⊟[
         ⊟{
            "idcourse":null,
            "name":"MPKT",
            "credits":6,
            "students":null
         }
      ]
   },
   ⊟{
      "npm":"124",
      "name":"Anak Orion",
      "gpa":3.7,
      "courses":⊟[
         ⊟{
            "idcourse":null,
            "name":"PSP",
            "credits":4,
            "students":null
         },
         ⊟{
            "idcourse":null,
            "name":"SDA",
            "credits":3,
            "students":null
         }
      ]
   },
   ⊟{
      "npm":"1406542311",
      "name":"Riska Avi",
      "gpa":3.3
```

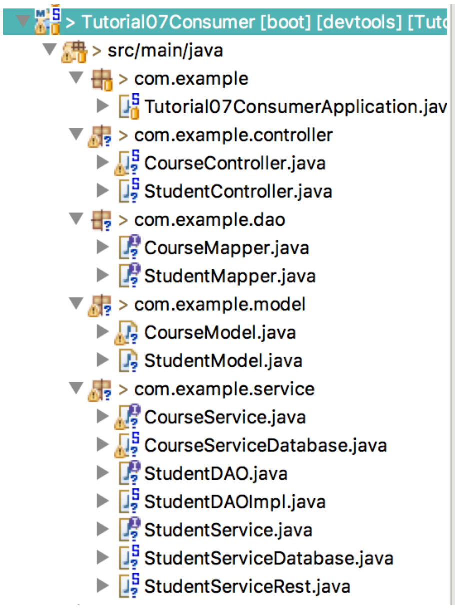
2. **Latihan 2:** Buatlah *service* untuk class Course. Buatlah *controller* baru yang terdapat *service* untuk melihat suatu course dengan masukan ID Course (view by ID) dan *service* untuk melihat semua course (view all).



```java
package com.example.rest;

import java.util.List;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.ui.Model;
import org.springframework.web.bind.annotation.PathVariable;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RestController;

import com.example.model.CourseModel;
import com.example.service.CourseService;
import com.example.service.StudentService;

@RestController
@RequestMapping("/rest")
public class CourseRestController {
    @Autowired
    CourseService courseService;

    @RequestMapping("/course/view/{id_course}")
    public CourseModel view(@PathVariable(value = "id_course") String id_course) {
        CourseModel course = courseService.selectCourse(id_course);
        return course;
    }

