



Routing dan Handler pada Express

Routing dan Handler ExpressJs

- Route : mengacu pada bagaimana titik akhir aplikasi (URI) menanggapi permintaan klien.
- Handler atau fungsi callback.
- Route Handler : fungsi callback aplikasi yang berfungsi memproses data dari permintaan request dari route .
- Express Route dapat menerima satu atau lebih fungsi callback yang berfungsi menangani proses handler di Route express.

```
var cb0 = function (req, res, next) {  
  console.log('CB0')  
  next()  
}  
var cb1 = function (req, res, next) {  
  console.log('CB1')  
  next()  
}  
var cb2 = function (req, res) {  
  res.send('Hello from C!')  
}
```

```
app.get('/example/c', [cb0, cb1, cb2])
```

HTTP Methods di ExpressJs

- Http Method : sebuah protokol perangkat lunak WEB yang berfungsi untuk berkomunikasi antar perangkat lunak. Jenisnya : GET, PUT, POST...
- Di expressJs:
 - Syntax: `app.METHOD(path, callback [, callback ...])`
 - Support Method : checkout, copy, delete, get, head, lock , merge , mkactivity , mkcol , move , m-search , notify , options , patch , post , purge , put , report , search , subscribe , trace , unlock , unsubscribe
-
-

Routes dan Regular Expression di ExpressJs

- Regex : deretan karakter yang digunakan untuk pencarian string atau teks dengan menggunakan pola (pattern)
- Route path atau endpoint bisa terdiri : strings, string patterns, or regular expressions.
- The characters ?, +, *, and () are subsets of their regular expression counterparts. The hyphen (-) and the dot (.) are interpreted literally by string-based paths.
- If you need to use the dollar character (\$) in a path string, enclose it escaped within ([and]). For example, the path string for requests at “/data/\$book”, would be “/data/([\\$])book”.
- Ex:

```
app.get('/', function (req, res)
{
  res.send('root')
}) ;
```

Example Regex Route

- `'/about' => /about`
- `'/random.text' => /random.text`
- `"/ab?cd" => 'acd, abcd`
- `'/ab+cd' => abcd, abbcd, abbbcd`
- `'/ab*cd' => abcd, abxcd, abRANDOMcd, ab123cd`
- `'/ab(cd)?e' => /abe, /abcde`
- `/a/ => "a"`
- `/.fly$/ => butterfly and dragonfly, but not butterflyman, dragonflyman`

Route dan parameter dari requests

- Atau segmen URL yang berfungsi untuk menangkap nilai yang ditentukan pada posisinya di URL.
- Ex :



Route path: /users/:userId/books/:bookId
Request URL:
http://localhost:3000/users/34/books/8989
req.params: { "userId": "34", "bookId": "8989" }

```
app.get('/users/:userId/books/:bookId',  
function (req, res) {  
  res.send(req.params)  
})
```

Response methods

- Object Response : Obyek yang bertanggung jawab untuk mengirim response data ke client.
- Object Response memiliki beberapa method untuk mengirim response data ke client dan hentikan siklus permintaan-respons. Berikut adalah response method yang tersedia di ExpressJs

`res.download()` => Prompt a file to be downloaded.

`res.end()` => End the response process.

`res.json()` => Send a JSON response.

`res.jsonp()` => Send a JSON response with JSONP support.

`res.redirect()` => Redirect a request.

`res.render()` => Render a view template.

`res.send()` => Send a response of various types.

`res.sendFile()` => Send a file as an octet stream.

`res.sendStatus()` => Set the response status code and send its string representation as the response body.

Contoh Route dan Response method

```
app.get('/redirect_to_admin', function (req, res) {  
  res.redirect('/admin')  
})  
  
app.get('/admin', function (req, res) {  
  res.redirect('/admin')  
})  
  
app.get('/json', function (req, res) {  
  res.json({status: 'ok', message: 'anda  
berhasil <a href="/download">Download  
</a>', })  
})
```

```
app.get('/json_errro', function (req, res) {  
  res.status(500).json({ error: 'message' })  
})  
  
app.get('/download', function (req, res) {  
  res.download('/report-12345.pdf', 'report.pdf')  
})
```


Selamat Mengerjakan...