

Nama : Chatlea Cinta Putri Widyanto

Kelas : XI Laravel

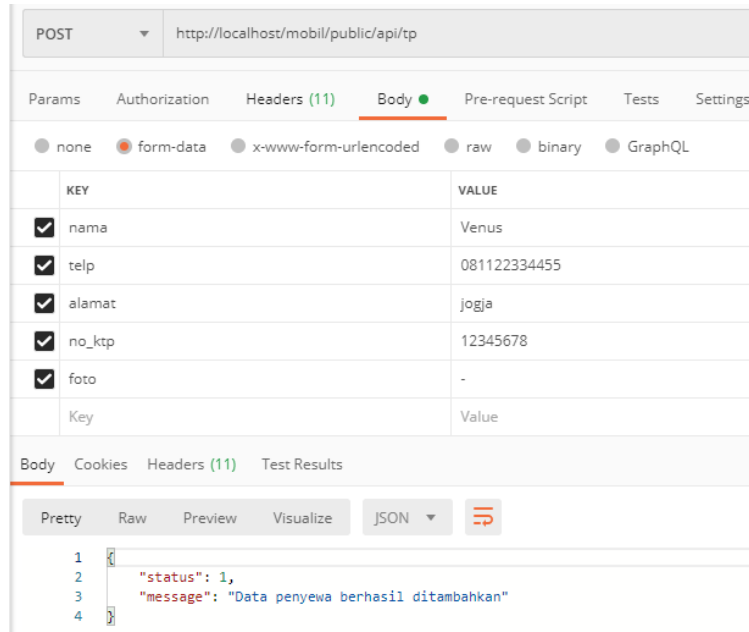
No. Absen : 09

TUGAS CRUD MASTER

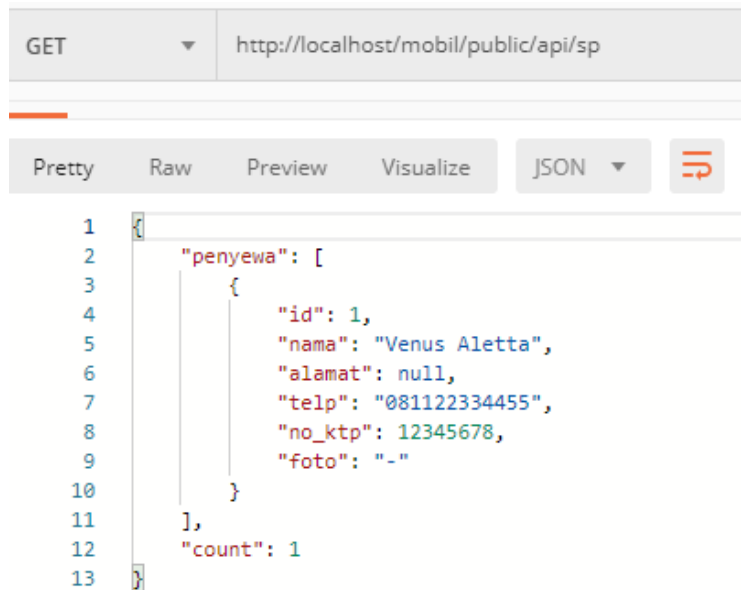
HASIL POSTMAN

1. Penyewa

a) Create



b) Read



c) Update

PUT ▼ http://localhost/mobil/public/api/uppl/1

Params Authorization Headers (11) **Body** ● Pre-request Script Tests Se

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

KEY	VALUE
<input checked="" type="checkbox"/> nama	Venus Aletta
<input checked="" type="checkbox"/> telp	081122334455
<input checked="" type="checkbox"/> alamat	Jakarta Selatan
<input checked="" type="checkbox"/> no_ktp	12345678
<input checked="" type="checkbox"/> foto	-
Key	Value

Body Cookies Headers (11) Test Results

Pretty Raw Preview Visualize JSON ▼

```
1 {
2   "status": 1,
3   "message": "Data penyewa berhasil diubah"
4 }
```

d) Delete

DELETE ▼ http://localhost/mobil/public/api/dp/2

Body Cookies Headers (11) Test Results

Pretty Raw Preview Visualize JSON ▼

```
1 {
2   "status": 1,
3   "message": "Data penyewa berhasil dihapus"
4 }
```