    @RequestMapping("/course/viewall")
    public List<CourseModel> view() {
        List<CourseModel> courses = courseService.selectAllCourses();
        return courses;
    }
}
```

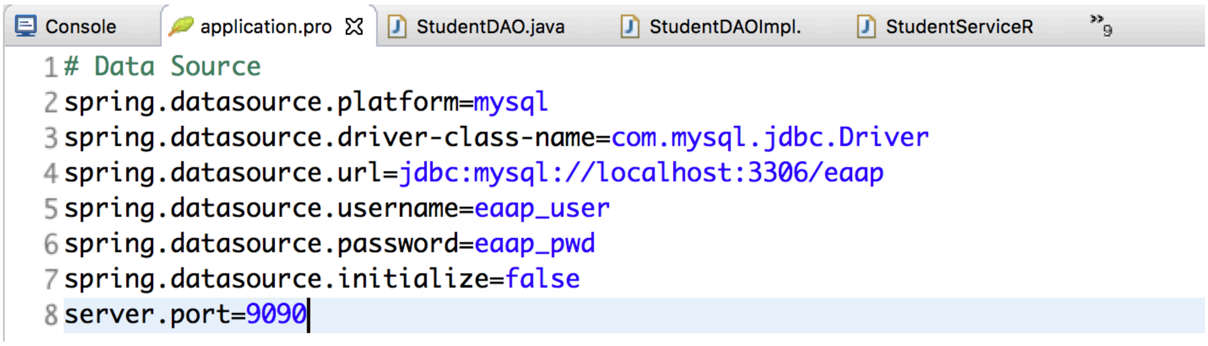
Membuat Service Consumer

1. **Import berkas-berkas yang ada di tutorial 6 untuk Service Consumer ini.**



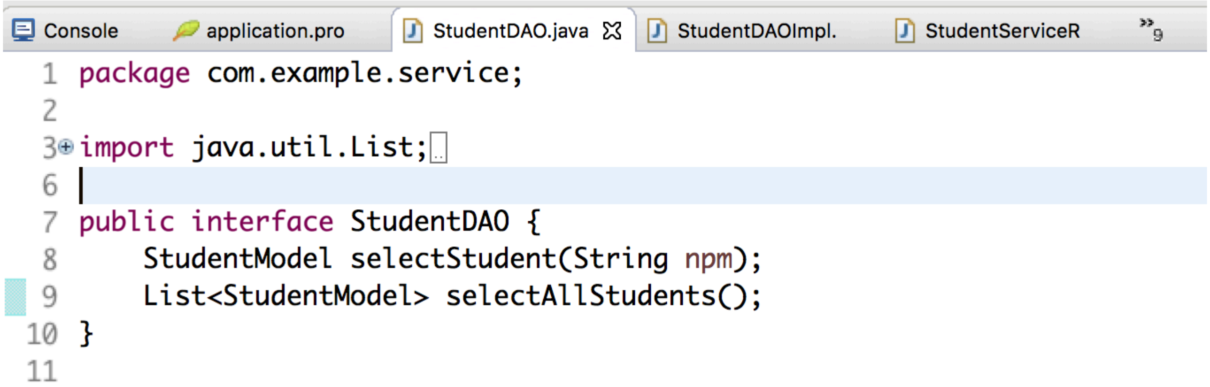
```
> Tutorial07Consumer [boot] [devtools] [Tut
  > src/main/java
    > com.example
      Tutorial07ConsumerApplication.jav
    > com.example.controller
      CourseController.java
      StudentController.java
    > com.example.dao
      CourseMapper.java
      StudentMapper.java
    > com.example.model
      CourseModel.java
      StudentModel.java
    > com.example.service
      CourseService.java
