

# **LAPORAN JOOBSHEET 8**

(Implementasi Restful API)

**Disusun sebagai**

**MATA KULIAH :**

**Pemrograman Web Lanjut**



**TI-2B**

**OLEH**

**FACKO ELLANDA (1841720154)**

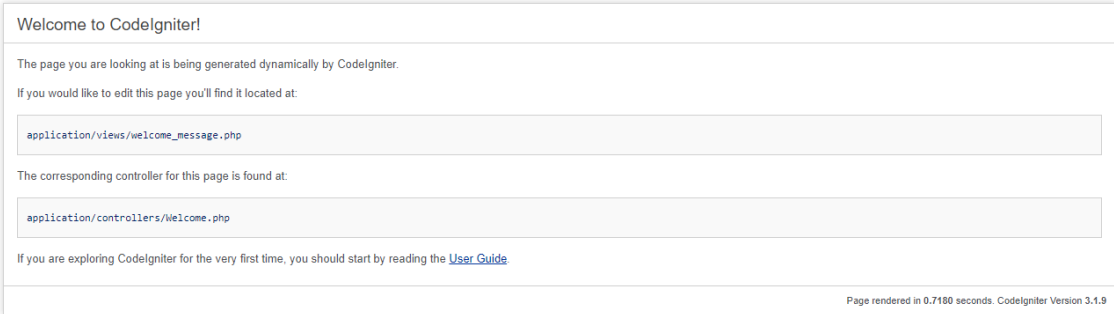
**PROGAM STUDI D-IV TEKNIK INFORMATIKA**

**JURUSAN TEKNOLOGI INFORMASI**

**POLITEKNIK NEGERI MALANG**

**2020**

## Praktikum – Bagian 1: Instalasi library restful codeigniter

Langkah	Keterangan
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 Codekedelapan dan coba akses di <a href="http://localhost/ci/codekedelapan/">http://localhost/ci/codekedelapan/</a>. Pastikan tampilan seperti gambar dibawah ini:</p> 
4	<p>Selanjutnya lakukan konfigurasi dasar pada <b>config.php</b> seperti berikut:</p> <pre><code>\$config['base_url'] = 'http://localhost/ci/codekedelapan/';  /*  -----    Index File  -----    Typically this will be your index.php file, unless you've renamed it to   something else. If you are using mod_rewrite to remove the page set this   variable so that it is blank.   */ \$config['index_page'] = '';</code></pre> <p>Dan file <b>autoload.php</b></p> <pre><code>\$autoload['libraries'] = array('database');  \$autoload['helper'] = array('url');</code></pre>

Dan database.php seperti berikut:

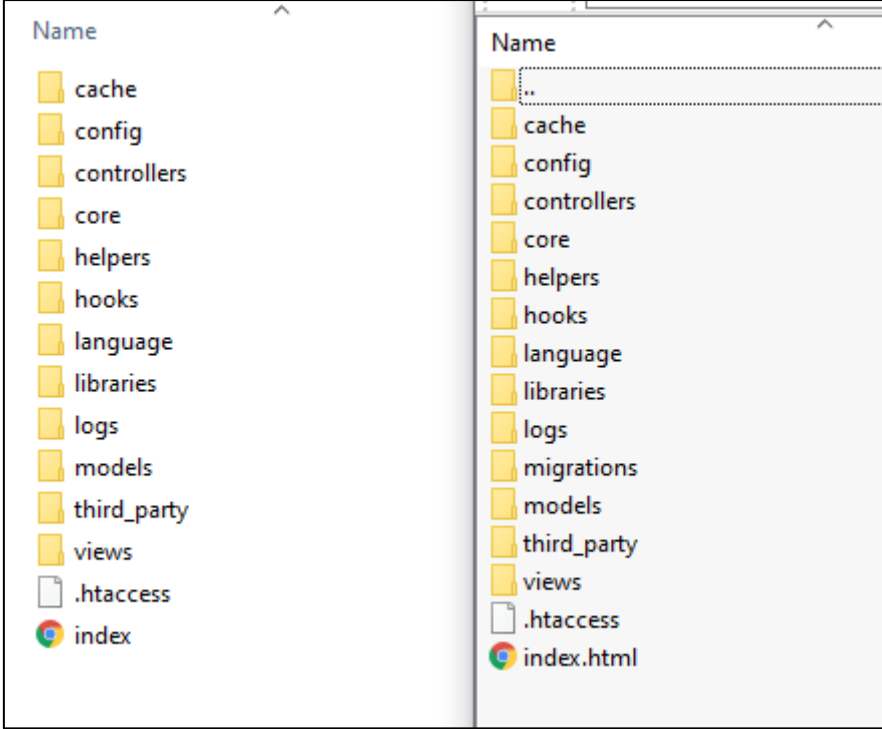
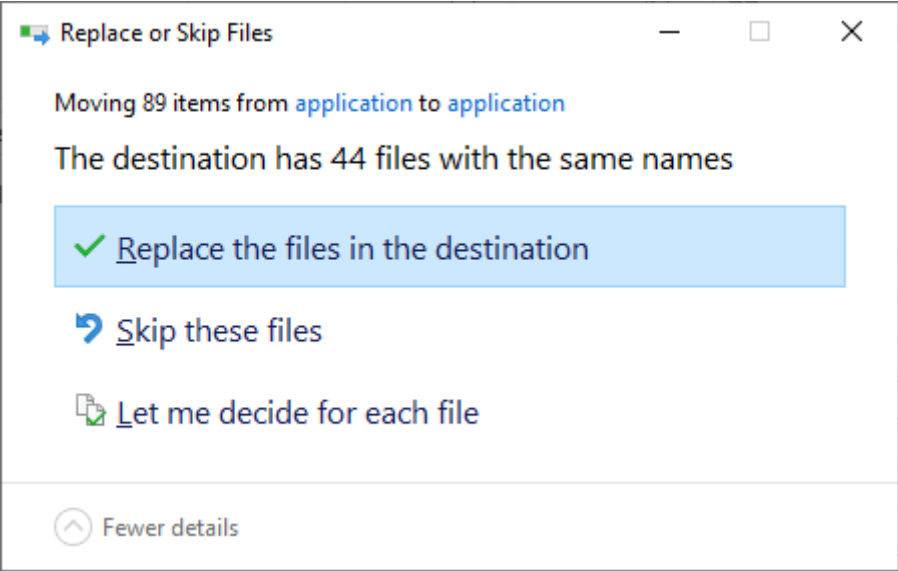
```
$db['default'] = array(
    'dsn' => '',
    'hostname' => 'localhost',
    'username' => 'root',
    'password' => '',
    'database' => 'rest_server',
    'dbdriver' => 'mysqli',
    'dbprefix' => '',
    'pconnect' => FALSE,
    'db_debug' => (ENVIRONMENT !== 'production'),
    'cache_on' => FALSE,
    'cachedir' => '',
    'char_set' => 'utf8',
    'dbcollat' => 'utf8_general_ci',
    'swap_pre' => '',
    'encrypt' => FALSE,
    'compress' => FALSE,
    'stricton' => FALSE,
    'failover' => array(),
    'save_queries' => TRUE
```

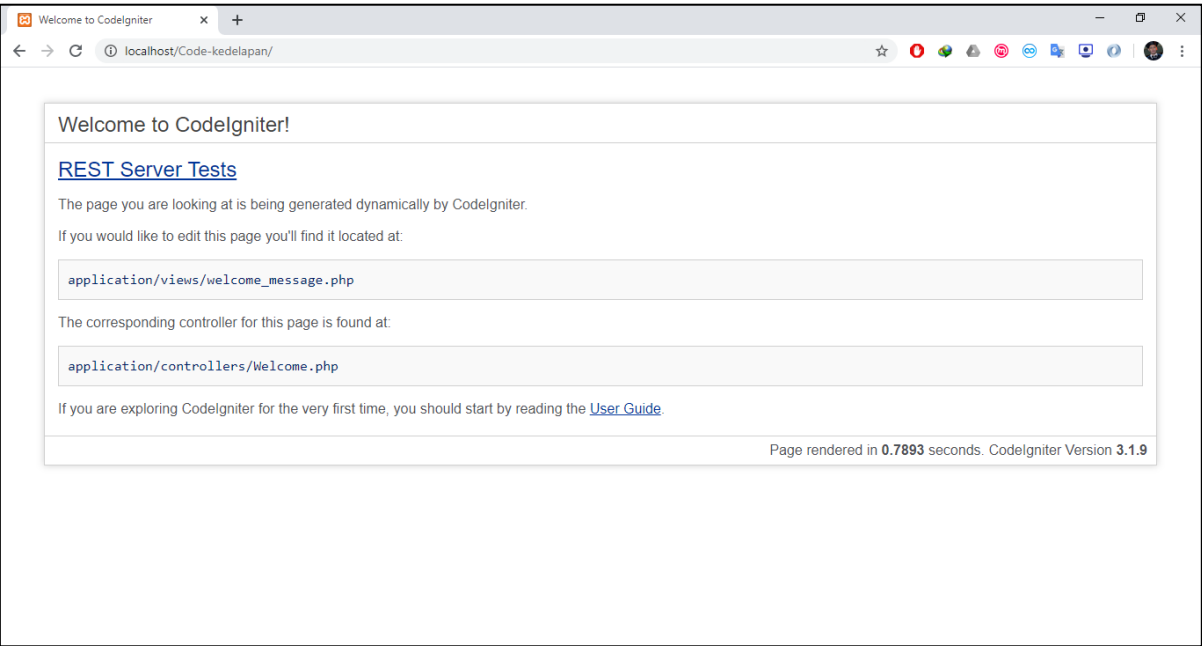
- 5 Selanjutnya buatlah database **rest\_server** pada phpmyadmin dan masukkan data dummy. Untuk data dummy dapat dibuat sebagai berikut:

```
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'),
(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`);
```

