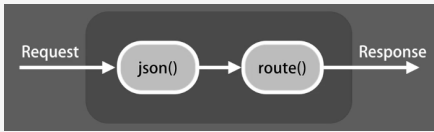


Module	Express (cont)	Built - in modules (mostly used) (cont)
Module module has exports property e.g module.exports = fn() => {} module. exports = {hello, move}	events app.use(body-parser.json()) parse body of req into json:- app.use(express.json()) serves static files:- app.use(express.static(path.join(__dirname, 'public')))	to handle events EventEmitter = require('events') emitter = new EventEmitter() emitter.on(eventName, listener) emitter.emit(eventName, [args])
Require returns module.exports	route handle is a middleware. request-processing-pipeline: Req -	file to handle file system fs = require('fs') fs.readFile(path, callback) fs.copyFile(source, target, mode)
config Storing configuration settings: default, dev & prod `config.get('property')`	Request processing pipeline  <p>express.json() will pass request to route handler and route will terminate request by sending response.</p>	system set environment vars: export PORT = 3000
Node wraps you code with function expression & invokes it, this function expression has these argument: exports, module, __dirname & __filename. Now you have access to these argument within your code. You can use these argument e.g module.exports = hello() to expose ur module	Built - in modules (mostly used) path to handle paths path = require('path') path.join(__dirname, 'public') os provide operating system info os = require('os') os.freemem() os.uptime() os.userInfo()	www w w
Express install npm i express app = require('express')	http make Node to act as webserver server = require('http') server.createServer(req, res) => {} server.listen(3000)	Debugging dbDebugger = require('debug') startDebugger = require('debug') :start')
web server app.get('/', (req, res) => { res.send('hello') }) app.listen(3000)	process Node process info - global object port = process.env.PORT 3000	Two names are will create:- app:db & app:start. You can replace console.log with dbDebugger or startDebugger. Define console output:- export DEBUG= app:start or export DEBUG= app:db or export DEBUG= app:start,app:db or export DEBUG= app:*
route params req.params query params req.query		



NPM

<code>npm init</code>	Create package.json file alt: <code>npm init --yes</code>
<code>npm i</code>	Install npm package alt: <code>npm i --save-dev</code> <code>npm i --save</code>
<code>npm list</code>	lookup dependencies & version alt: <code>npm list --depth=0</code>
<code>npm outdated</code>	check 4 outdated npm package
<code>npm update</code>	update npm package - only updates minor & patches
<code>npm un</code>	uninstall npm package
<code>npm login</code>	login in npm registry
<code>npm adduser</code>	create user on npm registry

C

By **digotetso**
cheatography.com/digotetso/

Not published yet.
Last updated 11th April, 2018.
Page 2 of 2.

Sponsored by **CrosswordCheats.com**
Learn to solve cryptic crosswords!
<http://crosswordcheats.com>