

Nama : Budi Indrawan  
NPM : 1606954754

## Tutorial Web Service Menggunakan Spring Boot Framework

### A. Ringkasan Tutorial 7

- Pada tutorial4 ini mempelajari cara membuat *Web Service* yang digunakan untuk membantu aplikasi berhubungan dengan aplikasi yang lain. Dan pada tutorial ini terdapat *service Producer* sebagai *layer backend* yang menyediakan *Web service* tersebut dalam sebuah URL yang akan mengembalikan representasi data menggunakan JSON.

### B. Tutorial

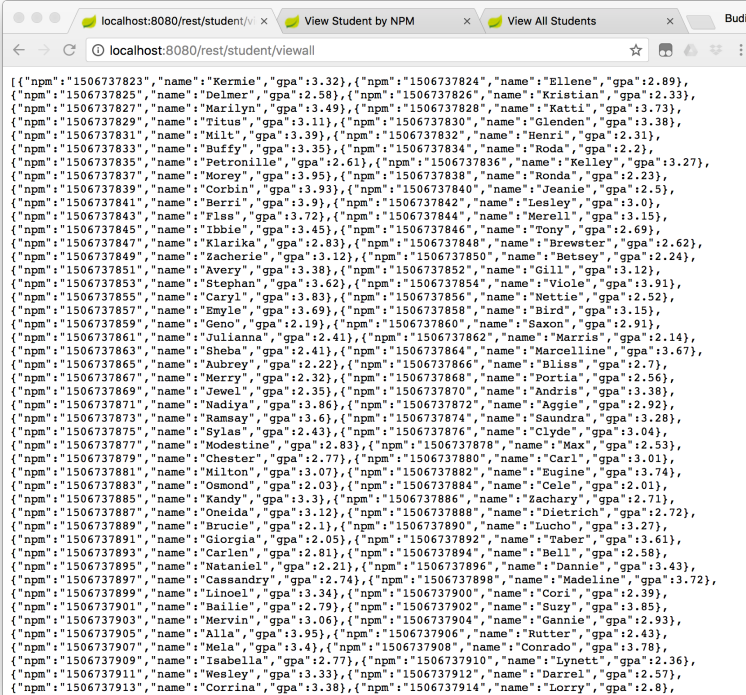
- ❖ **LATIHAN 1** : membuat service untuk mengembalikan seluruh student yang ada di basis data
- Membuat method dengan mapping “/student/viewall”



```
1 package com.example.rest;
2
3 import java.util.List;
4
5 @RestController
6 @RequestMapping ( "/rest" )
7 public class StudentRestController {
8
9     @Autowired
10     StudentService studentService;
11
12     @RequestMapping ( "/student/view/{npm}" )
13     public StudentModel view ( @PathVariable ( value = "npm" ) String npm ) {
14         StudentModel student = studentService.selectStudent ( npm );
15         return student;
16     }
17
18     @RequestMapping ( "/student/viewall" )
19     public List < StudentModel > viewAll ( Model model ) {
20         List < StudentModel > students = studentService.selectAllStudents ();
21         model.addAttribute ( "students", students );
22         return students;
23     }
24 }
```