2. Mobil

a) Create

POST ▼ http://localhost/mobil/public/api/tm

Params Authorization Headers (11) **Body** ● Pre-request Script

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

KEY	VALUE
<input checked="" type="checkbox"/> merk	Innova
<input checked="" type="checkbox"/> plat_no	N1476HB
<input checked="" type="checkbox"/> foto	-
<input checked="" type="checkbox"/> ket	Warna silver
Key	Value

Body Cookies Headers (11) Test Results

Pretty Raw Preview Visualize JSON ▼

```
1 {
2   "status": 1,
3   "message": "Data mobil berhasil ditambahkan"
4 }
```

b) Read

GET http://localhost/mobil/public/api/sm

Body Cookies Headers (11) Test Results

Pretty Raw Preview Visualize JSON

```
1 {
2   "mobil": [
3     {
4       "id": 1,
5       "merk": "Innova",
6       "plat_no": "N1476HB",
7       "foto": "-",
8       "ket": "Warna silver, kondisi baik, bensin full"
9     }
10  ],
11  "count": 1
12 }
```

c) Update

PUT http://localhost/mobil/public/api/upm/1

Params Authorization Headers (11) Body Pre-request Script Tests Settings

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

	KEY	VALUE
<input checked="" type="checkbox"/>	merk	Innova
<input checked="" type="checkbox"/>	plat_no	N1476HB
<input checked="" type="checkbox"/>	ket	Warna silver, kondisi baik, bensin full
<input checked="" type="checkbox"/>	foto	-
	Key	Value

Body Cookies Headers (11) Test Results

Pretty Raw Preview Visualize JSON

```
1 {
2   "status": 1,
3   "message": "Data mobil berhasil diubah"
4 }
```

d) Delete

DELETE http://localhost/mobil/public/api/dm/1

Body Cookies Headers (11) Test Results

Pretty Raw Preview Visualize JSON

```
1 {
2   "status": 1,
3   "message": "Data mobil berhasil dihapus"
4 }
```

3. Jenis Mobil

a) Create

POST http://localhost/mobil/public/api/tjm

Params Authorization Headers (11) **Body** Pre-request Script Tests

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

	KEY	VALUE
<input checked="" type="checkbox"/>	nama_jenis	sedan
	Key	Value

Body **Cookies** Headers (11) Test Results

Pretty Raw Preview Visualize JSON

```
1 {
2   "status": 1,
3   "message": "Data jenis mobil berhasil ditambahkan"
4 }
```

b) Read

GET http://localhost/mobil/public/api/sjm

Body **Cookies** Headers (11) Test Results

Pretty Raw Preview Visualize JSON

```
1 {
2   "jenis": [
3     {
4       "id": 1,
5       "nama_jenis": "Sedan"
6     }
7   ],
8   "count": 1
9 }
```

c) Update

PUT http://localhost/mobil/public/api/ujm/1

Params Authorization Headers (11) **Body** Pre-request Script

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

	KEY	VALUE
<input checked="" type="checkbox"/>	nama_jenis	Sedan
	Key	Value

Body **Cookies** Headers (11) Test Results

Pretty Raw Preview Visualize JSON

```
1 {  
2   "status": 1,  
3   "message": "Data jenis mobil berhasil diubah"  
4 }
```

d) Delete

DELETE http://localhost/mobil/public/api/djm/1

Body **Cookies** Headers (11) Test Results

Pretty Raw Preview Visualize JSON

```
1 {  
2   "status": 1,  
3   "message": "Data jenis mobil berhasil dihapus"  
4 }
```

API.PHP

```
routes > api.php
15
16 Route::middleware('auth:api')->get('/user', function (Request $request) {
17     return $request->user();
18 });
19 Route::post('register', 'PetugasController@register');
20 Route::post('login', 'PetugasController@login');
21 Route::get('/', function(){
22     return Auth::user()->level;
23 }->middleware('jwt.verify');
24
25 Route::get('user', 'PetugasController@getAuthenticatedUser')->middleware('jwt.verify');
26
27 //penyewa
28 Route::get('ip/{id}', 'PenyewaController@index')->middleware('jwt.verify');
29 Route::post('tp', 'PenyewaController@store')->middleware('jwt.verify');
30 Route::put('/upp/{id}', 'PenyewaController@update')->middleware('jwt.verify');
31 Route::delete('dp/{id}', 'PenyewaController@destroy')->middleware('jwt.verify');
32 Route::get('sp', 'PenyewaController@tampil')->middleware('jwt.verify');
33
34 //mobil
35 Route::get('im/{id}', 'MobilController@index')->middleware('jwt.verify');
36 Route::post('tm', 'MobilController@store')->middleware('jwt.verify');
37 Route::put('/upm/{id}', 'MobilController@update')->middleware('jwt.verify');
38 Route::delete('dm/{id}', 'MobilController@destroy')->middleware('jwt.verify');
39 Route::get('sm', 'MobilController@tampil')->middleware('jwt.verify');
40
41 //jenis mobil
42 Route::get('ijm/{id}', 'JenisMobilController@index')->middleware('jwt.verify');
43 Route::post('tjm', 'JenisMobilController@store')->middleware('jwt.verify');
44 Route::put('/ujm/{id}', 'JenisMobilController@update')->middleware('jwt.verify');
45 Route::delete('djm/{id}', 'JenisMobilController@destroy')->middleware('jwt.verify');
46 Route::get('sjm', 'JenisMobilController@tampil')->middleware('jwt.verify');
47
```

