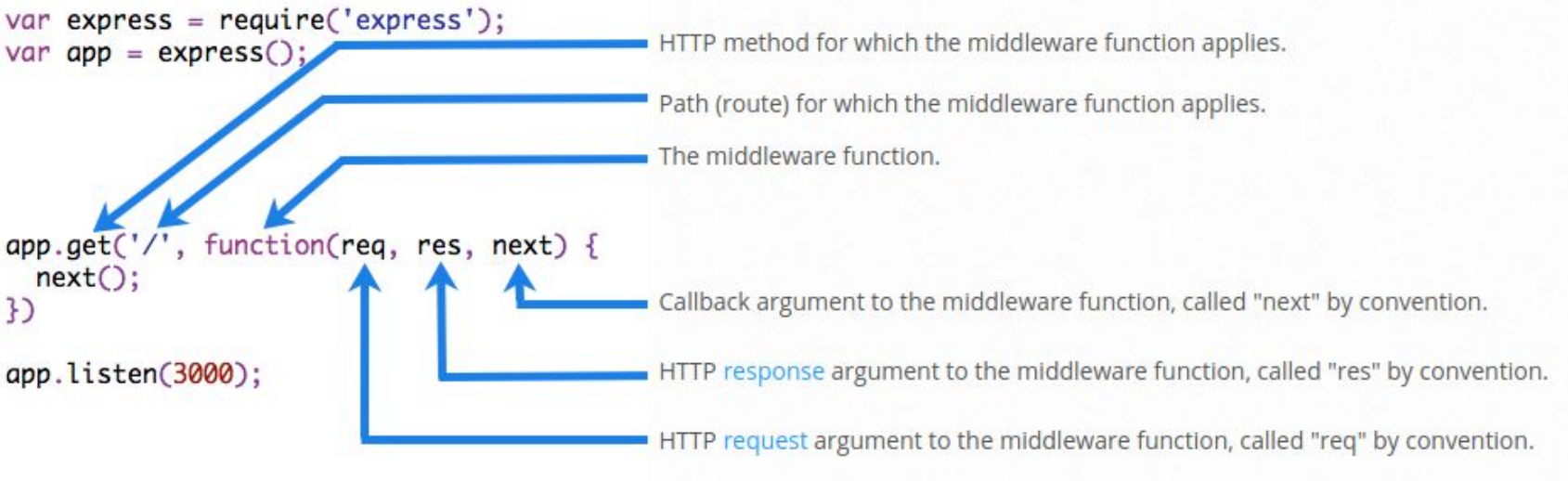
# Middleware

M02-H05

# Middleware di express

- fungsi dieksekusi di tengah atau diakhir, setelah permintaan yang masuk kemudian menghasilkan sebuah keluaran yang bisa saja keluaran akhir dilewati atau bisa digunakan oleh middleware berikutnya hingga siklusnya selesai.
- Functions middleware :
  - objek request (req),
  - objek respons (res),
  - next middleware
- Aplikasi Express : serangkaian panggilan fungsi middleware.

# Membuat Middleware di ExpressJs



# Configurable Middleware

```
//./my-middleware.js
module.exports = function (options) {
  return function (req, res, next) {
    // Implement the middleware function based on the options object
    next()
  }
  }

//app.js
var mw = require('./my-middleware.js')

app.use(mw({ option1: '1', option2: '2' }))
```

# Jenis Middleware di Express

- Application-level middleware
- Router-level middleware
- Error-handling middleware
- Built-in middleware
- Third-party middleware

# Application level middleware dan Router-level middleware

```
//Application level
var express = require('express')
var app = express()

app.use(function (req, res, next) {
  console.log('Time:', Date.now())
  next()
  })
```

```
//Router-level
var express = require('express')
var app = express()
var router = express.Router()
// a middleware function with no mount path. This
code is executed for every request to the router
router.use(function (req, res, next) {
  console.log('Time:', Date.now())
  next()

  })
// a middleware sub-stack shows request info for
any type of HTTP request to the /user/:id path
router.use('/user/:id', function (req, res, next)
{
  console.log('Request URL:', req.originalUrl)
  next()
}, function (req, res, next) {
  console.log('Request Type:', req.method)
  next()
})
```

```
// handler for the /user/:id path, which renders a special page
router.get('/user/:id', function (req, res, next) {
  console.log(req.params.id)
  res.render('special')
})

// mount the router on the app
app.use('/', router)
```

# Error-handling middleware

```
app.use(function (err, req, res, next) {
  console.error(err.stack)
  res.status(500).send('Something broke!')
})
```

# Built-in middleware

- https://github.com/senchalabs/connect#middleware

# Third-party middleware

- http://expressjs.com/en/resources/middleware.html

# Selamat melakukan lab mandiri