      CourseServiceDatabase.java
      StudentDAO.java
      StudentDAOImpl.java
      StudentService.java
      StudentServiceDatabase.java
      StudentServiceRest.java
```

2. Karena *Service Producer* dan *Service Consumer* harus dijalankan secara bersamaan, ubah berkas **application.properties**



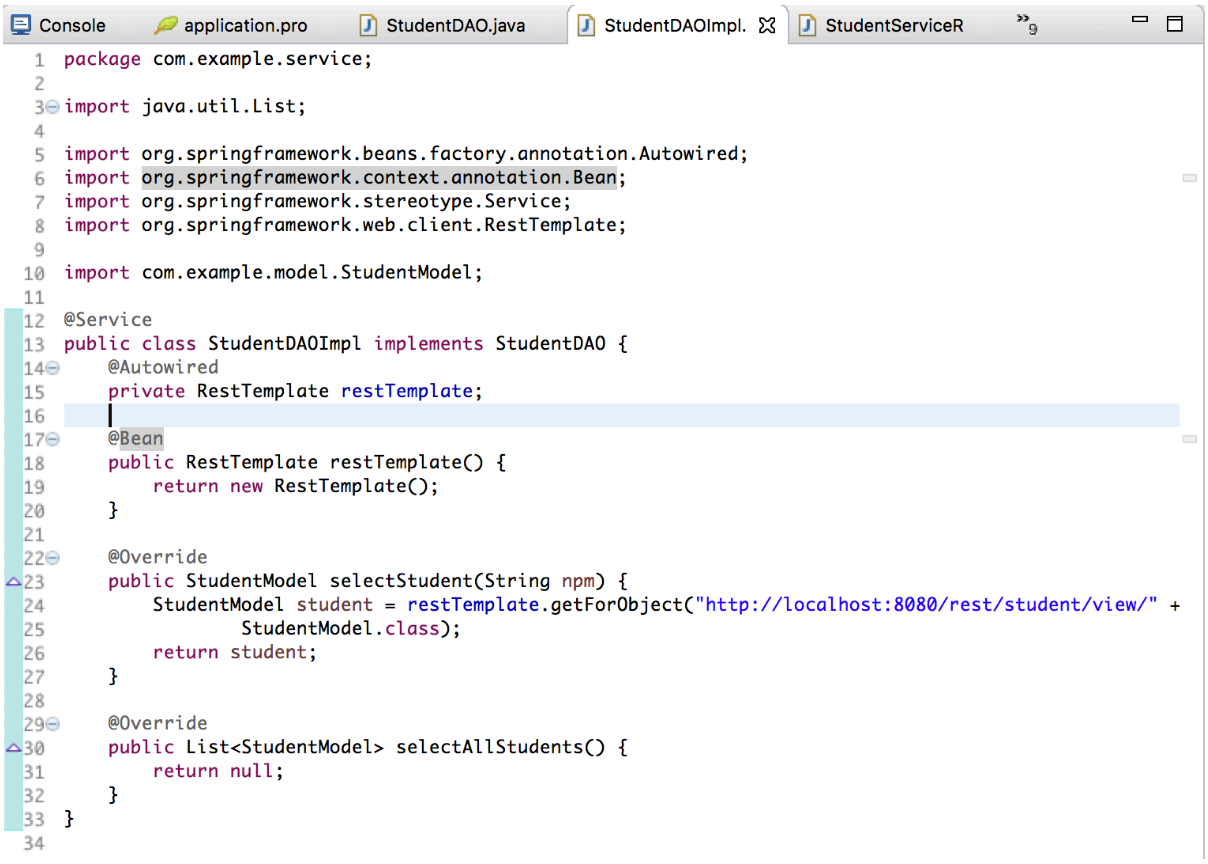
```
# Data Source
spring.datasource.platform=mysql
spring.datasource.driver-class-name=com.mysql.jdbc.Driver
spring.datasource.url=jdbc:mysql://localhost:3306/eaap
spring.datasource.username=eaap_user
spring.datasource.password=eaap_pwd
spring.datasource.initialize=false
server.port=9090
```

3. Di DAO tambahkan ***interface*** dengan nama StudentDAO



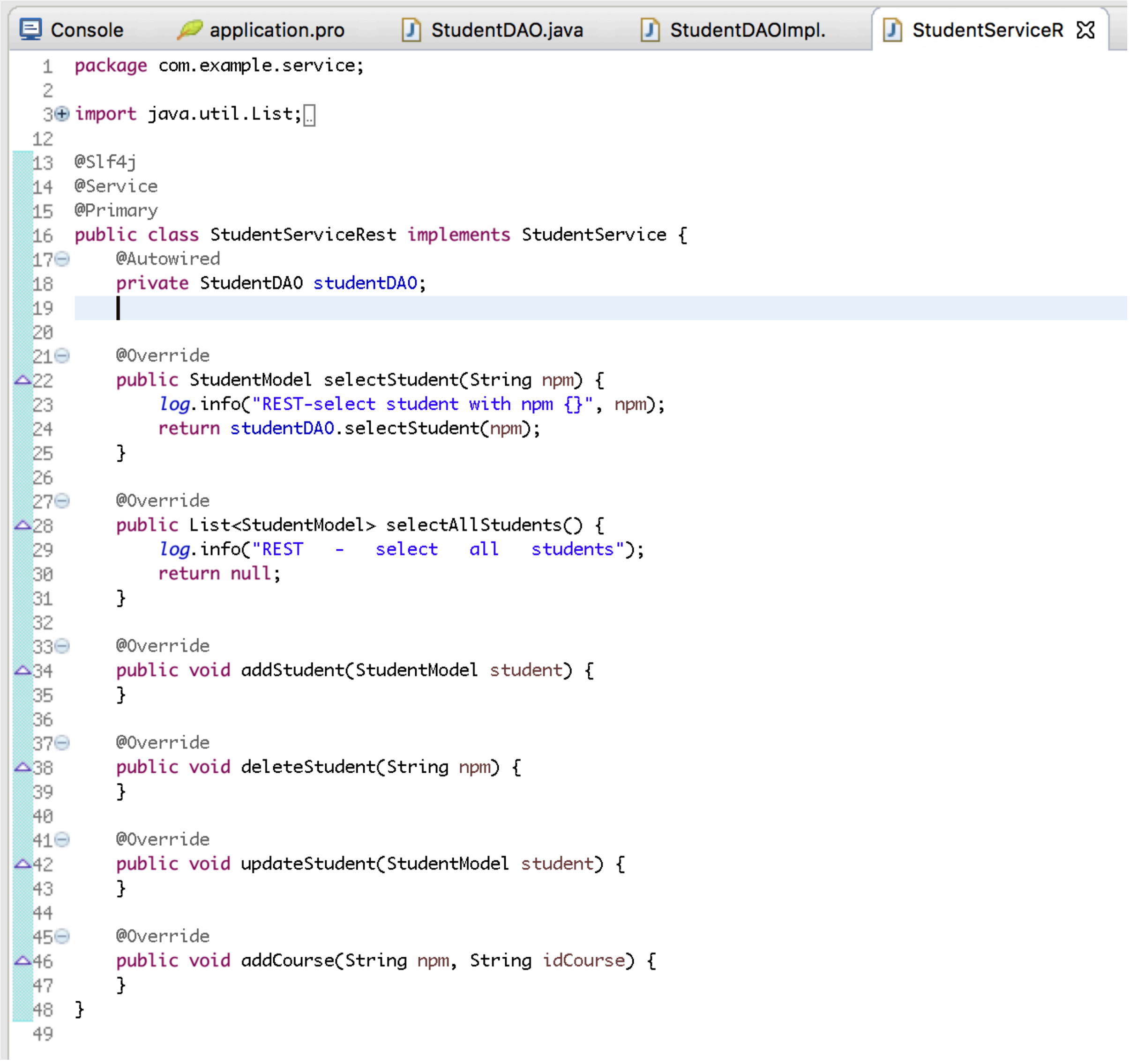
```java
package com.example.service;

