

PART 14 UPLOAD FILE-FILTER



Shera Zahra Alya Nasywa Hardian

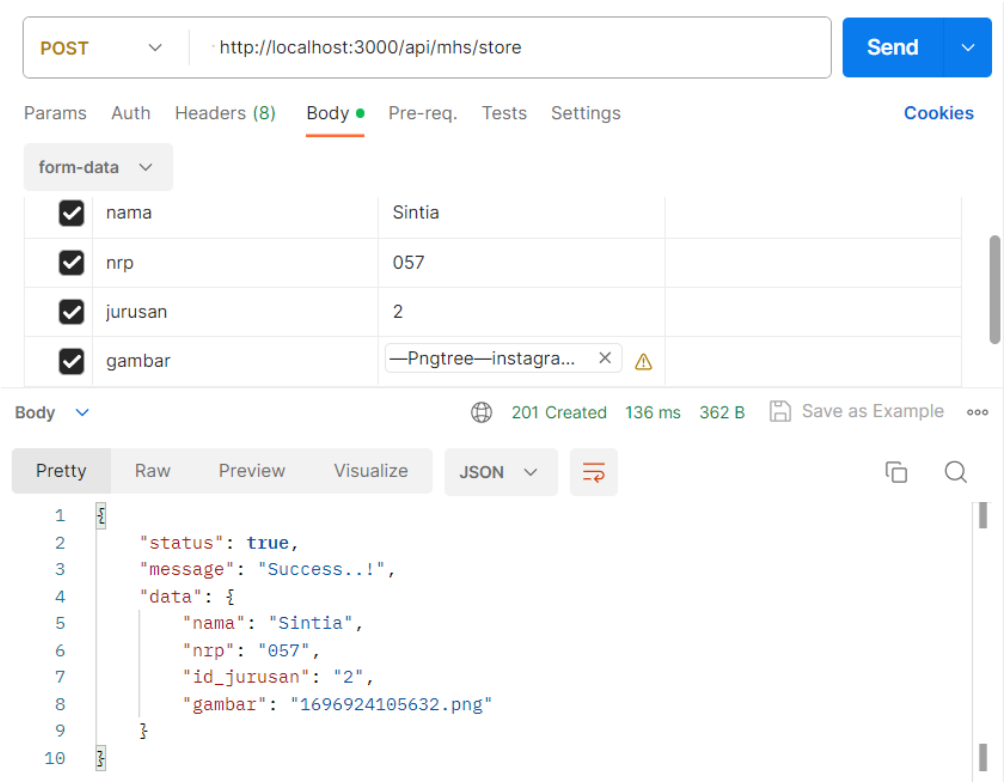
(3122522026)

