

# Module 6 Express JS



#### Outline

- Understanding express js framework
- Scaffolding project with express-generator
- Understanding routing and middleware



# **Express - Official site says**

EXPRESS 4.14.0

Hast, unopinionated, minimalist web framework for Node. s



## **Express - What can be developed**

- Web Applications
- API



#### **Express - Setup**

npm install express --save



#### **HelloWorld Express**

```
var express = require('express')
var app = express()
app.get('/', function (req, res) {
 res.send('Hello World!')
app.listen(3000, function () {
 console.log('Example app
listening on port 3000!')
```



### **Express Generator - Setup & Usage**

\$ npm install express-generator -g

```
$ express -h
 Usage: express [options] [dir]
 Options:
   -h, --help
                  output usage information
       --version
                     output the version number
   -e, --ejs
                    add ejs engine support
      --hbs
                    add handlebars engine support
                     add pug engine support
       --puq
   -H, --hogan
              add hogan.js engine support
   -v, --view <engine> add view <engine> support (ejslhbslhjsljadelpugltwiglvash) (defaults to jade)
   -c, --css <engine> add stylesheet <engine> support (less|stylus|compass|sass) (defaults to plain css)
       --git
                  add .gitignore
   -f, --force
                     force on non-empty directory
```



#### **Express Generator - Create new app**

express --view=jade myapp



#### **Express Generator - Introspect Project**

Review the folder structure



#### **Express Generator - Run the app**

npm start



### Routing

- Routing refers to the definition of application end points (URIs) and how they respond to client requests.
- Simply put unique address for a resource on a server





# Routing - How to use app.METHOD(PATH, HANDLER)

- app is an instance of express.
- METHOD is an HTTP request method, in lowercase.
- PATH is a path on the server.
- HANDLER is the function executed when the route is matched.



#### **Routing - Example routes**

```
app.get('/products', HANDLER)
app.post('/products', HANDLER)
app.put('/products/:id', HANDLER)
app.delete('/products/:id', HANDLER)
app.all('/products/status', HANDLER)
```



#### **Middleware**

- Functions that have access to the request <u>object</u>
   (req), the response object (res), and the next
   function in the application's request-response
   cycle.
- The next function is a function in the Express router which, when invoked, executes the middleware succeeding the current middleware.

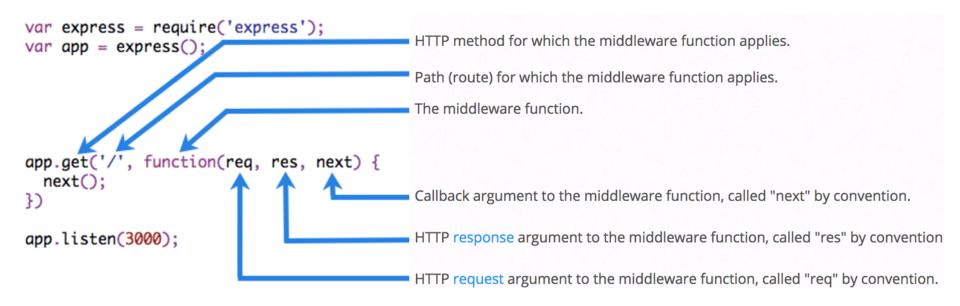


#### What Middleware can do

- Execute any code.
- Make changes to the request and the response objects.
- End the request-response cycle.
- Call the next middleware in the stack.



#### **Anatomy of Middleware**





#### **Request Object**

 Represents the HTTP request and has properties for the request query string, parameters, body, HTTP headers, and so on

Some most used properties

req.baseUrl	req.cookies	req.query
req.hostname	req.method	req.params
req.path		



#### **Response Object**

• Represents the **HTTP response** and has properties for cookies, sending responses, headers etc.,

Some most used methods

res.attachment	res.cookie	res.download
res.end	res.json	res.redirect
res.render	res.send	res.sendFile
res.sendStatus	res.set	res.status



### **Check your knowledge - 1**

- Name 2 types of applications that can be developed with express js?
- Middleware functions do not have access to req and res, true or false?
- app.GET is used to serve http get requests, true or false?
- Which concept refers to "definition of application end points"
- Name methods to end request-response cycle



### **Check your knowledge - 2**

- Which request property is used to read query params?
- Is it possible to find the current request http method?
- Which method is used to send json response to client?
- How to set header on response?
- How to set cookie on response?
- res.end vs res.send?