import java.util.List;


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

```

4. Selanjutnya kita akan membuat implementasi kelas StudentDAO tersebut dengan nama kelas StudentDAOImpl.java.



```java
package com.example.service;

import java.util.List;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.context.annotation.Bean;
import org.springframework.stereotype.Service;
import org.springframework.web.client.RestTemplate;

import com.example.model.StudentModel;

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

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

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

    @Override
    public List<StudentModel> selectAllStudents() {
        return null;
    }
}

```

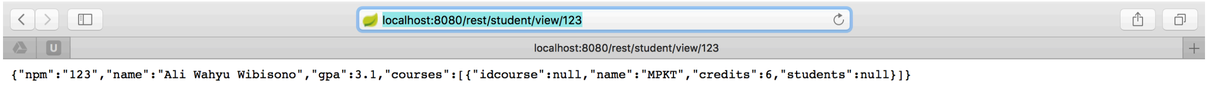
5. Selanjutnya, kita ingin mengubah agar StudentService mengambil data dari *web service* bukan dari *database* . Kita tidak perlu menghapus *class* **StudentServiceDatabase** . Karena *scalable system* , kita cukup menambahkan *class* baru yaitu **StudentServiceRest** yang mengimplement StudentService di *package* service.



```java
package com.example.service;

import java.util.List;

@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  students");
        return null;
    }

    @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) {
    }
}
```

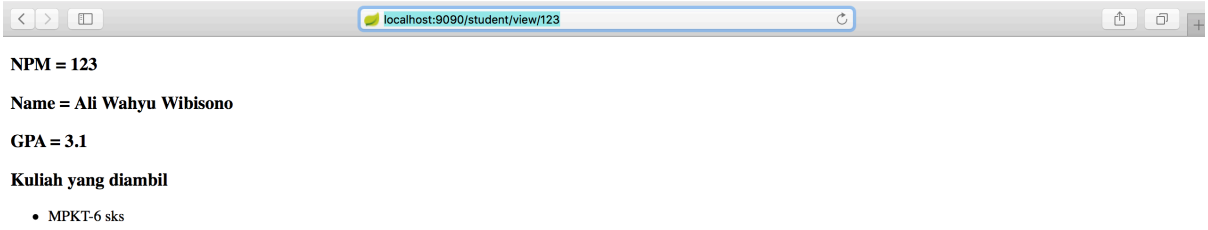
Output :

localhost:8080/rest/student/view/123.


localhost:8080/rest/student/view/123
localhost:8080/rest/student/view/123

```
{"npm":"123","name":"Ali Wahyu Wibisono","gpa":3.1,"courses":[{"idcourse":null,"name":"MPKT","credits":6,"students":null}]}
```

localhost:9090/student/view/123



NPM = 123

Name = Ali Wahyu Wibisono

GPA = 3.1

Kuliah yang diambil

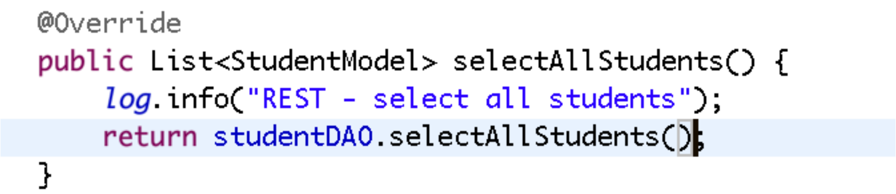
- MPKT-6 sks

Di console :

