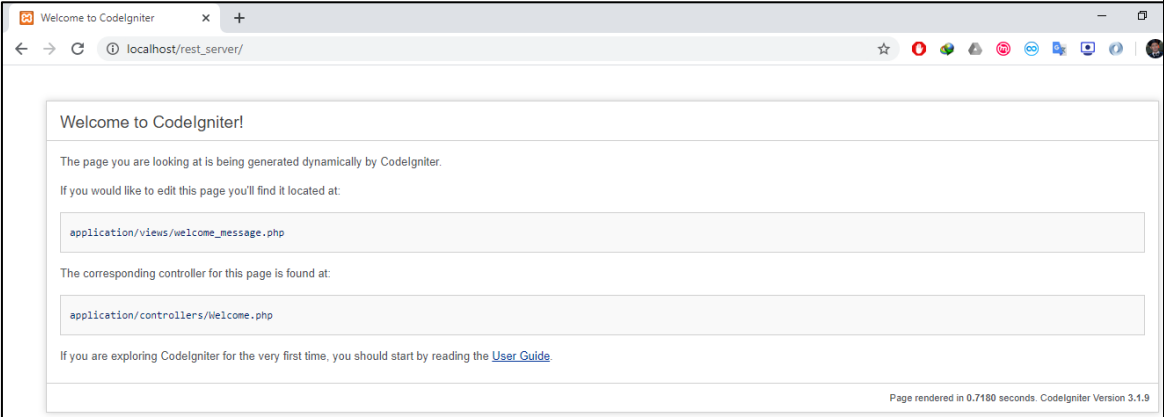


**PEMROGRAMAN WEB LANJUT**  
**JOBSHEET 8**



**Oleh:**  
**SUGENG PRASTIYO**  
**NIM. 1841720113**

**PROGRAM STUDI TEKNIK INFORMATIKA**  
**JURUSAN TEKNOLOGI INFORMASI**  
**POLITEKNIK NEGERI MALANG**  
**2020**

No	Uraian Jawaban Praktikum bagian 1(Instalasi library restfull codeigniter)
1	Silahkan unduh aplikasi Postman terlebih dahulu.
2	Pada praktikum kali ini kita akan menggunakan source codeigniter yang baru dan Library Rest Server. Untuk versi codeigniter yang digunakan adalah versi 3.1.9. sedangkan untuk Library Rest Server dapat di unduh pada link berikut :
3	<p>Extract codeigniter yang sudah di unduh dan letakkan pada folder httdocs. Beri nama Code-kedelapan dan coba akses di <a href="http://localhost/Code-kedelapan/">http://localhost/Code-kedelapan/</a>. Pastikan tampilan seperti gambar dibawah ini:</p> 
4	<p>Selanjutnya lakukan konfigurasi dasar pada <b>config.php</b> seperti berikut:</p> <pre data-bbox="302 1184 992 1549"> 24    25  */ 26  \$config['base_url'] = ''; 27 28  /*  37  */ 38  \$config['index_page'] = 'index.php'; 39 40  /* </pre> <p>Menjadi seperti berikut :</p> <pre data-bbox="302 1591 1279 1785"> 24    25  */ 26  \$config['base_url'] = 'http://localhost/rest_server/'; 27 28  /* </pre>

```

37  */
38  $config['index_page'] = '';
39
40  /*

```

Dan file **autoload.php**

```

60  */
61  $autoload['libraries'] = array();
62
63  /*

```

```

91  */
92  $autoload['helper'] = array();
93
94  /*

```

Menjadi

```

60  */
61  $autoload['libraries'] = array('database');
62
63  /*

```

```

91  */
92  $autoload['helper'] = array('url');
93
94  /*

```

Dan database.php seperti berikut:

```

76  $db['default'] = array(
77      'dsn' => '',
78      'hostname' => 'localhost',
79      'username' => 'root',
80      'password' => '',
81      'database' => 'rest_server',
82      'dbdriver' => 'mysqli',