MODEL

1. Penyewa

```
app > Penyewa.php
1 <?php
2
3 namespace App;
4
5 use Illuminate\Database\Eloquent\Model;
6
7 class Penyewa extends Model
8 {
9     protected $table = "penyewa";
10    protected $primaryKey = "id";
11    protected $fillable = ['nama', 'alamat', 'telp', 'no_ktp', 'foto'];
12    public $timestamps = false;
13
14    public function Pelanggan(){
15        return $this->HasMany('App/Pelanggan', 'id');
16    }
17
18 }
19
```

2. Mobil

```
app > Mobil.php
1  <?php
2
3  namespace App;
4
5  use Illuminate\Database\Eloquent\Model;
6
7  class Mobil extends Model
8  {
9      protected $table = "mobil";
10     protected $primaryKey = "id";
11     protected $fillable = ['merk', 'plat_no', 'foto', 'ket'];
12     public $timestamps = false;
13
14     public function Mobil(){
15         return $this->HasMany('App/Mobil','id');
16     }
17
18 }
19
```

3. Jenis mobil

```
app > JenisMobil.php
1  <?php
2
3  namespace App;
4
5  use Illuminate\Database\Eloquent\Model;
6
7  class JenisMobil extends Model
8  {
9      protected $table = "jenis_mobil";
10     protected $primaryKey = "id";
11     protected $fillable = ['nama_jenis'];
12     public $timestamps = false;
13
14 }
15
```

CONTROLLER

1. PenyewaController

```
app > Http > Controllers > PenyewaController.php
1  <?php
2
3  namespace App\Http\Controllers;
4  use Illuminate\Http\Request;
5  use Illuminate\Support\Facades\Validator;
6  use DB;
7  use App\Penyewa;
8  use Auth;
9
10 class PenyewaController extends Controller
11 {
12     public function index($id){
13         $penyewa=DB::table('penyewa')
14             ->where('pnypewa.id',$id)
15             ->get();
16         return response()->json($penyewa);
17     }
18     public function store(Request $req){
19         if(Auth::user()->level=="admin"){
20             $validator = Validator::make($req->all(),
21             [
22                 'nama' => 'required',
23                 'alamat' => 'required',
24                 'telp' => 'required',
25                 'no_ktp' => 'required',
26                 'foto' => 'required'
27             ]);
28             if($validator->fails()){
29                 return Response()->json($validator->errors()->toJson(),400);
30             }
31
32             $simpan = Penyewa::create([
33                 'nama' => $req->nama,
34                 'alamat' => $req->alamat,
35                 'telp' => $req->telp,
36                 'no_ktp' => $req->no_ktp,
37                 'foto' => $req->foto
38             ]);
39             $status = 1;
40             $message = "Data penyewa berhasil ditambahkan";
```



```

40     $message = "Data penyewa berhasil ditambahkan";
41     if($simpan){
42         return Response()->json(compact('status', 'message'));
43     }else {
44         return Response()->json(['status'=> 0]);
45     }
46 }
47 else {
48     return response()->json(['status'=>'anda bukan admin']);
49 }
50 }
51 public function update($id, Request $req)
52 {
53     if(Auth::user()->level=="admin"){
54         $validator=Validator::make($req->all(),
55         [
56             'nama' => 'required',
57             'alamat' => 'required',
58             'telp' => 'required',
59             'no_ktp' => 'required',
60             'foto' => 'required'
61         ]);
62         if($validator->fails()){
63             return Response()->json($validator->errors()->toJson(),400);
64         }
65         $ubah=Penyewa::where('id',$id)->update([
66             'nama' => $req->nama,
67             'alamat' => $req->alamat,
68             'telp' => $req->telp,
69             'no_ktp' => $req->no_ktp,
70             'foto' => $req->foto
71         ]);
72         $status = 1;
73         $message = "Data penyewa berhasil diubah";
74         if($ubah){
75             return Response()->json(compact('status', 'message'));
76         }else {
77             return Response()->json(['status'=> 0]);
78         }
79     }
80     else {