```
[nio-9090-exec-1] com.example.service.StudentServiceRest    : REST-select student wit
```

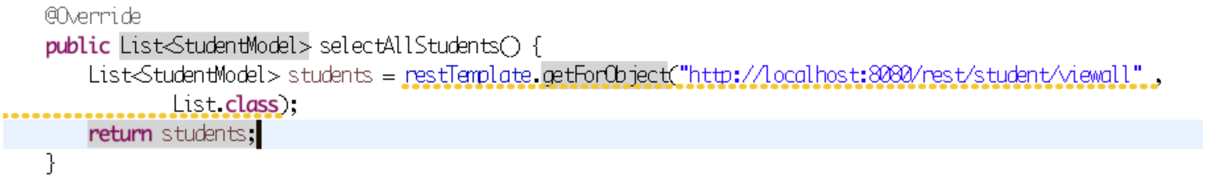
LATIHAN

1. **Latihan 3:** Implementasikan *service consumer* untuk view all Students dengan melengkapi *method* selectAllStudents yang ada di kelas **StudentServiceRest** .



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

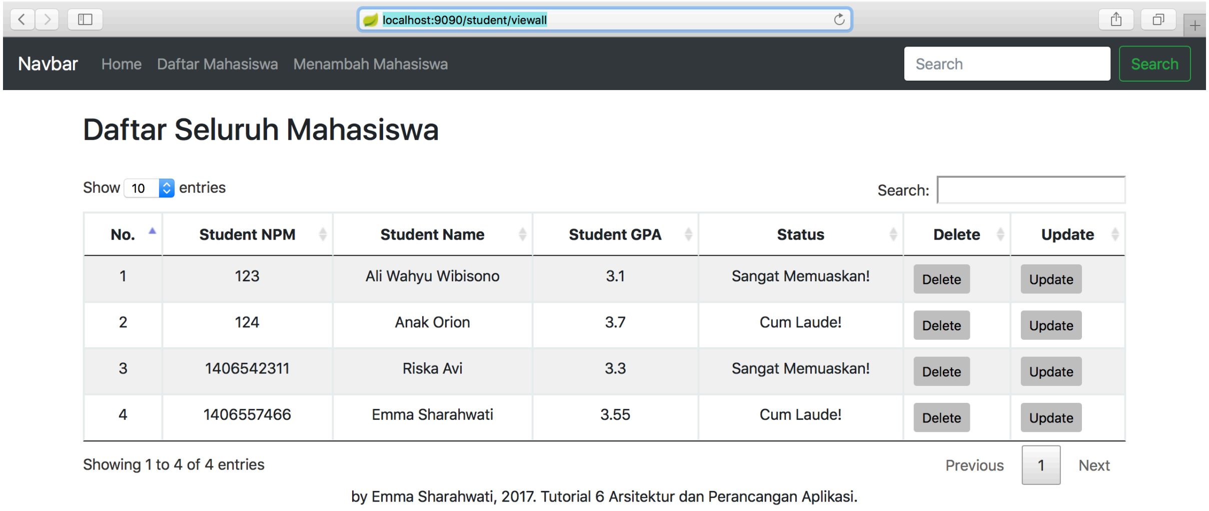
method selectAllStudent di studentDAOImpl.java juga diubah menjadi seperti ini:



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

Output :

localhost:9090/student/viewall



Daftar Seluruh Mahasiswa

Show 10 entries

Search:

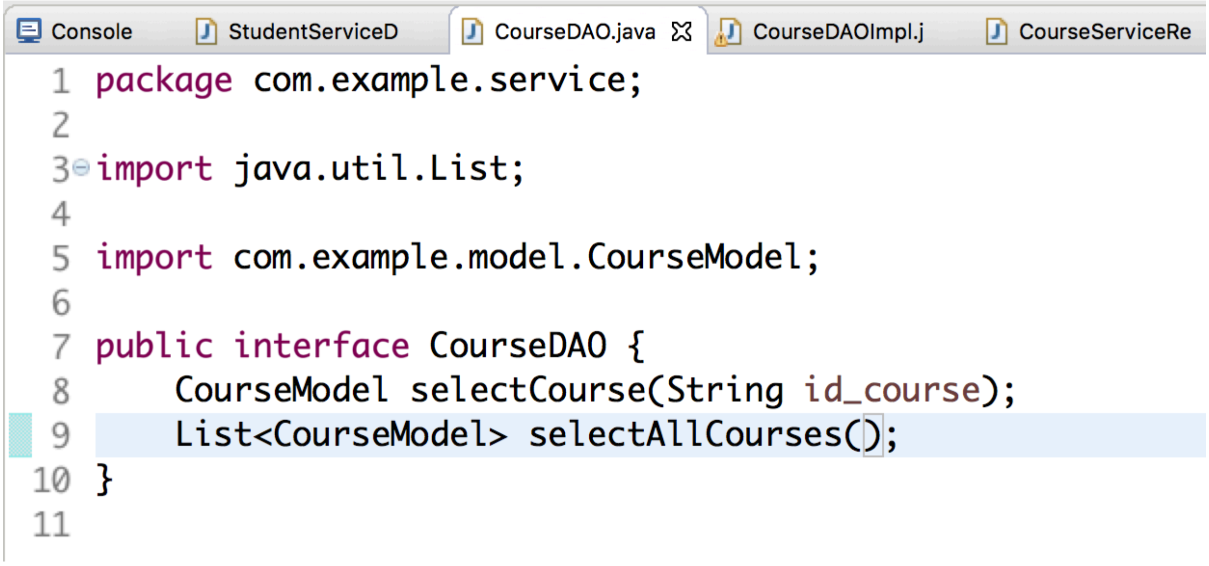
No.	Student NPM	Student Name	Student GPA	Status	Delete	Update
1	123	Ali Wahyu Wibisono	3.1	Sangat Memuaskan!	Delete	Update
2	124	Anak Orion	3.7	Cum Laude!	Delete	Update
3	1406542311	Riska Avi	3.3	Sangat Memuaskan!	Delete	Update
4	1406557466	Emma Sharahwati	3.55	Cum Laude!	Delete	Update

Showing 1 to 4 of 4 entries

by Emma Sharahwati, 2017. Tutorial 6 Arsitektur dan Perancangan Aplikasi.

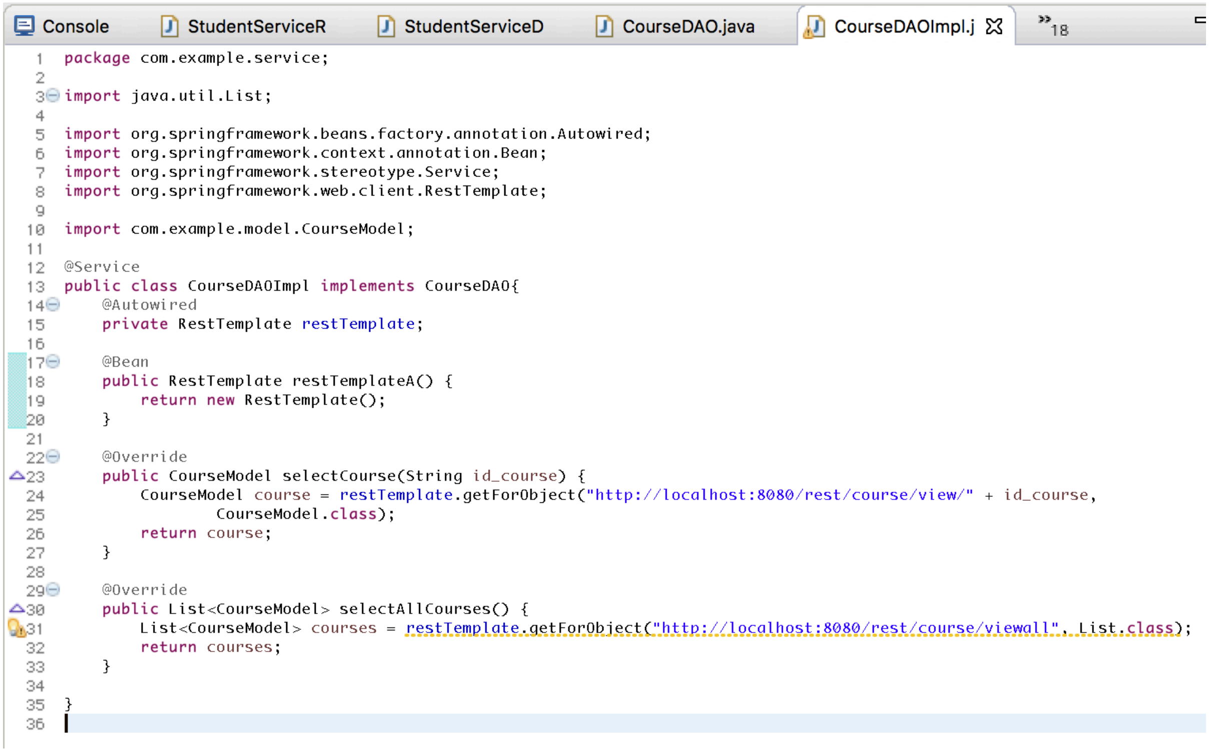
2. **Latihan 4:** Implementasikan *service consumer* untuk *class* CourseModel dengan membuat *class-class* DAO dan *service* baru.

- Membuat interface CourseDAO

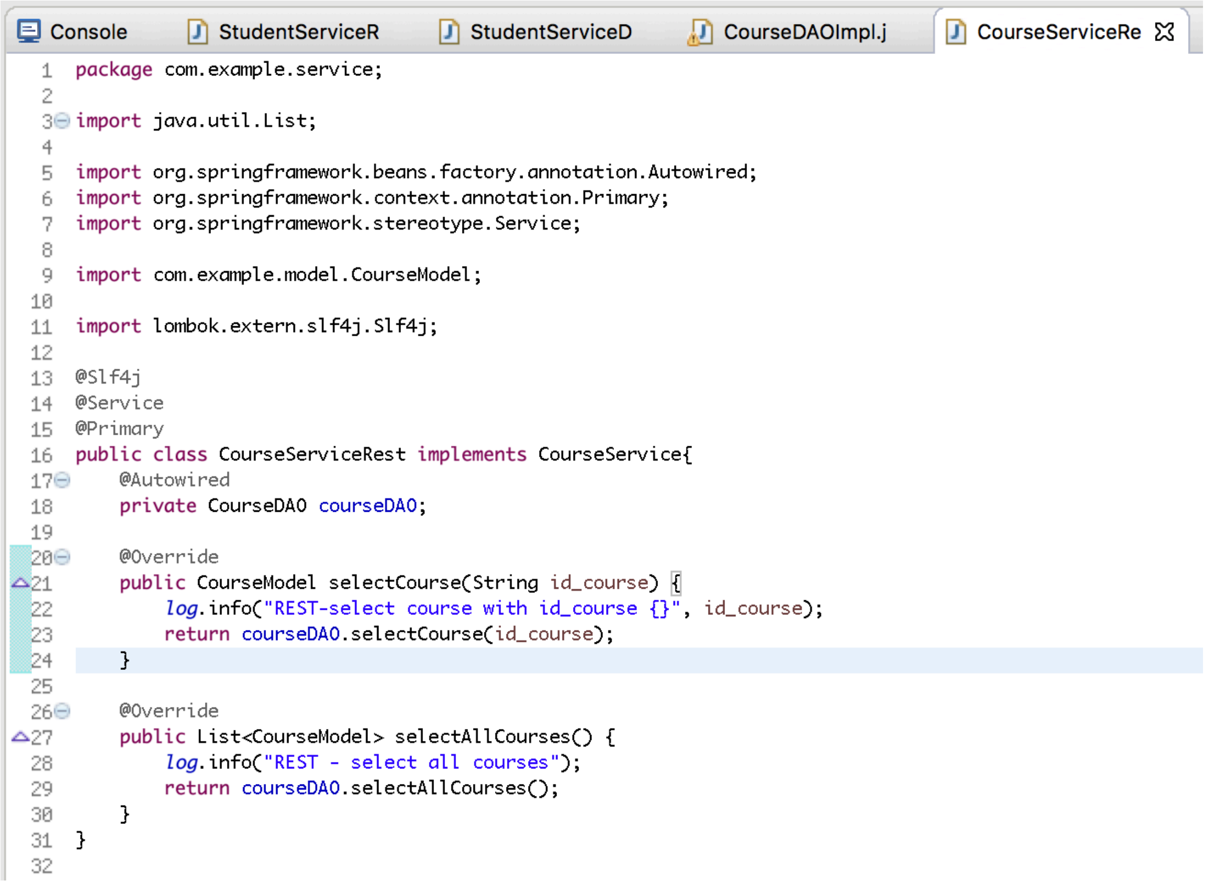