```

- 5 Selanjutnya buatlah database **rest\_server** pada phpmyadmin dan masukkan data dummy. Untuk data dummy dapat dibuat sebagai berikut:

```

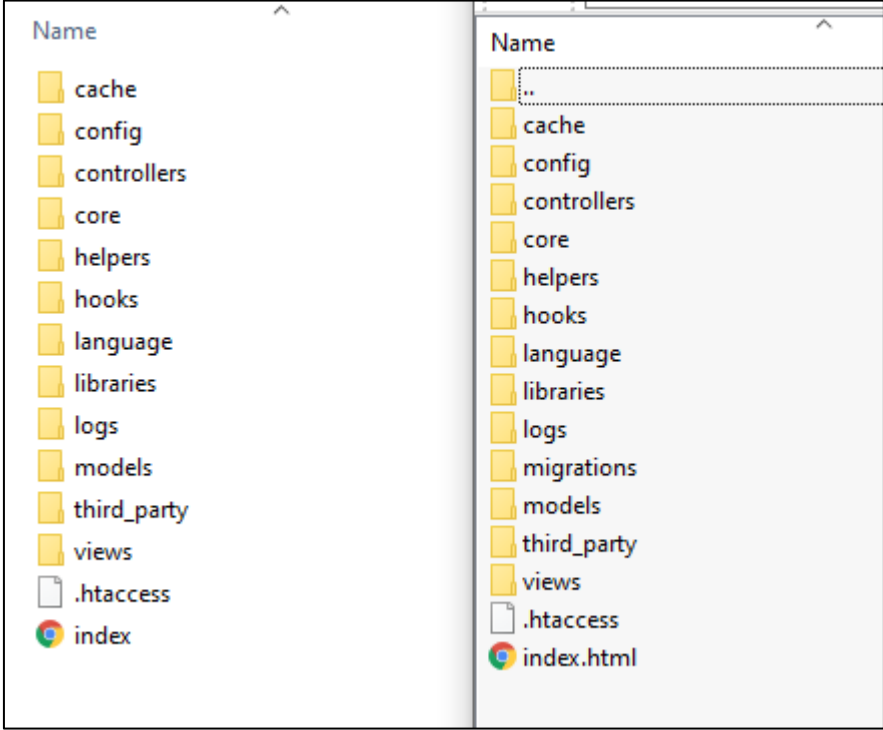
CREATE TABLE `mahasiswa` (
  `id` int(11) NOT NULL,
  `nrp` char(9) NOT NULL,
  `nama` varchar(250) NOT NULL,
  `email` varchar(250) DEFAULT NULL,
  `jurusan` varchar(64) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

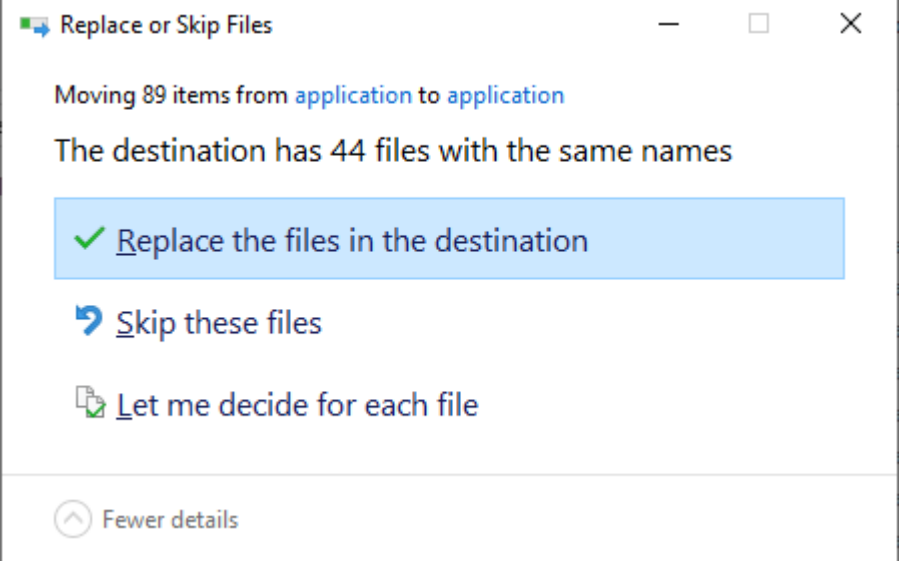
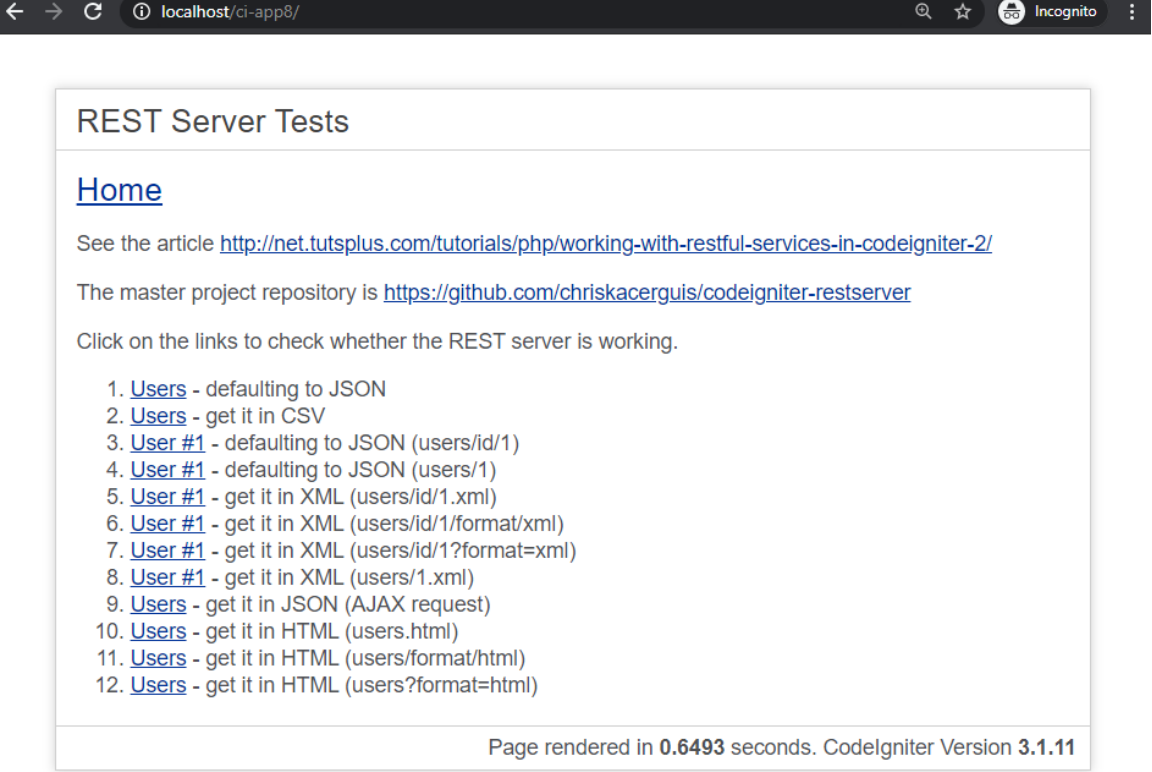
```

```

INSERT INTO `mahasiswa` (`id`, `nrp`, `nama`, `email`, `jurusan`) VALUES
(1, '043040001', 'Doddy Ferdiansyah', 'doy@gmail.com', 'Teknik Mesin'),