3 D3 PSDKU Sumenep

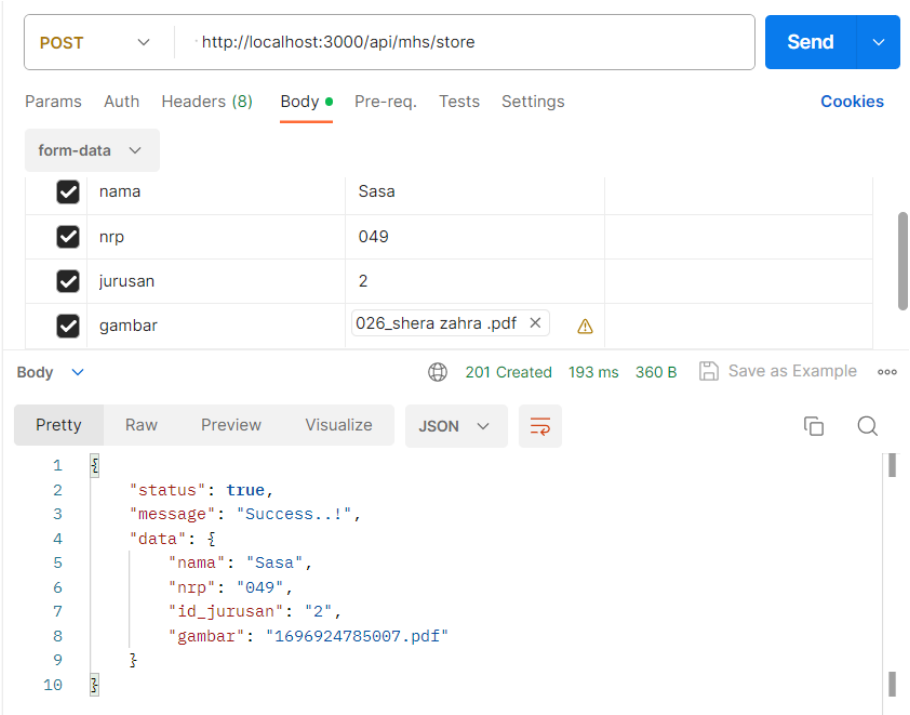
**PRODI D3 TEKNIK INFORMATIKA
DEPARTEMEN TEKNIK INFORMATIKA DAN KOMPUTER
PENS PSDKU SUMENEP**

```
21   const fileFilter = (req, file, cb) => {
22     if (file.mimetype === 'image/jpeg' || file.mimetype === 'image/png') {
23       cb(null, true);
24     } else {
25       cb(new Error ('Jenis file tidak diizinkan'), false);
26     }
27   };
```

```
28   const upload = multer({storage: storage, fileFilter: fileFilter})
29
```



```
20
21   const fileFilter = (req, file, cb) => {
22     if (file.mimetype === 'image/jpeg' || file.mimetype === 'image/png' || file.mimetype === 'application/pdf') {
23       cb(null, true);
24     } else {
25       cb(new Error ('Jenis file tidak diizinkan'), false);
26     }
27   };
```



Link github:
<https://github.com/sherazahra/express.git>

TUGAS BARU

```
config > JS db.js > ...
1  let mysql = require('mysql');
2  let connection = mysql.createConnection({
3    host:      'localhost',
4    user:      'root',
5    password:  '',
6    database:  'tugas'
7  });
8
9  connection.connect(function (error) {
10   if(!error){
11     console.log(error)
12   }else{
13     console.log('koneksi berhasil');
14   }
15 })
16
17 module.exports = connection;
```

```
JS index.js > ...
1  const express = require('express')
2  const app = express()
3  const port = 3000
4
5  const bodyPs = require('body-parser');
6  app.use(bodyPs.urlencoded({ extended: false}));
7  app.use(bodyPs.json());
8
9  app.get('/', (req, res) => {
10   res.send("Halo lovedek")
11 });
12
13 const transmisiRouter = require('./routes/transmisi');
14 app.use('/api/transmisi', transmisiRouter);
15
16 const kendaraanRouter = require('./routes/kendaraan');
17 app.use('/api/kendaraan', kendaraanRouter);
18
19 app.listen(port, () => {
20   console.log(`aplikasi berjalan di http://localhost:${port}`)
21 })
22
```

CRUD TRANSMISI

```
1  const express = require('express');
2  const router = express.Router();
3  const {body, validationResult } = require('express-validator');
4
5  const connection = require('../config/db.js');
6
7  router.get('/', function (req, res){
8    connection.query('select * from transmisi order by id_transmisi desc', function(err, rows){
9      if(err){
10        return res.status(500).json({
11          status: false,
12          message: 'Server Failed',
13        })
14      }else{
15        return res.status(200).json({
16          status: true,
17          message: 'Data transmisi',
18          data: rows
19        })
20      }
21    })
22  });
23
```

```

24 router.post('/store', [
25   body('nama_transmisi').notEmpty()
26 ], (req, res) => {
27   const error = validationResult(req);
28   if(!error.isEmpty()){
29     return res.status(422).json({
30       error: error.array()
31     })
32   }
33   let Data = {
34     nama_transmisi: req.body.nama_transmisi
35   }
36   connection.query('insert into transmisi set ?', Data, function(err, rows){
37     if(err){
38       return res.status(500).json({
39         status: false,
40         message: 'Server Error',
41       })
42     }else{
43       return res.status(201).json({
44         satus: true,
45         message: 'Success..!',
46         data: rows[0]
47       })
48     }
49   })
50 })
51