```

```

77         return Response()->json(['status'=> 0]);
78     }
79 }
80 else {
81     return response()->json(['status'=>'anda bukan admin']);
82 }
83 }
84 public function tampil(){
85     $data = Penyewa::get();
86     $count = $data->count();
87     $penyewa = array();
88     foreach ($data as $d){
89
90         $penyewa[] = array([
91             'id' => $d->id,
92             'nama' => $d->nama,
93             'alamat' => $d->alamat,
94             'telp' => $d->telp,
95             'no_ktp' => $d->no_ktp,
96             'foto' => $d->foto
97         ]);
98     }
99     return Response()->json(compact('penyewa','count'));
100 }
101 public function destroy($id)
102 {
103     if(Auth::user()->level=="admin"){
104         $hapus=Penyewa::where('id',$id)->delete();
105         $status = 1;
106         $message = "Data penyewa berhasil dihapus";
107         if($hapus){
108             return Response()->json(compact('status', 'message'));
109         }else {
110             return Response()->json(['status'=> 0]);
111         }
112     }
113     else {
114         return response()->json(['status'=>'anda bukan admin']);
115     }
116 }

```

2. MobilController

```
app > Http > Controllers > MobilController.php
1  <?php
2
3  namespace App\Http\Controllers;
4
5  use Illuminate\Http\Request;
6  use Illuminate\Support\Facades\Validator;
7  use DB;
8  use App\Mobil;
9  use Auth;
10
11 class MobilController extends Controller
12 {
13     public function index($id){
14         $mobil=DB::table('mobil')
15             ->where('mobil.id',$id)
16             ->get();
17         return response()->json($mobil);
18     }
19     public function store(Request $req){
20         if(Auth::user()->level=="admin"){
21             $validator = Validator::make($req->all(),
22             [
23                 'merk' => 'required',
24                 'plat_no' => 'required',
25                 'foto' => 'required',
26                 'ket' => 'required'
27             ]);
28             if($validator->fails()){
29                 return Response()->json($validator->errors()->toJson(),400);
30             }
31
32             $simpan = Mobil::create([
33                 'merk' => $req->merk,
34                 'plat_no' => $req->plat_no,
35                 'foto' => $req->foto,
36                 'ket' => $req->ket
37             ]);
38             $status = 1;
39             $message = "Data mobil berhasil ditambahkan";
40             if($simpan){
41                 return Response()->json(compact('status','message'));
```

```

40         if($simpan){
41             return Response()->json(compact('status', 'message'));
42         }else {
43             return Response()->json(['status'=> 0]);
44         }
45     }
46     else {
47         return response()->json(['status'=>'anda bukan admin']);
48     }
49 }
50 public function update($id, Request $req)
51 {
52     if(Auth::user()->level=="admin"){
53         $validator=Validator::make($req->all(),
54             [
55                 'merk' => 'required',
56                 'plat_no' => 'required',
57                 'foto' => 'required',
58                 'ket' => 'required'
59             ]);
60         if($validator->fails()){
61             return Response()->json($validator->errors()->toJson(),400);
62         }
63         $ubah=Mobil::where('id',$id)->update([
64             'merk' => $req->merk,
65             'plat_no' => $req->plat_no,
66             'foto' => $req->foto,
67             'ket' => $req->ket
68         ]);
69         $status = 1;
70         $message = "Data mobil berhasil diubah";
71         if($ubah){
72             return Response()->json(compact('status', 'message'));
73         }else {
74             return Response()->json(['status'=> 0]);
75         }
76     }
77     else {
78         return response()->json(['status'=>'anda bukan admin']);
79     }