```

	<pre> (2, '023040123', 'Erik', 'erik@gmail.com', 'Teknik Industri'), (3, '043040321', 'Rommy Fauzi', 'rommy@gmail.com', 'Teknik Planologi'), (4, '033040023', 'Fajar Darmawan ', 'fajar@yahoo.com', 'Teknik Informatika '), (5, '113040321', 'Ferry Mulyanto', 'ferry@yahoo.com', 'Manajemen');  ALTER TABLE `mahasiswa`   ADD PRIMARY KEY (`id`);  ALTER TABLE `mahasiswa`   MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=33; </pre>
6	<p>Selanjutnya extract file Library Rest Server, buka folder <b>application</b> dan duplikasi file dan folder tersebut kedalam folder <b>application</b> pada project <b>Code-kedelapan</b>. Klik <b>Replace</b> files jika ada notifikasi duplikat.</p> 

	 <p>Setelah proses duplikasi file selesai, pastikan file-file yang kita sudah konfigurasi di awal tetap terkonfigurasi.</p>
7	<p>Selanjutnya buka project <a href="http://localhost/Code-kedelapan/">http://localhost/Code-kedelapan/</a>. Jika berhasil, tampilan akan berubah menjadi seperti ini:</p> 
8	Tahap instalasi Library Rest Server selesai
<b>NO</b>	<b><u>Praktikum – Bagian 2: Membuat Aplikasi CRUD Restfull Codeigniter</u></b>
1	Buatlah file <b>Mahasiswa.php</b> pada <b>controller/api</b> . Selanjutnya tuliskan kode berikut:

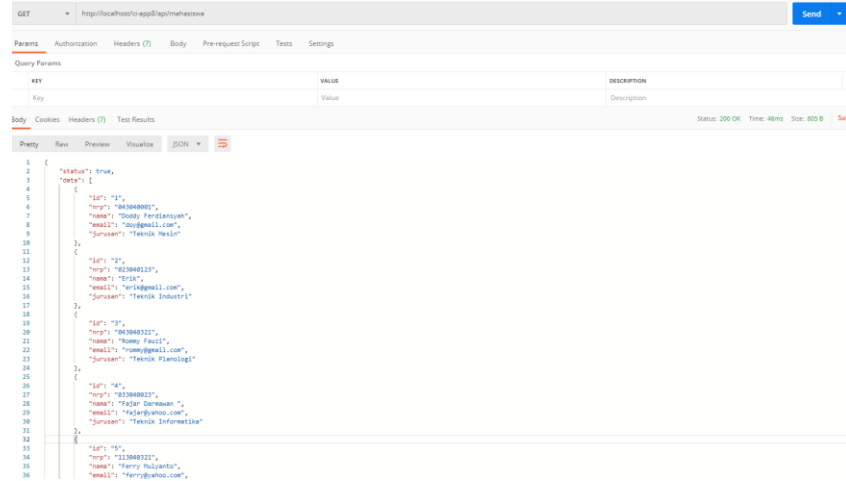
	<pre> application &gt; controllers &gt; api &gt; Mahasiswa.php &gt; Mahasiswa 1  &lt;?php 2 3  use Restserver\Libraries\REST_Controller; 4 5  defined('BASEPATH') or exit('No direct script access allowed'); 6 7  require APPPATH . 'libraries/REST_Controller.php'; 8  require APPPATH . 'libraries/Format.php'; 9 10 class Mahasiswa extends REST_Controller 11 { 12     public function __construct() 13     { 14         parent::__construct(); 15         \$this-&gt;load-&gt;model('Mahasiswa_model', 'mahasiswa'); 16     } 17     public function index_get() 18     { 19         \$id = \$this-&gt;get('id'); 20         if (\$id === null) { 21             \$mahasiswa = \$this-&gt;mahasiswa-&gt;getMahasiswa(); 22         } else { 23             \$mahasiswa = \$this-&gt;mahasiswa-&gt;getMahasiswa(\$id); 24         } 25         if (\$mahasiswa) { 26             \$this-&gt;response([ 27                 'status' =&gt; true, 28                 'data' =&gt; \$mahasiswa 29             ], REST_Controller::HTTP_OK); 30         } else { 31             \$this-&gt;response([ 32                 'status' =&gt; false, 33                 'message' =&gt; 'id not found' 34             ], REST_Controller::HTTP_NOT_FOUND); 35         } 36     } 37 } 38 </pre>
--	---

2 Selanjutnya Buatlah file **Mahasiswa\_model.php** pada model. Selanjutnya tuliskan kode berikut:

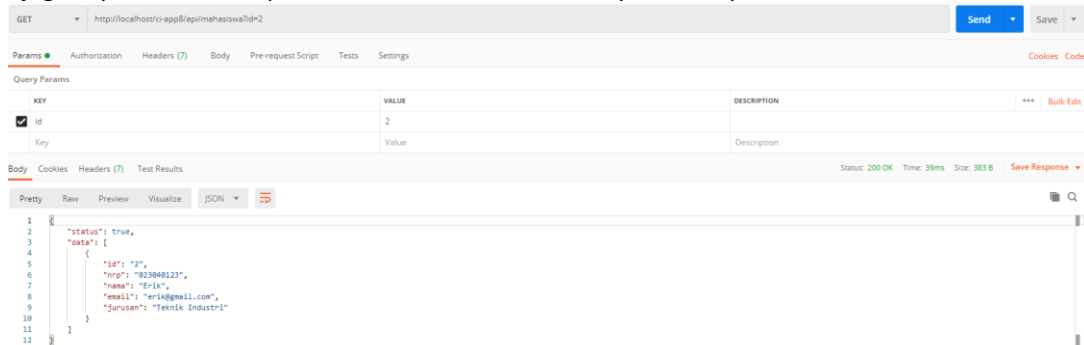
	<pre> application &gt; models &gt; Mahasiswa_model.php &gt; Mahasiswa_model &gt; getMahasiswa 1  &lt;?php 2 3  class Mahasiswa_model extends CI_model 4  { 5      public function getMahasiswa(\$id = null) 6      { 7          if (\$id === null) { 8              return \$this-&gt;db-&gt;get('mahasiswa')-&gt;result_array(); 9          } else { 10             return \$this-&gt;db-&gt;get_where('mahasiswa', ['id' =&gt; \$id])-&gt;result_array(); 11         } 12     } 13 } 14 </pre>
--	--