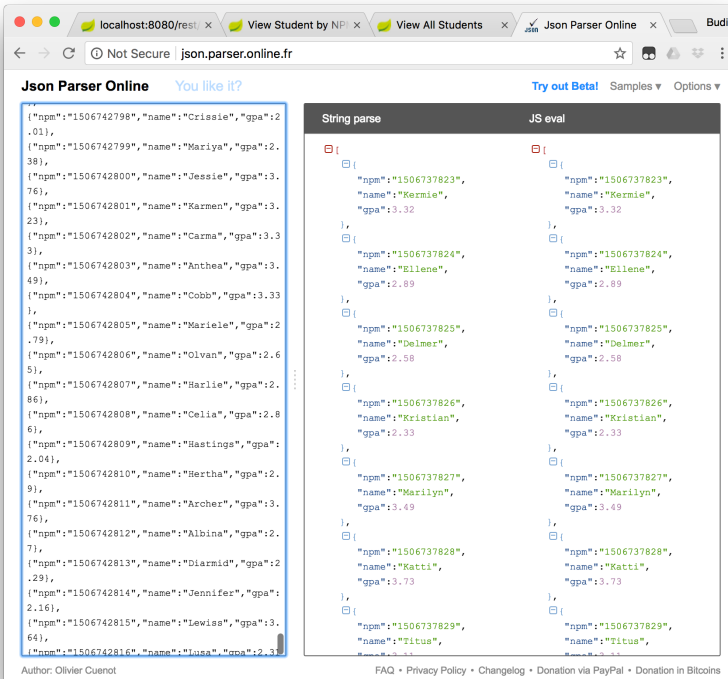
Nama : Budi Indrawan  
NPM : 1606954754

- Hasil yang didapatkan apabila mengakses <http://localhost:8080/rest/student/viewall>



```
[{"npm":"1506737823","name":"Kermie","gpa":3.32}, {"npm":"1506737824","name":"Ellene","gpa":2.89}, {"npm":"1506737825","name":"Delmer","gpa":2.58}, {"npm":"1506737826","name":"Kristian","gpa":2.33}, {"npm":"1506737827","name":"Marilyn","gpa":3.49}, {"npm":"1506737828","name":"Kattiti","gpa":3.73}, {"npm":"1506737829","name":"Titus","gpa":3.11}, {"npm":"1506737830","name":"Glenden","gpa":3.38}, {"npm":"1506737831","name":"Milt","gpa":3.39}, {"npm":"1506737832","name":"Henri","gpa":2.31}, {"npm":"1506737833","name":"Buffy","gpa":3.35}, {"npm":"1506737834","name":"Roda","gpa":2.2}, {"npm":"1506737835","name":"Petronille","gpa":2.61}, {"npm":"1506737836","name":"Kelley","gpa":3.27}, {"npm":"1506737837","name":"Morey","gpa":3.95}, {"npm":"1506737838","name":"Ronda","gpa":2.23}, {"npm":"1506737839","name":"Corbin","gpa":3.93}, {"npm":"1506737840","name":"Jessie","gpa":2.5}, {"npm":"1506737841","name":"Berri","gpa":3.9}, {"npm":"1506737842","name":"Lesley","gpa":3.0}, {"npm":"1506737843","name":"Flss","gpa":3.72}, {"npm":"1506737844","name":"Merell","gpa":3.15}, {"npm":"1506737845","name":"Ibbie","gpa":3.45}, {"npm":"1506737846","name":"Tony","gpa":2.69}, {"npm":"1506737847","name":"Klarika","gpa":2.83}, {"npm":"1506737848","name":"Brewster","gpa":2.62}, {"npm":"1506737849","name":"Zacherie","gpa":3.12}, {"npm":"1506737850","name":"Betsey","gpa":2.24}, {"npm":"1506737851","name":"Avery","gpa":3.38}, {"npm":"1506737852","name":"Gill","gpa":3.12}, {"npm":"1506737853","name":"Stephan","gpa":3.62}, {"npm":"1506737854","name":"Viole","gpa":3.91}, {"npm":"1506737855","name":"Caryl","gpa":3.83}, {"npm":"1506737856","name":"Nettie","gpa":2.52}, {"npm":"1506737857","name":"Emyle","gpa":3.69}, {"npm":"1506737858","name":"Bird","gpa":3.15}, {"npm":"1506737859","name":"Geno","gpa":2.19}, {"npm":"1506737860","name":"Saxon","gpa":2.91}, {"npm":"1506737861","name":"Julianna","gpa":2.41}, {"npm":"1506737862","name":"Marris","gpa":2.14}, {"npm":"1506737863","name":"Sheba","gpa":2.41}, {"npm":"1506737864","name":"Marcelline","gpa":3.67}, {"npm":"1506737865","name":"Aubrey","gpa":2.22}, {"npm":"1506737866","name":"Bliss","gpa":2.7}, {"npm":"1506737867","name":"Merry","