



Nama : Agsal Fairrohmad A.P
No : 04/TI2A

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 : https://drive.google.com/drive/folders/10YaCrqrgG2M1Zk0r-trln0hROP9q0hpi?usp=sharing
3	Extract codeigniter yang sudah di unduh dan letakkan pada folder httdocs. Beri nama Code-kedelapan dan coba akses di http://localhost/Code-kedelapan/ . P
4	<p>Selanjutnya lakukan konfigurasi dasar pada config.php seperti berikut:</p> <pre>24 25 */ 26 \$config['base_url'] = ''; 27 28 /* 37 */ 38 \$config['index_page'] = 'index.php'; 39 40 /*</pre> <p>Menjadi seperti berikut :</p> <pre>24 25 */ 26 \$config['base_url'] = 'http://localhost/rest_server/'; 27 28 /* 37 */ 38 \$config['index_page'] = ''; 39 40 /*</pre> <p>Dan file autoload.php</p> <pre>60 */ 61 \$autoload['libraries'] = array(); 62 63 /*</pre>

```

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;

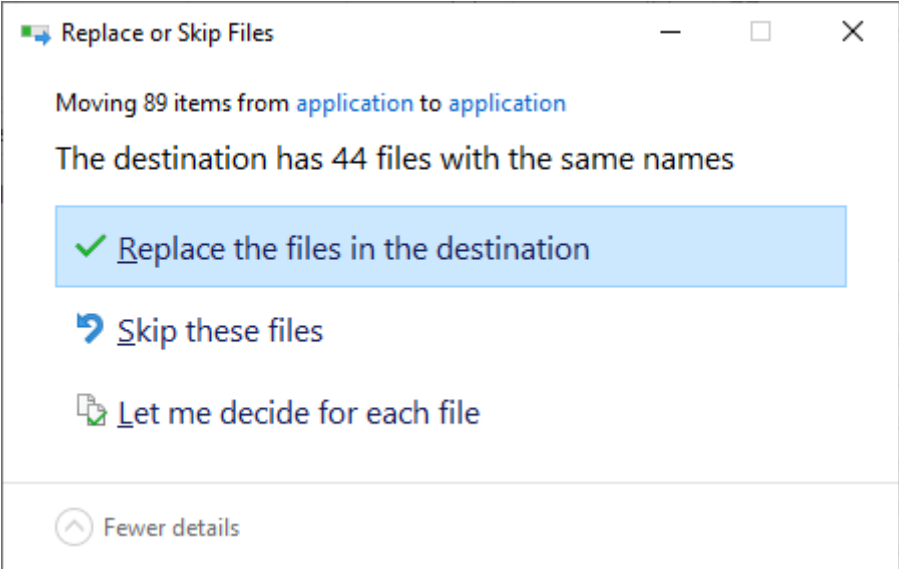
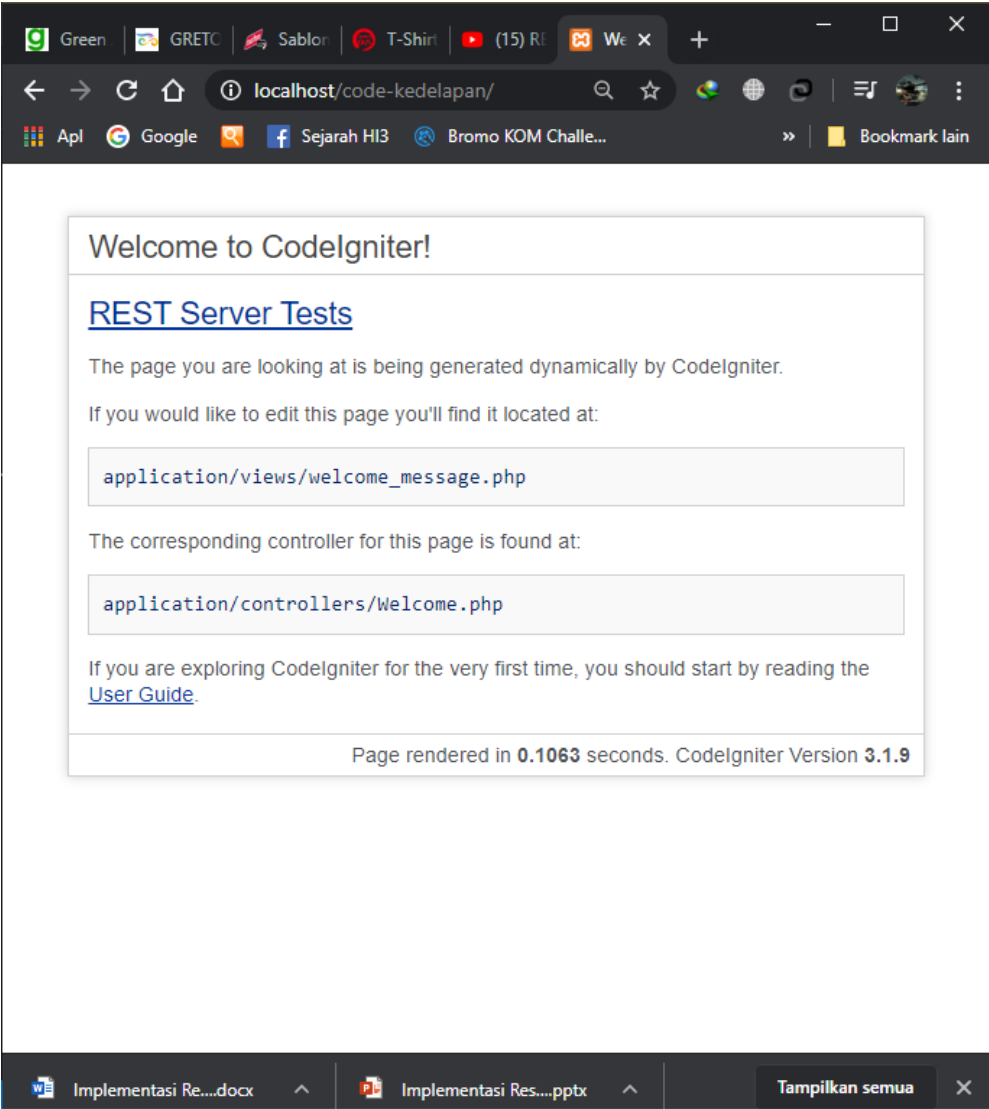
INSERT INTO `mahasiswa` (`id`, `nrp`, `nama`, `email`, `jurusan`) VALUES
(1, '043040001', 'Doddy Ferdiansyah', 'doy@gmail.com', 'Teknik Mesin'),
(2, '023040123', 'Erik', 'erik@gmail.com', 'Teknik Industri'),
(3, '043040321', 'Rommy Fauzi', 'rommy@gmail.com', 'Teknik Planologi'),
(4, '033040023', 'Fajar 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;