3

Buka aplikasi postman, masukkan alamat <http://localhost/ci-app8/api/mahasiswa> dengan metode **GET**. dan jika berhasil maka tampilannya akan seperti ini:



Kita juga dapat melakukan pencarian berdasarkan id, hasilnya akan seperti ini:



4

Untuk fungsi lain **Create**, **Update** dan **Delete**. Lanjutkan kode berikut pada file **mahasiswa.php**:

```

41 public function index_delete()
42 {
43     $id = $this->delete('id');
44     if ($id === null) {
45         $this->response([
46             'status' => false,
47             'message' => 'provide an id!'
48         ], REST_Controller::HTTP_BAD_REQUEST);
49     } else {
50         if ($this->mahasiswa->deleteMahasiswa($id) > 0) {
51             //ok
52             $this->response([
53                 'status' => true,
54                 'id' => $id,
55                 'message' => 'deleted.'
56             ], REST_Controller::HTTP_OK);
57         } else {
58             //id not found
59             $this->response([
60                 'status' => false,
61                 'message' => 'id not found!'
62             ], REST_Controller::HTTP_BAD_REQUEST);
63         }
64     }
65 }
66 public function index_post()
67 {
68     $data = [
69         'nrp' => $this->post('nrp'),
70         'nama' => $this->post('name'),
71         'email' => $this->post('email'),
72         'jurusan' => $this->post('jurusan'),
73     ];
74     if ($this->mahasiswa->createMahasiswa($data) > 0) {
75         $this->response([
76             'status' => true,
77             'message' => 'new mahasiswa has been created'
78         ], REST_Controller::HTTP_CREATED);
79     } else {
80         // id not found
81         $this->response([
82             'status' => false,
83             'message' => 'failde to create new data!'
84         ], REST_Controller::HTTP_BAD_REQUEST);
85     }
86 }

```



		<pre> 87     public function index_put() 88     { 89         \$id = \$this-&gt;put('id'); 90         \$data = [ 91             'nrp' =&gt; \$this-&gt;put('nrp'), 92             'nama' =&gt; \$this-&gt;put('nama'), 93             'email' =&gt; \$this-&gt;put('email'), 94             'jurusan' =&gt; \$this-&gt;put('jurusan'), 95         ]; 96 97         if (\$this-&gt;mahasiswa-&gt;updateMahasiswa(\$data, \$id) &gt; 0) { 98             \$this-&gt;response([ 99                 'status' =&gt; true, 100                 'message' =&gt; 'data mahasiswa has been updated' 101             ], REST_Controller::HTTP_OK); 102         } else { 103             //id not found 104             \$this-&gt;response([ 105                 'status' =&gt; false, 106                 'message' =&gt; 'failed to update data!' 107             ], REST_Controller::HTTP_BAD_REQUEST); 108         } 109     } 110 } </pre>
5	Dan lanjutkan kode pada file <b>Mahasiswa_model.php</b> :	<pre> application &gt; models &gt; Mahasiswa_model.php &gt; Mahasiswa_model &gt; updateMahasiswa 1  &lt;?php 2 3  defined('BASEPATH') or exit('No direct script access allowed'); 4 5  class Mahasiswa_model extends CI_Model 6  { 7 8      public function getMahasiswa(\$id = null) 9      { 10         if (\$id === null) { 11             return \$this-&gt;db-&gt;get('mahasiswa')-&gt;result_array(); 12         } else { 13             return \$this-&gt;db-&gt;get_where('mahasiswa', ['id' =&gt; \$id])-&gt;result_array(); 14         } 15     } 16     public function deleteMahasiswa(\$id) 17     { 18         \$this-&gt;db-&gt;delete('mahasiswa', ['id' =&gt; \$id]); 19         return \$this-&gt;db-&gt;affected_rows(); 20     } 21     public function createMahasiswa(\$data) 22     { 23         \$this-&gt;db-&gt;insert('mahasiswa', \$data); 24         return \$this-&gt;db-&gt;affected_rows(); 25     } 26     public function updateMahasiswa(\$data, \$id) 27     { 28         \$this-&gt;db-&gt;update('mahasiswa', \$data, ['id' =&gt; \$id]); 29         return \$this-&gt;db-&gt;affected_rows(); 30     } 31 } 32 </pre>
6	Selanjutnya lakukan operasi <b>Create</b> data pada aplikasi postman, untuk operasi <b>Create</b> , <b>Update</b> dan <b>Delete</b> , arahkan parameter ke <b>Body</b> dan <b>x-www-form-urlencoded</b> seperti berikut:	