```java
package com.example.service;

import java.util.List;

import com.example.model.CourseModel;

public interface CourseDAO {
    CourseModel selectCourse(String id_course);
    List<CourseModel> selectAllCourses();
}

```

- Membuat Class CourseDAOImpl dan CourseServiceRest



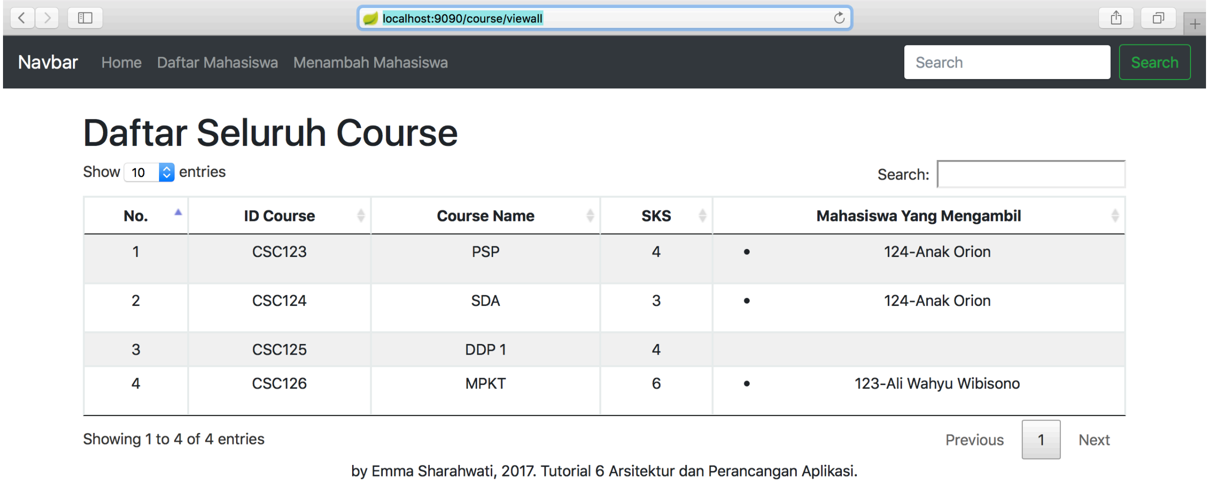
```java
package com.example.service;

import java.util.List;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.context.annotation.Bean;
import org.springframework.stereotype.Service;
import org.springframework.web.client.RestTemplate;

import com.example.model.CourseModel;