```

- 6 Selanjutnya extract file Library Rest Server, buka folder **application** dan duplikasi file dan folder tersebut kedalam folder **application** pada project **Code-kedelapan**. Klik **Replace** files jika ada notifikasi duplikat.

	 <p>Setelah proses duplikasi file selesai, pastikan file-file yang kita sudah konfigurasi di awal tetap terkonfigurasi.</p>
7	<p>Selanjutnya buka project http://localhost/Code-kedelapan/. 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	<p>Buatlah file Mahasiswa.php pada controller/api. Selanjutnya tuliskan kode berikut:</p>  <pre> 1 defined('BASEPATH') or exit('No direct script access allowed') 2 3 require APPPATH . 'libraries/REST_Controller.php'; 4 require APPPATH . 'libraries/Format.php'; 5 6 class Mahasiswa extends REST_Controller 7 { 8 9 public function __construct() 10 { 11 parent::__construct(); 12 \$this->load->model(['Mahasiswa_model', 'mahasiswa']); 13 } 14 15 public function index_get() 16 { 17 \$id = \$this->get('id'); 18 if (\$id === null) { 19 \$mahasiswa = \$this->mahasiswa->getMahasiswa(); 20 } else { 21 \$mahasiswa = \$this->mahasiswa->getMahasiswa(\$id); 22 } 23 24 if (\$mahasiswa) { 25 \$this->response([26 'status' => true, 27 'data' => \$mahasiswa 28], REST_Controller::HTTP_OK); 29 } else { 30 \$this->response([31 'status' => false, 32 'message' => 'id not found' 33], REST_Controller::HTTP_NOT_FOUND); 34 } 35 } 36 </pre>
2	<p>Selanjutnya Buatlah file Mahasiswa_model.php pada model. Selanjutnya tuliskan kode berikut:</p>