gpa":2.32}, {"npm":"1506737868","name":"Portia","gpa":2.56}, {"npm":"1506737869","name":"Jewel","gpa":2.35}, {"npm":"1506737870","name":"Andris","gpa":3.38}, {"npm":"1506737871","name":"Nadiya","gpa":3.86}, {"npm":"1506737872","name":"Aggie","gpa":2.92}, {"npm":"1506737873","name":"Ramsay","gpa":3.6}, {"npm":"1506737874","name":"Saundra","gpa":3.28}, {"npm":"1506737875","name":"Sylas","gpa":2.43}, {"npm":"1506737876","name":"Clyde","gpa":3.04}, {"npm":"1506737877","name":"Modestine","gpa":2.83}, {"npm":"1506737878","name":"Max","gpa":2.53}, {"npm":"1506737879","name":"Chester","gpa":2.77}, {"npm":"1506737880","name":"Carl","gpa":3.01}, {"npm":"1506737881","name":"Milton","gpa":3.07}, {"npm":"1506737882","name":"Eugene","gpa":3.74}, {"npm":"1506737883","name":"Osmond","gpa":2.03}, {"npm":"1506737884","name":"Cele","gpa":2.01}, {"npm":"1506737885","name":"Kandy","gpa":3.3}, {"npm":"1506737886","name":"Zachary","gpa":2.71}, {"npm":"1506737887","name":"Onelda","gpa":3.12}, {"npm":"1506737888","name":"Dietrich","gpa":2.72}, {"npm":"1506737889","name":"Brucie","gpa":2.1}, {"npm":"1506737890","name":"Lucho","gpa":3.27}, {"npm":"1506737891","name":"Giorgia","gpa":2.05}, {"npm":"1506737892","name":"Taber","gpa":3.61}, {"npm":"1506737893","name":"Carlen","gpa":2.81}, {"npm":"1506737894","name":"Bell","gpa":2.58}, {"npm":"1506737895","name":"Natanial","gpa":2.21}, {"npm":"1506737896","name":"Dannie","gpa":3.43}, {"npm":"1506737897","name":"Cassandry","gpa":2.74}, {"npm":"1506737898","name":"Madeline","gpa":3.72}, {"npm":"1506737899","name":"Linoel","gpa":3.34}, {"npm":"1506737900","name":"Cori","gpa":2.39}, {"npm":"1506737901","name":"Baillie","gpa":2.79}, {"npm":"1506737902","name":"Suzy","gpa":3.85}, {"npm":"1506737903","name":"Merrin","gpa":3.06}, {"npm":"1506737904","name":"Gannie","gpa":2.93}, {"npm":"1506737905","name":"Alia","gpa":3.95}, {"npm":"1506737906","name":"Rutter","gpa":2.43}, {"npm":"1506737907","name":"Mela","gpa":3.4}, {"npm":"1506737908","name":"Conrado","gpa":3.78}, {"npm":"1506737909","name":"Isabella","gpa":2.77}, {"npm":"1506737910","name":"Lynett","gpa":2.36}, {"npm":"1506737911","name":"Wesley","gpa":3.33}, {"npm":"1506737912","name":"Darrel","gpa":2.57}, {"npm":"1506737913","name":"Corrina","gpa":3.38}, {"npm":"1506737914","name":"Lorry","gpa":2.8}, {"npm":"1506737915","name":"Jusa","gpa":2.33}
```