	<pre>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/ci/codekedelapan/">http://localhost/ci/codekedelapan/</a>. Jika berhasil, tampilan akan berubah menjadi seperti ini:</p>

	
8	Tahap instalasi Library Rest Server selesai

## Praktikum – Bagian 2: Membuat Aplikasi CRUD Restful Codeigniter

Langkah	Keterangan
1	Buatlah file <b>Mahasiswa.php</b> pada <b>controller/api</b> . Selanjutnya tuliskan kode berikut:

```

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

```

2

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

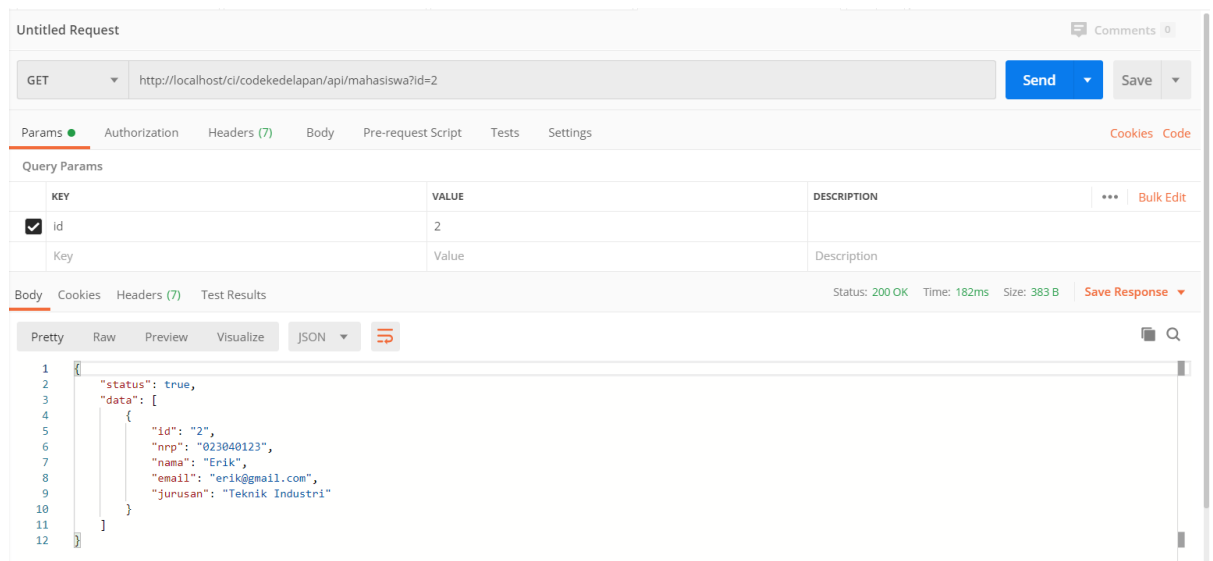
```

application > models > Mahasiswa_model.php
1  <?php
2
3  defined('BASEPATH') OR exit('No direct script access allowed');
4
5  class Mahasiswa_model extends CI_Model {
6
7     public function getMahasiswa($id=null){
8         if ($id === null) {
9             return $this->db->get('mahasiswa')->result_array();
10         } else {
11             return $this->db->get_where('mahasiswa', ['id' => $id])->result_array();
12         }
13     }
14
15 }

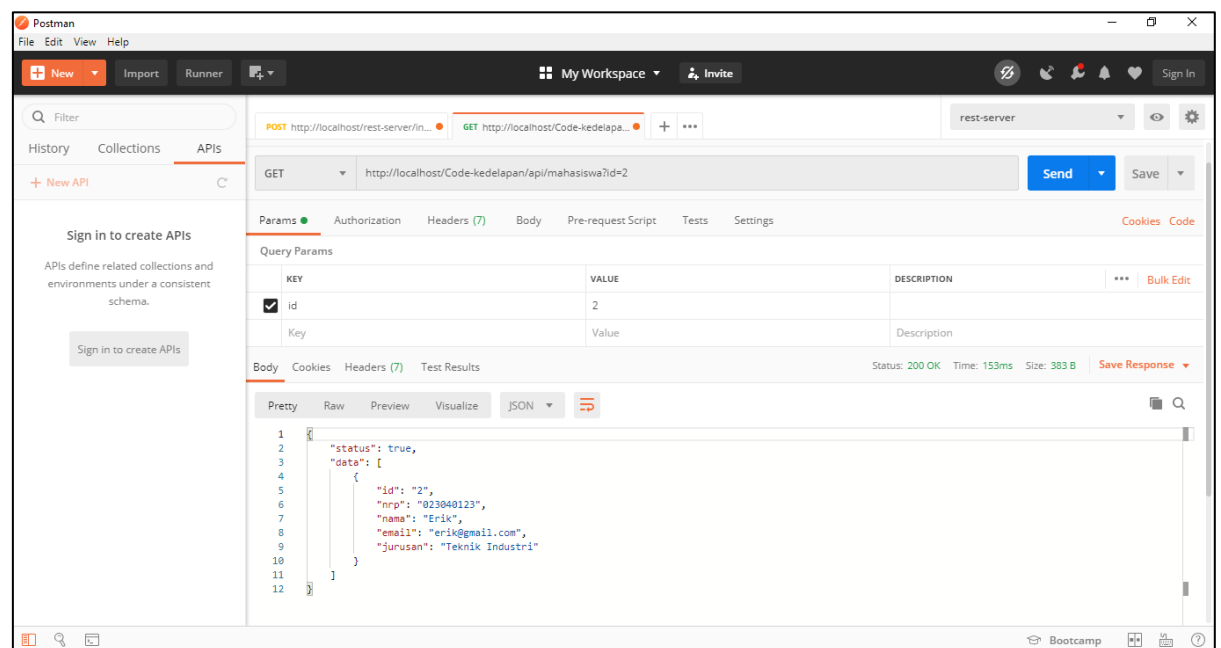
```

3

Buka aplikasi postman, masukkan alamat <http://localhost/ci/codekedelapan/api/mahasiswa?id=2> 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**:

```

39     function index_delete() {
40         $id = $this->delete('id');
41         if ($id === null) {
42             $this->response([
43                 'status' => false,
44                 'message' => 'provide an id'
45             ], REST_Controller::HTTP_BAD_REQUEST);
46         } else {
47             if ($this->mahasiswa->deleteMahasiswa($id) > 0) {
48                 $this->response([
49                     'status' => true,
50                     'id' => $id,
51                     'message' => 'deleted'
52                 ], REST_Controller::HTTP_OK);
53             } else {
54                 $this->response([
55                     'status' => false,
56                     'message' => 'id not found!'
57                 ], REST_Controller::HTTP_BAD_REQUEST);
58             }
59         }
60     }

```

```

63     public function index_post(){
64         $data = [
65             'nrp' => $this->post('nrp'),
66             'nama' => $this->post('nama'),
67             'email' => $this->post('email'),
68             'jurusan' => $this->post('jurusan'),
69         ];
70
71         if ($this->mahasiswa->createMahasiswa($data) > 0) {
72             $this->response([
73                 'status' => true,
74                 'message' => 'new mahasiswa has been created'
75             ], REST_Controller::HTTP_CREATED);
76         } else {
77             $this->response([
78                 'status' => false,
79                 'message' => 'failed to create new data'
80             ], REST_Controller::HTTP_BAD_REQUEST);
81         }
82     }
83