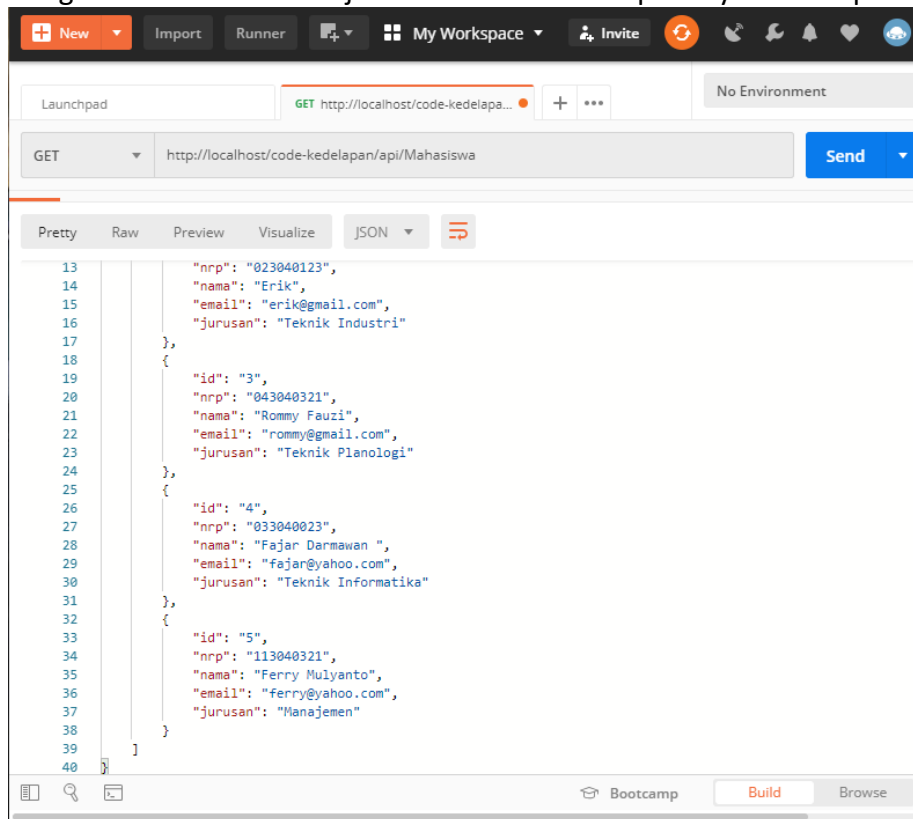
```

1  <?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->db->get('mahasiswa')->result_array()
12         } else {
13             return $this->db->get_where('mahasiswa', ['id' =>
14         }
15     }

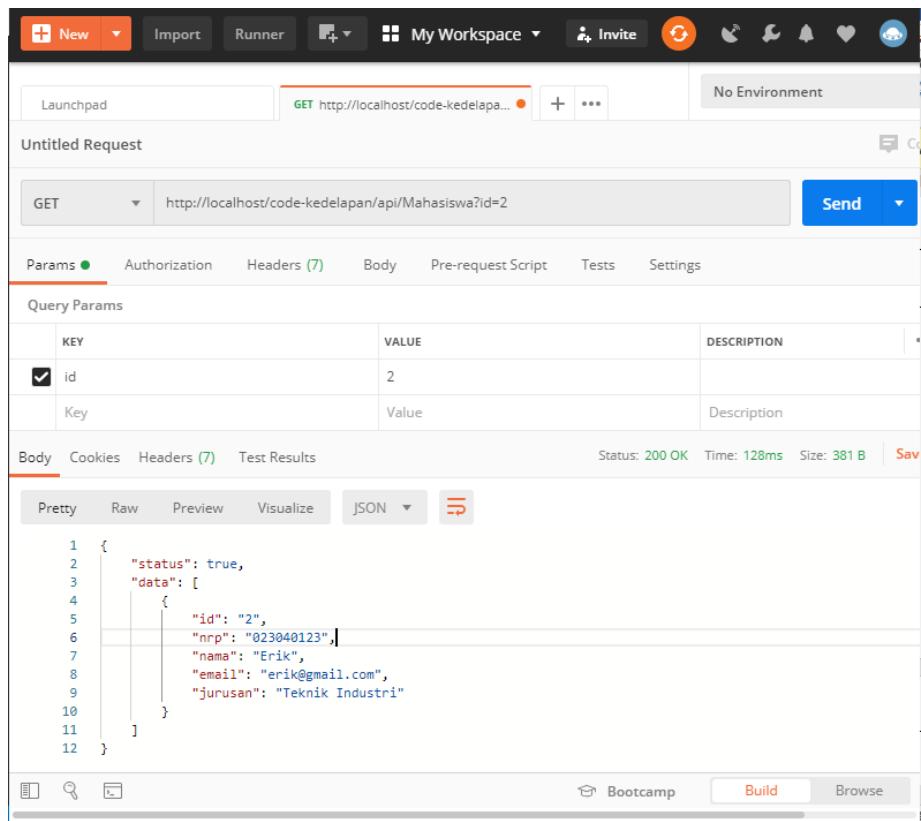
```