```

```

52 router.get('/:id', function (req, res) {
53   let id = req.params.id;
54   connection.query(`select * from transmisi where id_transmisi = ${id}`, function (err, rows) {
55     if(err){
56       return res.status(500).json({
57         status: false,
58         message: 'Server Error',
59       })
60     }
61     if(rows.length <=0){
62       return res.status(404).json({
63         status: false,
64         message: 'Not Found',
65       })
66     }
67     else{
68       return res.status(200).json({
69         status: true,
70         message: 'Data transmisi',
71         data: rows[0]
72       })
73     }
74   })
75 })
76

```

```

77 router.patch('/update/:id', [
78   body('nama_transmisi').notEmpty()
79 ], (req,res) => {
80   const error = validationResult(req);
81   if(!error.isEmpty()){
82     return res.status(422).json({
83       error: error.array()
84     });
85   }
86   let id = req.params.id;
87   let Data = {
88     nama_transmisi: req.body.nama_transmisi
89   }
90   connection.query(`update transmisi set ? where id_transmisi = ${id}`, Data, function (err, rows) {
91     if(err){
92       return res.status(500).json({
93         status: false,
94         message: 'Server Error',
95       })
96     }else {
97       return res.status(200).json({
98         status: true,
99         message: 'Update Success..!'
100       })
101     }
102   })
103 })
104

```

```

105 router.delete('/delete/:id', function(req, res){
106   let id = req.params.id;
107   connection.query(`delete from transmisi where id_transmisi = ${id}`, function (err, rows) {
108     if(err){
109       return res.status(500).json({
110         status: false,
111         message: 'Server Error',
112       })
113     }else{
114       return res.status(500).json({
115         status: true,
116         message: 'Data berhasil di hapus!',
117       })
118     }
119   })
120 })
121
122 module.exports = router;

```

GET

http://localhost:3000/api/transmisi

Send

Params

Auth

Headers (8)

Body

Pre-req.

Tests

Settings

Cookies

Body

200 OK 107 ms 287 B

Save as Example

Pretty

Raw

Preview

Visualize

JSON

```

1 {
2   "status": true,
3   "message": "Data transmisi",
4   "data": []
5 }

```

POST

▼

http://localhost:3000/api/transmisi/store

Params

Authorization

Headers (8)

Body ●

Pre-request Script

Tests

Settings

● none

● form-data

● x-www-form-urlencoded

● raw

● binary

● GraphQL

	Key	Value
<input checked="" type="checkbox"/>	nama_transmisi	matic
	Key	Value

Body

Cookies

Headers (7)

Test Results

⌐ Status: 201

Pretty

Raw

Preview

Visualize

JSON ▼

⌐

```
1 {
2   "satus": true,
3   "message": "Success..!"
4 }
```

GET

▼

http://localhost:3000/api/transmisi/1

Params

Authorization

Headers (8)

Body ●

Pre-request Script

Tests

Query Params

	Key	Value
	Key	Value

Body

Cookies

Headers (7)

Test Results

Pretty

Raw

Preview

Visualize

JSON ▼

⌐

```
1 {
2   "status": true,
3   "message": "Data transmisi",
4   "data": {
5     "id_transmisi": 1,
6     "nama_transmisi": "matic"
7   }
8 }
```



http://localhost:3000/api/transmisi/update/1

PATCH



http://localhost:3000/api/transmisi/update/1

Params Authorization Headers (8) **Body** Pre-request Script Tests Settings

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

	Key	Value
<input checked="" type="checkbox"/>	nama_transmisi	manual

Body Cookies Headers (7) Test Results

Pretty

Raw

Preview

Visualize

JSON



```
1 {
2   "status": true,
3   "message": "Update Success..!"
4 }
```

GET



http://localhost:3000/api/transmisi/1

Params Authorization Headers (8) **Body** Pre-request Script Tests Settings

Query Params

	Key	Value
	Key	Value

Body Cookies Headers (7) Test Results

Pretty

Raw

Preview

Visualize

JSON



```
1 {
2   "status": true,
3   "message": "Data transmisi",
4   "data": {
5     "id_transmisi": 1,
6     "nama_transmisi": "manual"
7   }
8 }
```

DELETE

▼

http://localhost:3000/api/transmisi/delete/1

ParamsAuthorizationHeaders (8)Body●Pre-request ScriptTestsSettings

Query Params

	Key	Value
--	-----	-------

BodyCookiesHeaders (7)Test Results

PrettyRawPreviewVisualizeJSON▼

⌵

1

{

2

"status": true,

3

"message": "Data berhasil di hapus!"