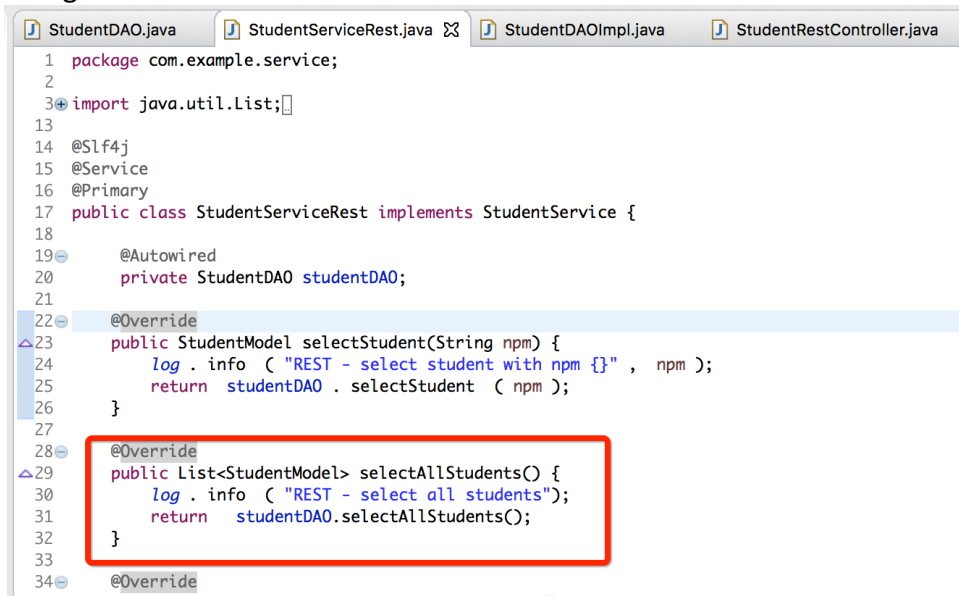
- Pembacaan data JSON



The screenshot shows the 'Json Parser Online' interface. On the left, a text area contains the raw JSON array of student data. On the right, there are two tabs: 'String parse' and 'JS eval'. The 'String parse' tab is active, displaying the JSON data in a structured, indented format. The 'JS eval' tab is also visible, showing the same data in a different format. The website footer includes the author's name 'Oliver Cuenot' and links for 'FAQ', 'Privacy Policy', 'Changelog', 'Donation via PayPal', and 'Donation in Bitcoins'.

Nama : Budi Indrawan  
NPM : 1606954754

- ❖ **LATIHAN 2** : Implementasikan *service consumer* untuk view all Students.
- Membuat method `selectAllStudents()` pada `StudentServiceRest.java` yang mengembalikan semua data student.



```
1 package com.example.service;
2
3 import java.util.List;
13
14 @Slf4j
15 @Service
16 @Primary
17 public class StudentServiceRest implements StudentService {
18
19     @Autowired
20     private StudentDAO studentDAO;
21
22     @Override
23     public StudentModel selectStudent(String npm) {
24         log.info ( "REST - select student with npm {}", npm );
25         return studentDAO . selectStudent ( npm );
26     }
27
28     @Override
29     public List<StudentModel> selectAllStudents() {
30         log . info ( "REST - select all students");
31         return studentDAO.selectAllStudents();
32     }
33
34     @Override
```