3

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

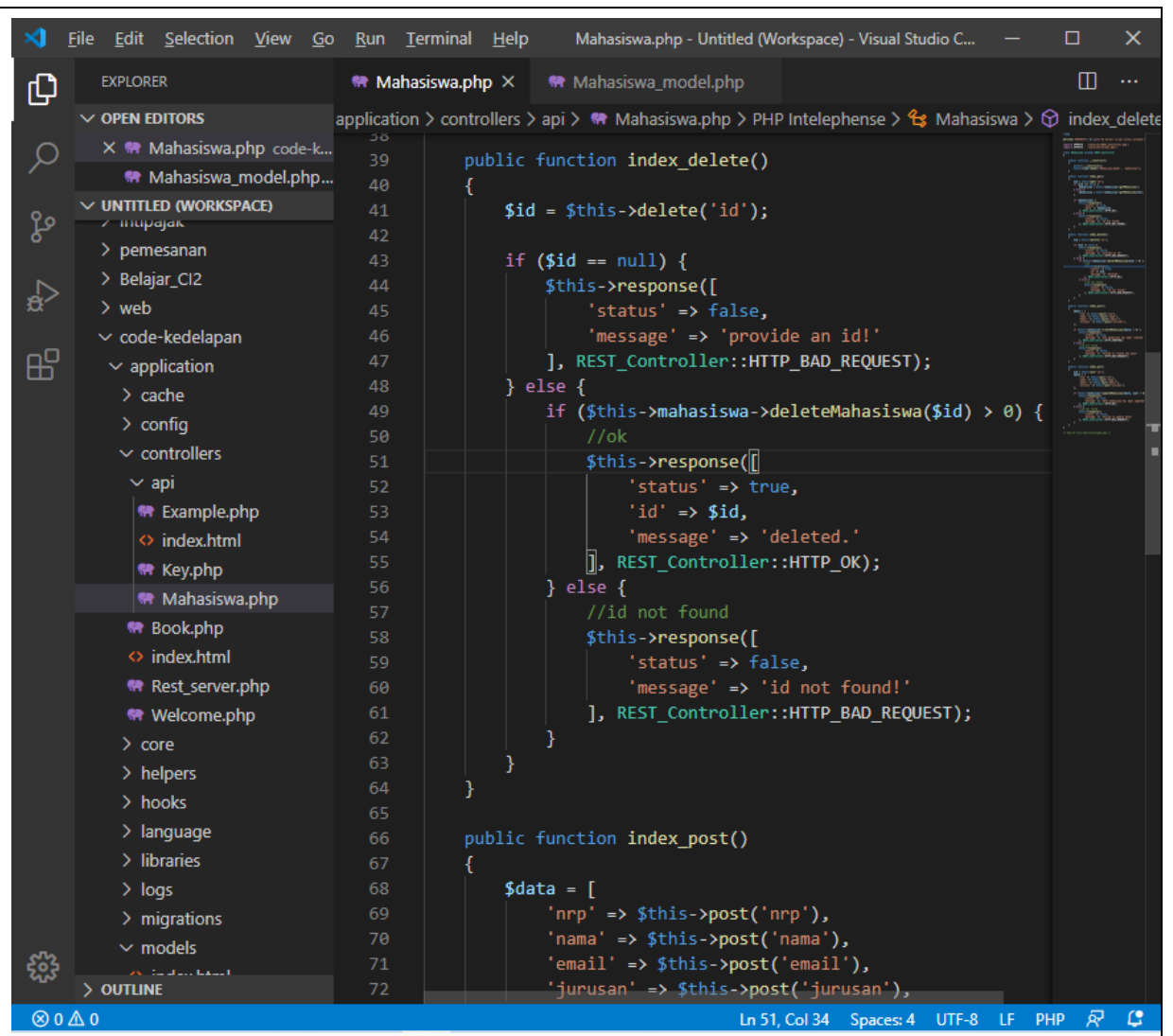


Kita juga dapat melakukan pencarian berdasarkan id, hasilnya akan seperti ini:



4

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



```
File Edit Selection View Go Run Terminal Help Mahasiswa.php - Untitled (Workspace) - Visual Studio C...
EXPLORER
OPEN EDITORS
Mahasiswa.php code-k...
Mahasiswa_model.php...
UNTITLED (WORKSPACE)
inupajak
pemesanan
Belajar_CI2
web
code-kedelapan
application
cache
config
controllers
api
Example.php
index.html
Key.php
Mahasiswa.php
Book.php
index.html
Rest_server.php
Welcome.php
core
helpers
hooks
language
libraries
logs
migrations
models
OUTLINE
Mahasiswa.php
application > controllers > api > Mahasiswa.php > PHP Intelephense > Mahasiswa > index_post
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
if ($this->mahasiswa->createMahasiswa($data) > 0) {
    $this->response([
        'status' => true,
        'message' => 'new mahasiswa has been created'
    ], REST_Controller::HTTP_CREATED);
} else {
    //id not found
    $this->response([
        'status' => false,
        'message' => 'failed to create new data!'
    ], REST_Controller::HTTP_BAD_REQUEST);
}

public function index_put()
{