4

}

CRUD TRANSPORTASI

```
routes > JS kendaraan.js > ...
1  const express = require('express');
2  const router = express.Router();
3
4  const connection = require('../config/db.js');
5  const { body, validationResult } = require('express-validator');
6
7  const fs = require('fs')
8
9  const multer = require('multer')
10 const path = require ('path');
11 const { error } = require('console');
12
13 const fileFilter = (req,file, cb) =>{
14   if (file.mimetype === 'image/jpeg' || file.mimetype === 'image/png'){
15     cb(null, true);
16   }else{
17     cb(new Error('jenis file tidak ditemukan'),false);
18   }
19 }
20
```



```

21  const storage = multer.diskStorage({
22    destination: (req, file, cb) =>{
23      cb(null, 'public/images')
24    },
25    filename: (req, file, cb) => {
26      console.log(file)
27      cb(null, Date.now() + path.extname(file.originalname) )
28    }
29  })
30  const upload = multer ({storage: storage, fileFilter: fileFilter})
31
32  router.get('/', function (req,res ){
33    connection.query('select * from kendaraan order by no_pol desc', function(err, rows){
34      if(err){
35        return res.status(500).json({
36          status: false,
37          message: 'Server failed',
38        })
39      }else{
40        return res.status(200).json({
41          status: true,
42          message: 'Data Kendaraan',
43          data: rows
44        })
45      }
46    })
47  });
48

```

```

49  router.post('/store', upload.single("gambar_kendaraan") , [
50    body('no_pol').notEmpty(),
51    body('nama_kendaraan').notEmpty(),
52    body('id_transmisi').notEmpty()
53  ],(req, res) => {
54    const error = validationResult(req);
55    if(!error.isEmpty()){
56      return res.status(442).json({
57        error: error.array()
58      });
59    }
60  }
61  let Data = {
62    no_pol: req.body.no_pol,
63    nama_kendaraan: req.body.nama_kendaraan,
64    id_transmisi: req.body.id_transmisi,
65    gambar_kendaraan: req.file.filename
66  }
67  connection.query('insert into kendaraan set ?', Data, function(err, rows){
68    if(err){
69      return res.status(500).json({
70        status: false,
71        message: 'Server error',
72      })
73    }else{
74      return res.status(201).json({
75        status: true,
76        message: 'Sukses',
77        data: rows[0]
78      })
79    }
80  })
81  });
82

```

```

83 router.patch('/update/:id',upload.single("gambar_kendaraan"), [
84     body('no_pol').notEmpty(),
85     body('nama_kendaraan').notEmpty(),
86     body('id_transmisi').notEmpty()
87 ], (req,res) => {
88     const error = validationResult(req);
89     if(!error.isEmpty()){
90         return res.status(422).json({
91             error: error.array()
92         });
93     }
94     let id = req.params.id;
95     let gambar_kendaraan = req.file ? req.file.filename : null;
96
97     connection.query(`select * from kendaraan where no_pol = ${id}`, function (err, rows) {
98         if(err){
99             return res.status(500).json({
100                 status: false,
101                 message: 'Server Error',
102             })
103         }
104         if(rows.length <=0){
105             return res.status(404).json({
106                 status: false,
107                 message: 'Not Found',
108             })
109         }
110         const namaFileLama = rows[0].gambar_kendaraan;
111
112         if (namaFileLama && gambar_kendaraan) {
113             const pathFileLama = path.join(__dirname, '../public/images', namaFileLama);
114             fs.unlinkSync(pathFileLama);
115         }
116

```

```

117         let Data = {
118             no_pol: req.body.no_pol,
119             nama_kendaraan: req.body.nama_kendaraan,
120             id_transmisi: req.body.id_transmisi,
121             gambar_kendaraan: req.file.filename
122         }
123         connection.query(`update kendaraan set ? where no_pol = ${id}`, Data, function (err, rows) {
124             if(err){
125                 return res.status(500).json({
126                     status: false,
127                     message: 'Server Error',
128                 })
129             }else {
130                 return res.status(200).json({
131                     status: true,
132                     message: 'Update Success..!'
133                 })
134             }
135         })
136     })
137 })
138