GET

Params Authorization Headers (7) **Body** Pre-request Script Tests Settings

☐ none ☐ form-data ☒ x-www-form-urlencoded ☐ raw ☐ binary ☐ GraphQL

KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> nrp	0403070028	
<input checked="" type="checkbox"/> nama	Joko Anwar	
<input checked="" type="checkbox"/> email	jk_anwar@gmail.com	
<input checked="" type="checkbox"/> jurusan	Teknik Kimia	
Key	Value	Description

POST

Params Authorization Headers (9) **Body** Pre-request Script Tests Settings Cookies Code

☐ none ☐ form-data ☒ x-www-form-urlencoded ☐ raw ☐ binary ☐ GraphQL

KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/> nrp	0403070028			
<input checked="" type="checkbox"/> nama	Joko Anwar			
<input checked="" type="checkbox"/> email	jk_anwar@gmail.com			
<input checked="" type="checkbox"/> jurusan	Teknik Kimia			
Key	Value	Description		

Body Cookies Headers (7) Test Results Status: 201 Created Time: 57ms Size: 325 B Save Response

Pretty Raw Preview Visualize JSON

```

1 {
2   "status": true,
3   "message": "new mahasiswa has been created"
4 }

```

7 Lakukan juga operasi **Update** data pada aplikasi postman, masukkan parameter id seperti berikut:

PUT

Params Authorization Headers (9) **Body** Pre-request Script Tests Settings Cookies Code

☐ none ☐ form-data ☒ x-www-form-urlencoded ☐ raw ☐ binary ☐ GraphQL

KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/> nrp	0403070028			
<input checked="" type="checkbox"/> nama	Joko Anwar Sahid			
<input checked="" type="checkbox"/> email	jk_anwar@gmail.com			
<input checked="" type="checkbox"/> jurusan	Teknik Kimia			
<input checked="" type="checkbox"/> id	34			
Key	Value	Description		

Body Cookies Headers (7) Test Results Status: 200 OK Time: 51ms Size: 321 B Save Response

Pretty Raw Preview Visualize JSON

```

1 {
2   "status": true,
3   "message": "data mahasiswa has been updated"
4 }

```

8 Lakukan juga operasi **Delete** data pada aplikasi postman, parameter yang dimasukkan cukup id saja, seperti berikut:

DELETE

http://localhost/ci-app8/api/mahasiswa

Send

Save

ParamsAuthorizationHeaders (9)BodyPre-request ScriptTestsSettingsCookiesCode

none

form-data

x-www-form-urlencoded

raw

binary

GraphQL

	KEY	VALUE	DESCRIPTION		Bulk Edit
<input type="checkbox"/>	nrp	0403070028			
<input type="checkbox"/>	nama	Joko Anwar			
<input type="checkbox"/>	email	jk_anwar@gmail.com			
<input type="checkbox"/>	jurusan	Teknik Informatika			
<input checked="" type="checkbox"/>	id	36			
	Key	Value	Description		

BodyCookiesHeaders (7)Test ResultsStatus: 200 OKTime: 51msSize: 308 BSave Response

PrettyRawPreviewVisualizeJSON

1

2

3

4

5

"status": true,

"id": "36",

"message": "deleted."