    $id = $this->put('id');
    $data = [
        'nrp' => $this->put('nrp'),
        'nama' => $this->put('nama'),
        'email' => $this->put('email'),
        'jurusan' => $this->put('jurusan'),
    ];

    if ($this->mahasiswa->updateMahasiswa($data, $id) > 0)
    $this->response([
        'status' => true,
        'message' => 'data mahasiswa has been updated'
    ], REST_Controller::HTTP_OK);
} else {
    //id not found
    $this->response([
        'status' => false,
        'message' => 'failed to update data!'
    ], REST_Controller::HTTP_BAD_REQUEST);
}
}
}

/* End of file Controllername.php */
```

Ln 68, Col 18 Spaces: 4 UTF-8 LF PHP

5

Dan lanjutkan kode pada file **Mahasiswa_model.php**:

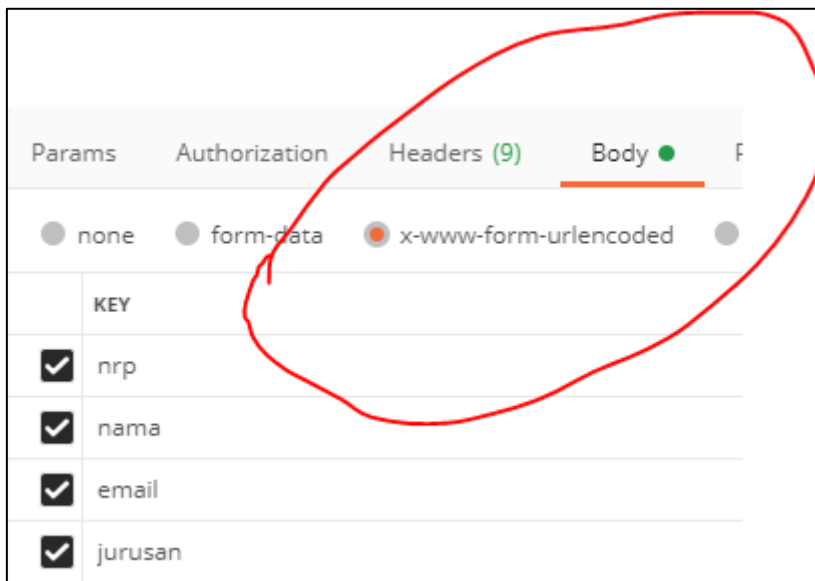
```

17 public function deleteMahasiswa($id)
18 {
19     $this->db->delete('mahasiswa', ['id' => $id]);
20     return $this->db->affected_rows();
21 }
22
23 public function createMahasiswa($data)
24 {
25     $this->db->insert('mahasiswa', $data);
26     return $this->db->affected_rows();
27 }
28
29 public function updateMahasiswa($data, $id)
30 {
31     $this->db->update('mahasiswa', $data, ['id' => $id]);
32     return $this->db->affected_rows();
33 }
34 }
35
36 /* End of file ModelName.php */
37