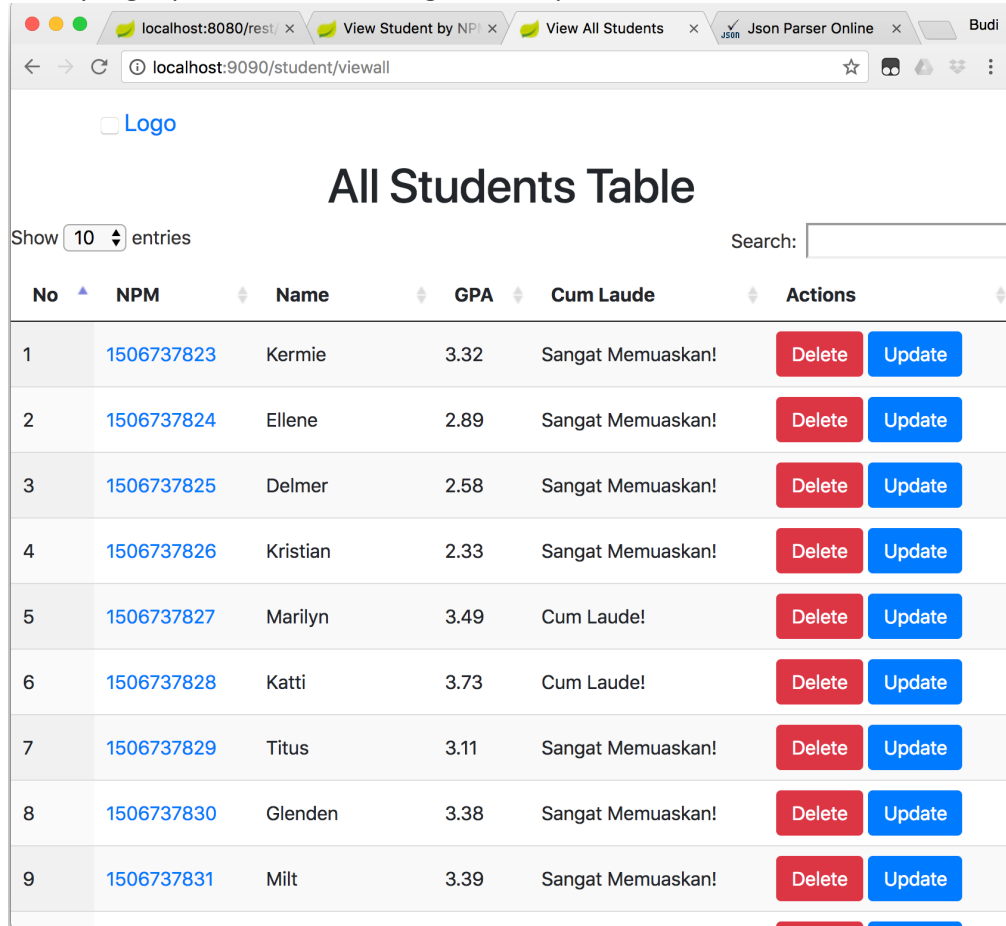
- Agar dapat menampilkan data student dalam bentuk list maka digunakan `ResponseEntity` untuk mengambil nilai object tersebut dalam bentuk list.



```
1 package com.example.dao;
2
3 import java.util.Arrays;
13
14 @Service
15 public class StudentDAOImpl implements StudentDAO {
16
17     @Autowired
18     private RestTemplate restTemplate;
19
20     @Override
21     public StudentModel selectStudent(String npm) {
22         StudentModel student = restTemplate . getForObject ( "http://localhost:8080/rest/student/view/" + npm, StudentModel . class );
23         return student;
24     }
25
26     @Override
27     public List<StudentModel> selectAllStudents() {
28         ResponseEntity<StudentModel[]> students = restTemplate.getForEntity("http://localhost:8080/rest/student/viewall", StudentModel[].class);
29         return Arrays.asList(students.getBody());
30     }
31
32 }
33
```

Nama : Budi Indrawan  
NPM : 1606954754

- Hasil yang diperoleh ketika mengakses <http://localhost:9090/student/viewall>



No	NPM	Name	GPA	Cum Laude	Actions
1	<a href="#">1506737823</a>	Kermie	3.32	Sangat Memuaskan!	<button>Delete</button> <button>Update</button>
2	<a href="#">1506737824</a>	Ellene	2.89	Sangat Memuaskan!	<button>Delete</button> <button>Update</button>
3	<a href="#">1506737825</a>	Delmer	2.58	Sangat Memuaskan!	<button>Delete</button> <button>Update</button>
4	<a href="#">1506737826</a>	Kristian	2.33	Sangat Memuaskan!	<button>Delete</button> <button>Update</button>
5	<a href="#">1506737827</a>	Marilyn	3.49	Cum Laude!	<button>Delete</button> <button>Update</button>
6	<a href="#">1506737828</a>	Katti	3.73	Cum Laude!	<button>Delete</button> <button>Update</button>
7	<a href="#">1506737829</a>	Titus	3.11	Sangat Memuaskan!	<button>Delete</button> <button>Update</button>
8	<a href="#">1506737830</a>	Glenden	3.38	Sangat Memuaskan!	<button>Delete</button> <button>Update</button>
9	<a href="#">1506737831</a>	Milt	3.39	Sangat Memuaskan!	<button>Delete</button> <button>Update</button>