```

```

app > Http > Controllers > MobilController.php
72         return Response()->json(compact('status', 'message'));
73     }else {
74         return Response()->json(['status'=> 0]);
75     }
76 }
77 else {
78     return response()->json(['status'=>'anda bukan admin']);
79 }
80 }
81 public function tampil(){
82     $data = Mobil::get();
83     $count = $data->count();
84     $mobil = array();
85     foreach ($data as $d){
86         $mobil[] = array(
87             'id' => $d->id,
88             'merk' => $d->merk,
89             'plat_no' => $d->plat_no,
90             'foto' => $d->foto,
91             'ket' => $d->ket
92         );
93     }
94     return Response()->json(compact('mobil', 'count'));
95 }
96 public function destroy($id)
97 {
98     if(Auth::user()->level=="admin"){
99         $hapus=Mobil::where('id',$id)->delete();
100         $status = 1;
101         $message = "Data mobil berhasil dihapus";
102         if($hapus){
103             return Response()->json(compact('status', 'message'));
104         }else {
105             return Response()->json(['status'=> 0]);
106         }
107     }
108 }
109 else {
110     return response()->json(['status'=>'anda bukan admin']);
111 }

```

3. JenisMobilController

```
app > Http > Controllers > JenisMobilController.php
1  <?php
2
3  namespace App\Http\Controllers;
4
5  use Illuminate\Http\Request;
6  use Illuminate\Support\Facades\Validator;
7  use DB;
8  use App\JenisMobil;
9  use Auth;
10
11 class JenisMobilController extends Controller
12 {
13     public function index($id){
14         $jenis_mobil=DB::table('jenis_mobil')
15         ->where('jenis_mobil.id',$id)
16         ->get();
17         return response()->json($jenis_cuci);
18     }
19     public function store(Request $req){
20         if(Auth::user()->level=="admin"){
21             $validator = Validator::make($req->all(),
22             [
23                 'nama_jenis' => 'required'
24             ]);
25             if($validator->fails()){
26                 return Response()->json($validator->errors()->toJson(),400);
27             }
28
29             $simpan = JenisMobil::create([
30                 'nama_jenis' => $req->nama_jenis
31             ]);
32             $status = 1;
33             $message = "Data jenis mobil berhasil ditambahkan";
34             if($simpan){
35                 return Response()->json(compact('status', 'message'));
36             }else {
37                 return Response()->json(['status'=> 0]);
38             }
39         }
40         else {
41             return response()->json(['status'=> 0, 'message'=> 'Anda bukan admin!']);
42         }
43     }
44 }
```

```

39     }
40     else {
41         return response()->json(['status'=>'anda bukan admin']);
42     }
43 }
44 public function update($id, Request $req)
45 {
46     if(Auth::user()->level=="admin"){
47         $validator=Validator::make($req->all(),
48         [
49             'nama_jenis' => 'required'
50         ]);
51         if($validator->fails()){
52             return Response()->json($validator->errors()->toJson(),400);
53         }
54         $ubah=JenisMobil::where('id',$id)->update([
55             'nama_jenis' => $req->nama_jenis
56         ]);
57         $status = 1;
58         $message = "Data jenis mobil berhasil diubah";
59         if($ubah){
60             return Response()->json(compact('status', 'message'));
61         }else {
62             return Response()->json(['status'=> 0]);
63         }
64     }
65     else {
66         return response()->json(['status'=>'anda bukan admin']);
67     }
68 }
69 public function tampil(){
70     $data = JenisMobil::get();
71     $count = $data->count();
72     $jenis = array();
73     foreach ($data as $d){
74
75         $jenis[] = array(
76             'id' => $d->id,
77             'nama_jenis' => $d->nama_jenis
78         );
79     }

```

```

app > Http > Controllers > JenisMobilController.php
60         return Response()->json(compact('status', 'message'));
61     }else {
62         return Response()->json(['status'=> 0]);
63     }
64 }
65 else {
66     return response()->json(['status'=>'anda bukan admin']);
67 }
68 }
69 public function tampil(){
70     $data = JenisMobil::get();
71     $count = $data->count();
72     $jenis = array();
73     foreach ($data as $d){
74
75         $jenis[] = array(
76             'id' => $d->id,
77             'nama_jenis' => $d->nama_jenis
78         );
79     }
80     return Response()->json(compact('jenis','count'));
81 }
82 public function destroy($id)
83 {
84     if(Auth::user()->level=="admin"){
85         $hapus=JenisMobil::where('id',$id)->delete();
86         $status = 1;
87         $message = "Data jenis mobil berhasil dihapus";
88         if($hapus){
89             return Response()->json(compact('status', 'message'));
90         }else {
91             return Response()->json(['status'=> 0]);
92         }
93     }
94     else {
95         return response()->json(['status'=>'anda bukan admin']);
96     }
97 }
98
99 }
100

```