```

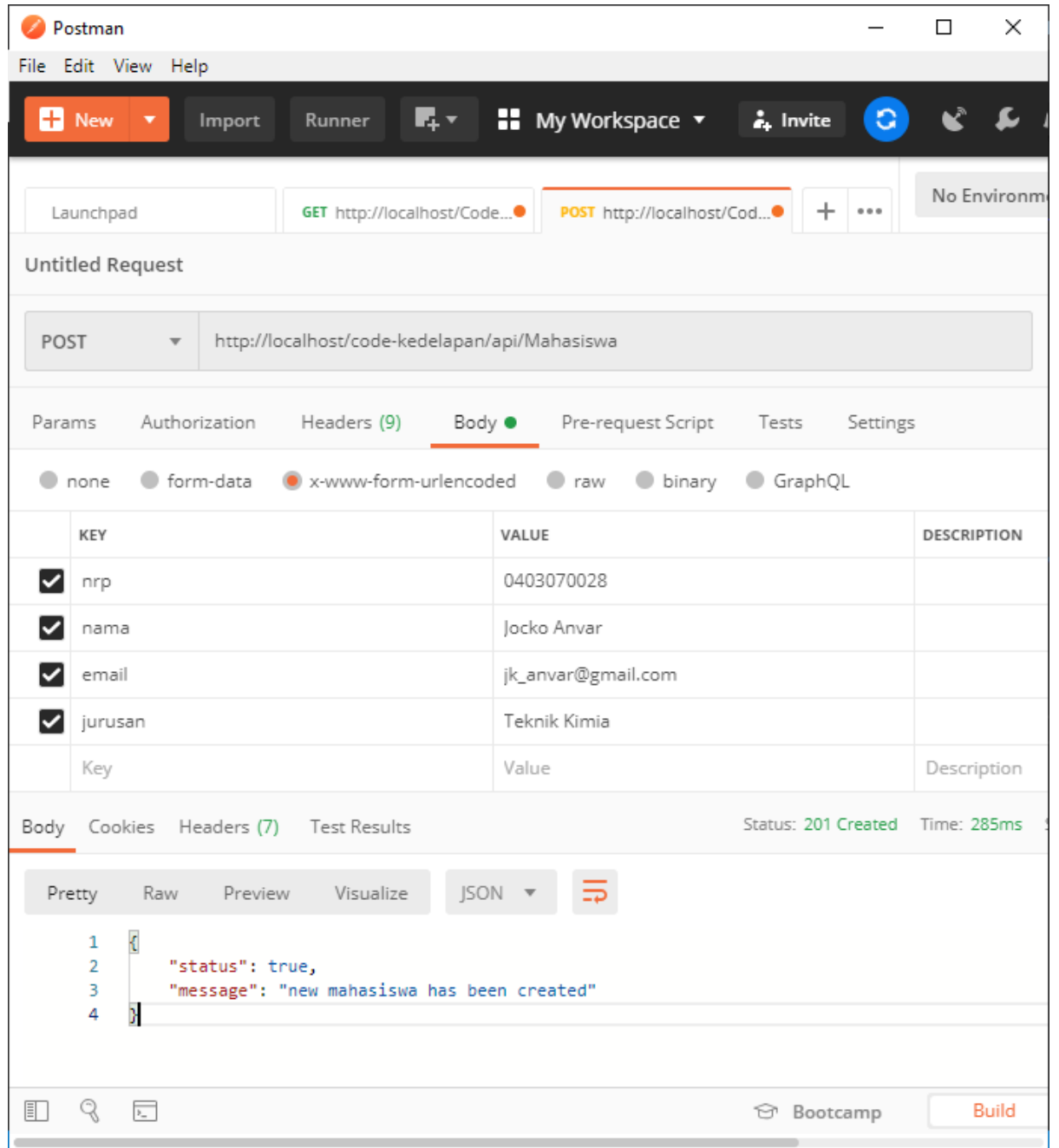
6

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:



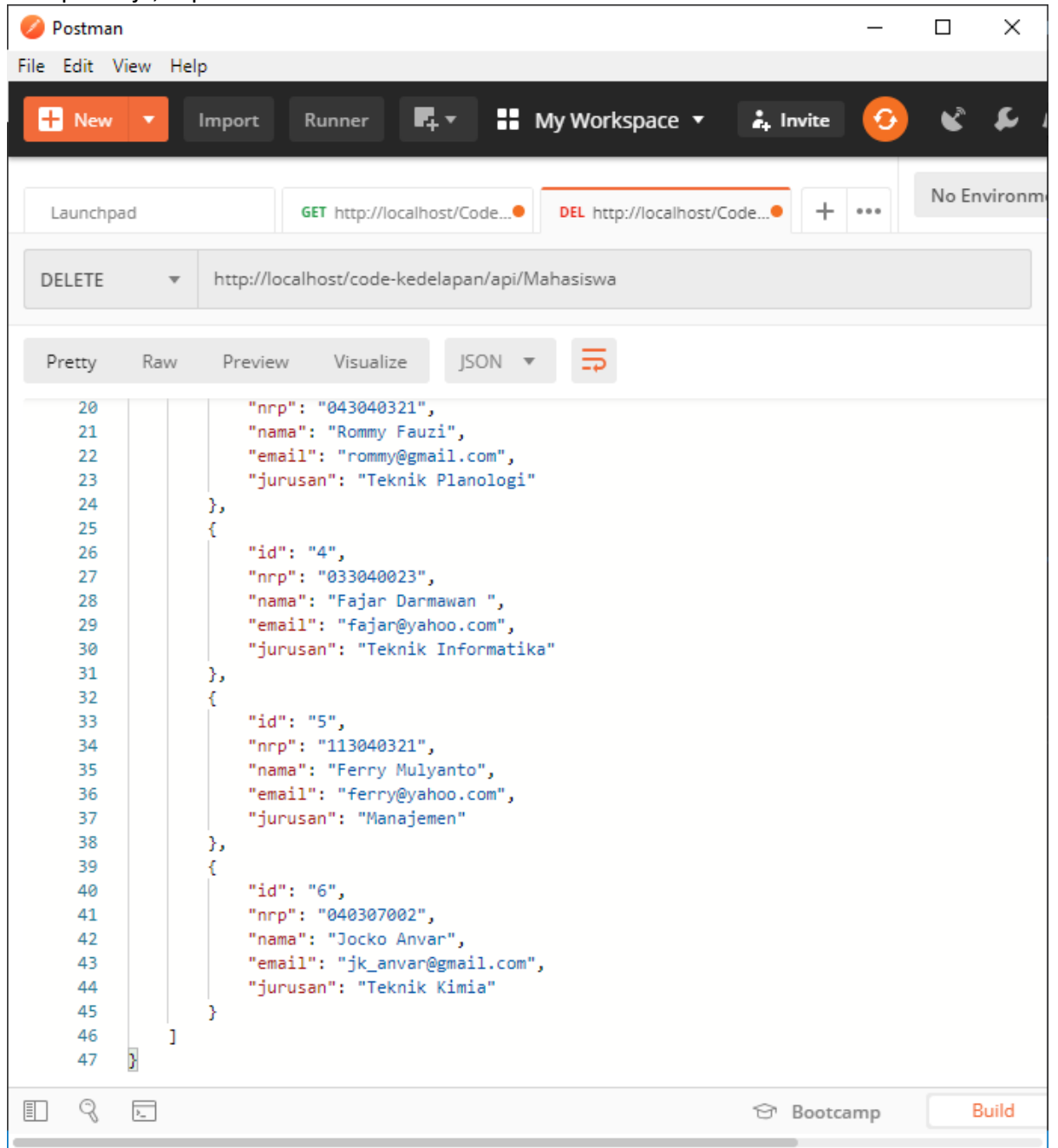
7

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

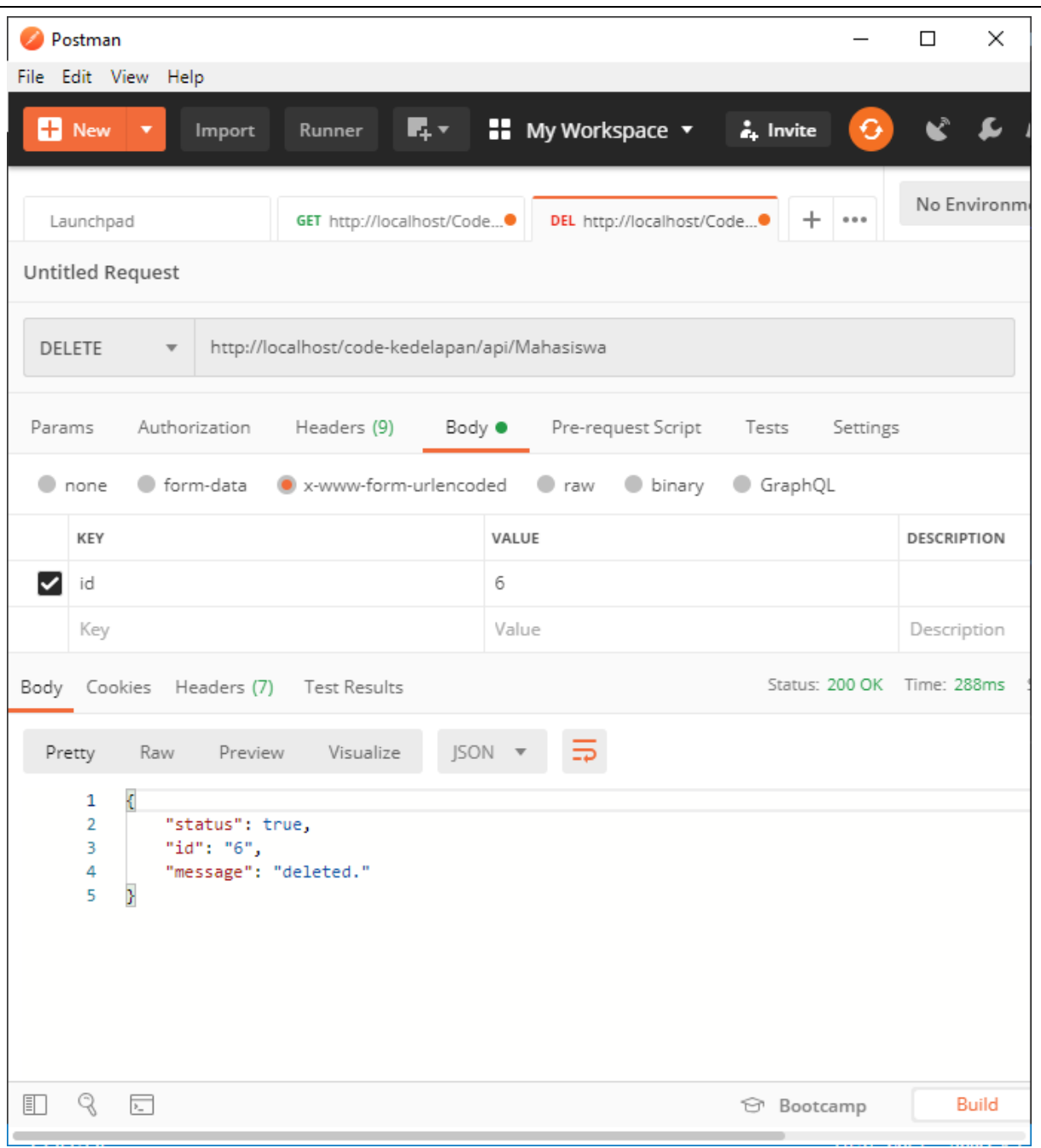


8

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



Setelah di delete



-- Selamat Mengerjakan --