@Service
public class CourseDAOImpl implements CourseDAO{
    @Autowired
    private RestTemplate restTemplate;

    @Bean
    public RestTemplate restTemplateA() {
        return new RestTemplate();
    }

    @Override
    public CourseModel selectCourse(String id_course) {
        CourseModel course = restTemplate.getForObject("http://localhost:8080/rest/course/view/" + id_course,
                CourseModel.class);
        return course;
    }

    @Override
    public List<CourseModel> selectAllCourses() {
        List<CourseModel> courses = restTemplate.getForObject("http://localhost:8080/rest/course/viewall", List.class);
        return courses;
    }

}

```



```java
package com.example.service;

import java.util.List;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.context.annotation.Primary;
import org.springframework.stereotype.Service;

import com.example.model.CourseModel;

import lombok.extern.slf4j.Slf4j;

@Slf4j
@Service
@Primary
public class CourseServiceRest implements CourseService{
    @Autowired
    private CourseDAO courseDAO;

    @Override
    public CourseModel selectCourse(String id_course) {
        log.info("REST-select course with id_course {}", id_course);
        return courseDAO.selectCourse(id_course);
    }

    @Override
    public List<CourseModel> selectAllCourses() {
        log.info("REST - select all courses");
        return courseDAO.selectAllCourses();
    }
}

```

Output :

http://localhost:9090/course/viewall



Daftar Seluruh Course

Show 10 entries Search:

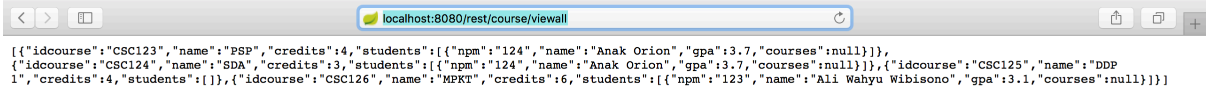
No.	ID Course	Course Name	SKS	Mahasiswa Yang Mengambil
1	CSC123	PSP	4	• 124-Anak Orion
2	CSC124	SDA	3	• 124-Anak Orion
3	CSC125	DDP 1	4	
4	CSC126	MPKT	6	• 123-Ali Wahyu Wibisono

Showing 1 to 4 of 4 entries

Previous 1 Next

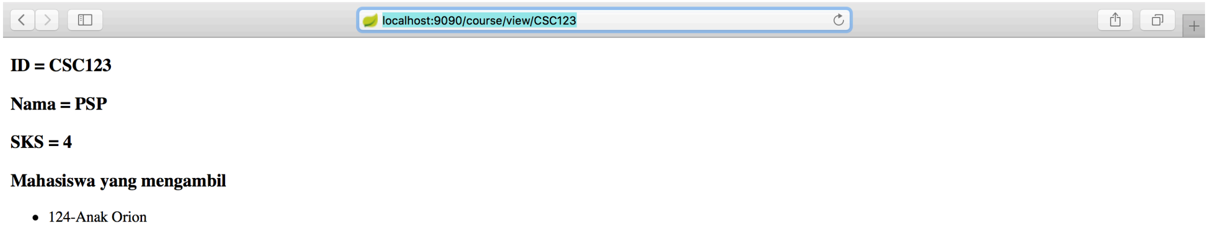
by Emma Sharahwati, 2017. Tutorial 6 Arsitektur dan Perancangan Aplikasi.

http://localhost:8080/rest/course/viewall



[{"idcourse":"CSC123","name":"PSP","credits":4,"students":[{"npm":"124","name":"Anak Orion","gpa":3.7,"courses":null}]},
{"idcourse":"CSC124","name":"SDA","credits":3,"students":[{"npm":"124","name":"Anak Orion","gpa":3.7,"courses":null}]},{"idcourse":"CSC125","name":"DDP
1","credits":4,"students":[]},{"idcourse":"CSC126","name":"MPKT","credits":6,"students":[{"npm":"123","name":"Ali Wahyu Wibisono","gpa":3.1,"courses":null}]}]

http://localhost:9090/course/view/CSC123



ID = CSC123

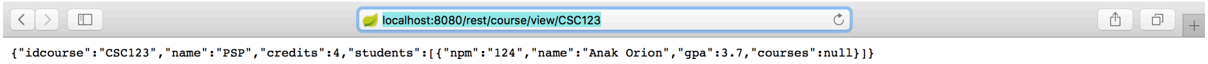
Nama = PSP

SKS = 4

Mahasiswa yang mengambil

- 124-Anak Orion

http://localhost:8080/rest/course/view/CSC123



```
{"idcourse":"CSC123","name":"PSP","credits":4,"students":[{"npm":"124","name":"Anak Orion","gpa":3.7,"courses":null}]}
```