```



5

Dan lanjutkan kode pada file **Mahasiswa\_model.php**:

```

15  public function deleteMahasiswa($id){
16      $this->db->delete('mahasiswa', ['id' => $id]);
17      return $this->db->affected_rows();
18  }
19
20
21  public function createMahasiswa($data){
22      $this->db->insert('mahasiswa', $data);
23      return $this->db->affected_rows();
24  }
25
26
27  public function updateMahasiswa($data, $id){
28      $this->db->update('mahasiswa', $data, ['id' => $id]);
29      return $this->db->affected_rows();
30  }
31
32
33  }

```

6

Selanjutnya lakukan operasi **Create** data pada aplikasi postman, untuk operasi **Create**, **Update** dan **Delete**, arahkan parameter ke **Body** dan **x-www-form-urlencoded** seperti berikut:

Params		Authorization		Headers (9)		Body ●	
		<input type="radio"/> none <input type="radio"/> form-data <input checked="" type="radio"/> x-www-form-urlencoded					
	KEY						
<input checked="" type="checkbox"/>	nrp						
<input checked="" type="checkbox"/>	nama						
<input checked="" type="checkbox"/>	email						
<input checked="" type="checkbox"/>	jurusan						

Untitled Request Comments 0

POST http://localhost/ci/codekedelapan/api/mahasiswa Send Save

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
<input checked="" type="checkbox"/> nrp	1841720154	
<input checked="" type="checkbox"/> nama	Facko Ellanda	
<input checked="" type="checkbox"/> email	fackoellanda89@gmail.com	
<input checked="" type="checkbox"/> jurusan	Teknik Kimia	

Body Cookies Headers (7) Test Results Status: 201 Created Time: 166ms Size: 325 B Save Response

Pretty Raw Preview Visualize JSON

```

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

```

Bootcamp Build Browse

GET http://localhost/ci/codekedelapan/api/mahasiswa Send Save

Pretty Raw Preview Visualize JSON

```

20 {
21   "nrp": "043040321",
22   "nama": "Rommy Fauzi",
23   "email": "rommy@gmail.com",
24   "jurusan": "Teknik Planologi"
25 },
26 {
27   "id": "4",
28   "nrp": "033040023",
29   "nama": "Fajar Darawan ",
30   "email": "fajar@yahoo.com",
31   "jurusan": "Teknik Informatika"
32 },
33 {
34   "id": "5",
35   "nrp": "113040321",
36   "nama": "Ferry Mulyanto",
37   "email": "ferry@yahoo.com",
38   "jurusan": "Manajemen"
39 },
40 {
41   "id": "33",
42   "nrp": "040307002",
43   "nama": "Joko Anwar",
44   "email": "jk_anwar@gmail.com",
45   "jurusan": "Teknik Informatika"
46 },
47 {
48   "id": "34",
49   "nrp": "184172015",
50   "nama": "Facko Ellanda",
51   "email": "fackoellanda89@gmail.com",
52   "jurusan": "Teknik Kimia"
53 }
54 ]

```

Bootcamp Build Browse

Nama sudah masuk didatabase

7

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

Untitled Request Comments 0

PUT http://localhost/ci/codekedelapan/api/mahasiswa Send Save

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	1841720154			
<input checked="" type="checkbox"/> nama	Facko Ellanda			
<input checked="" type="checkbox"/> email	fackoellanda89@gmail.com			
<input checked="" type="checkbox"/> jurusan	Teknik Informatika			
<input checked="" type="checkbox"/> id	34			
Key	Value	Description		

Body Cookies Headers (7) Test Results Status: 200 OK Time: 175ms Size: 320 B Save Response

Pretty Raw Preview Visualize JSON ⌵

```

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

```

Bootcamp Build Browse ⌵ ?

data didatabase berhasil diupdate

8

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

Untitled Request Comments 0

DELETE http://localhost/ci/codekedelapan/api/mahasiswa Send Save

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"/> id	34			
Key	Value	Description		

Body Cookies Headers (7) Test Results Status: 200 OK Time: 173ms Size: 307 B Save Response

Pretty Raw Preview Visualize JSON ⌵

```

1 {
2   "status": true,
3   "id": "34",
4   "message": "deleted"
5 }

```

Bootcamp Build Browse ⌵ ?

Berhasil didelete