```

```
139 router.delete('/delete/:id',function(req, res){
140   let id = req.params.id;
141   connection.query(`select * from kendaraan where no_pol = ${id}`, function (err, rows) {
142     if(err){
143       return res.status(500).json({
144         status: false,
145         message: 'server error',
146       })
147     }
148     if(rows.length <=0){
149       return res.status(404).json({
150         staus: false,
151         message: 'not Found',
152       })
153     }
154     const namaFileLama = rows[0].gambar;
155
156     if(namaFileLama) {
157       const pathFileLama = path.join(__dirname, '../public/images', namaFileLama);
158       fs.unlinkSync(pathFileLama);
159     }
160     connection.query(`delete from kendaraan where no_pol = ${id}`, function (err, rows){
161       if(err){
162         return res.status(500).json({
163           status: false,
164           message: 'Server failed',
165         })
166       }else{
167         return res.status(200).json({
168           status: true,
169           message: 'Data Berhasil Dihapus',
170         })
171       }
172     })
173   })
174 })
175 module.exports = router;
```

GET

⌵

http://localhost:3000/api/kendaraan

ParamsAuthorizationHeaders (8)Body●Pre-request ScriptTestsSettings

Query Params

	Key	Value
--	-----	-------

BodyCookiesHeaders (7)Test Results

PrettyRawPreviewVisualizeJSON⌵⌵

1

2

3

4

5

{
 "status": true,
 "message": "Data Kendaraan",
 "data": []
}

POST

http://localhost:3000/api/kendaraan/store

ParamsAuthorizationHeaders (8)BodyPre-request ScriptTestsSettings

none

form-data

x-www-form-urlencoded

raw

binary

GraphQL

	Key	Value	Descri
<input checked="" type="checkbox"/>	no_pol	P3567	
<input checked="" type="checkbox"/>	nama_kendaraan	BRIO	
<input checked="" type="checkbox"/>	id_transmisi	2	
<input checked="" type="checkbox"/>	gambar_kendaraan	<div>—Pngtree—instagram icon instagram logo_35...<div>×</div><div>⚠</div></div>	

BodyCookiesHeaders (7)Test Results

⌐ Status: 201 Create

Pretty

Raw

Preview

Visualize

JSON

⌵

⌵

```
1 {
2   "status": true,
3   "message": "Sukses"
4 }
```

GET

http://localhost:3000/api/kendaraan

ParamsAuthorizationHeaders (8)BodyPre-request ScriptTestsSet

Query Params

	Key	Value
	Key	Value

BodyCookiesHeaders (7)Test Results

⌐

Pretty

Raw

Preview

Visualize

JSON

⌵

⌵

```
1 {
2   "status": true,
3   "message": "Data Kendaraan",
4   "data": [
5     {
6       "no_pol": "P3567",
7       "nama_kendaraan": "BRIO",
8       "id_transmisi": 2,
9       "gambar_kendaraan": "1696928134861.png"
10    }
11  ]
12 }
```

PATCH

http://localhost:3000/api/kendaraan/update/3567

ParamsAuthorizationHeaders (8)BodyPre-request ScriptTestsSettings

none


form-data

x-www-form-urlencoded

raw

binary

GraphQL

	Key	Value
<input checked="" type="checkbox"/>	no_pol	3567
<input checked="" type="checkbox"/>	nama_kendaraan	PAJERO
<input checked="" type="checkbox"/>	id_transmisi	2
<input checked="" type="checkbox"/>	gambar_kendaraan	 x ⚠

BodyCookiesHeaders (7)Test ResultsStatus

PrettyRawPreviewVisualizeJSON

```
1 {
2   "status": true,
3   "message": "Update Success..."
4 }
```

DELETE

http://localhost:3000/api/kendaraan/delete/3567

ParamsAuthorizationHeaders (8)BodyPre-request ScriptTestsSettings

BodyCookiesHeaders (7)Test Results

PrettyRawPreviewVisualizeJSON

```
1 {
2   "status": true,
3   "message": "Data Berhasil Dihapus"
4 }
```

Link github:
https://github.com/sherazahra/tugas_part14.git