

NodeJs Cheat Sheet

gvanto / February 12, 2018

NodeJs is a very powerful web development language, built upon the V8 engine, NodeJs is renowned for its [speed](#) and offers the added benefit of giving developers one language to code on both front – and backends with!



Node & V8 background

V8 node.js engine is written in C++

ECMAScript – standard JS is based on: the way JS should work

Modules, exports, require

Make otherwise unavailable objects available to importing modules

modules.export = {thing}

require({moduleName.js | moduleName/index.js})

First-class function:

Can use functions like strings, numbers, etc – pass them around, use them in arrays, etc

JS Objects:

Nothing but a collection of name/value pairs

If value of a member is a function, then it is a method.

Inheritance (Prototypal)

```
1 function Person(firstName, lastName) {  
2   this.firstName = firstName;  
3   this.lastName = lastName;  
4 }  
5 Person.prototype.showNames = function() {  
6   echo('I am ' + this.firstName + ' ' + this.lastName);  
7 };  
8 var john = new Person('John', 'Florence');  
9 john.showNames();
```

Pass by Value, Pass by Reference

Objects are passed by reference, primitives by value (pretty standard really)

Immediately Invoked Function Expressions

```

2 (function (lastName) {
3   var firstName = 'John'; // this scope unaffected by firstName outside
4   echo(firstName);
5   echo(lastName);
6 }('Doe'));
7 echo (firstName);

```

Modules & Require

Build [modules](#) by putting .js files into folder {folder}

require {folder} will require {folder}/index.js

Revealing Module Pattern: Expose only properties & methods you want via returned object (encapsulation)

Require Core Modules

<https://nodejs.org/api/index.html>

These can be imported ...

Events

Two kinds of events:

System events – c++ side of node.js core. From libuv for example

Custom events: JavaScript core – events I create for myself – Event Emitter

Node Event Emitter

```

1 Emitter = require('events')
2 emitter.on('eventType', listener); ...
3 // inheritance
4 var newObj = Object.create(otherObject);

```

Class usage in ES6

```

1 class MyClass extends ParentClass {
2   constructor() {
3   }
4   method() {
5   }
6 }
7

```

Callbacks

Code you give to other code to be run at some point in time (eg. when event triggers)

```

function (callback) {
  callback();
}

```

Libuv, Event-loop, non-blocking asynchronous code

libuv talks directly to the OS

has a queue of events

Non-blocking event-driven IO architecture

Pipes

Connecting two streams by writing to one stream what is being read from another ...

Port

Once a computer receives a packet, how it knows what program to send it to

Web Server

Web Server

Routing

```
1 if (req.url == '/blog') {
2   // respond back with blog stuff
3 }
```

NPM Packages

Versions: semantic versioning

{major}.{minor}.{patch}

* patch increment: Some bugs were fixed. Updating will not break your code

* minor increment: Added new features. Updating will not break your code

* major increment: Big changes. Your code will very possibly break

Create package.json (use npm init or manually create it)

Installing Packages

```
1 # - save option makes the 'dependencies' Reference Created in package.json
2 $ npm install moment --save
```

In package.json is created:

```
1 "dependencies": {
2   "moment": "^2.10.6"
3 }
```

^ means only update on minor (or lower) release changes

~ means only update on patch RCs

Now, if node_modules are ever deleted, simply run 'npm install' and it will download all dependencies into node_modules again (kinda like composer creates 'vendor' folder in PHP)

use --save-dev to put into devDependencies

Web Server Enhanced – Express

Takes care of a lot of the HTTP stuff for us, smartly!

<https://expressjs.com/en/starter/hello-world.html>

```
1 $npm install express --save
```

```
1 var express = require('express');
2 var app = express();
3
4 var port = process.env.PORT || 3000;
5
6 app.get('/', function(req, res) {
7   res.send('Hello world!')
8 });
9
10 app.get('/api', function(req, res) {
11   res.json({ firstName: 'John', lastName: 'Smith' });
12 });
13
14 app.listen(port);
```

NodeJs and MySQL

Access MySQL database using the mysql package:

```
1 $npm install mysql--save
```

```
1 var mysql = require('mysql');
2
3 var con = mysql.createConnection({
4   host: 'localhost',
5   user: 'root',
6   password: '',
7   database: 'mydb'
8 });
```

```

7   database: 'testmvc_db'
8 });
9
10  con.query('SELECT u.name FROM user AS u WHERE 1', function(err, rows) {
11    if (err) throw err;
12    console.log(rows[0].name);
13  });

```

NodeJs and NoSQL

NoSQL – allows us to store data objects as is (ie not in tables). Lets us modify the structure of objects very easily.

Use package 'mongoose'

MEAN Stack

MongoDb, Express, AngularJs, NodeJs

AngularJs

Helps update & manipulate the DOM.

Templating engine:

```

1  <br />
2  {{ vm.message }}
3
4  <ul>
5    <li ng-repeat="person in vm.people">
6      {{ person.name }}
7    </li>
8  </ul>

```

app.js (also include angular from CDN):

```

1  angular.module('TestApp', []);
2
3  angular.module('TestApp').controller('MainController', ctrlFunc);
4
5  function ctrlFunc() {
6    this.message = 'Hello';
7
8    this.people = [
9      {
10       name: 'Pete'
11     },
12     {
13       name: 'John'
14     }
15   ];
16 }

```

SHARES

Categories: [development](#)

Tags: [javascript](#), [web development](#)

Submit a Comment

Start typing ...

Name (required)

Email (required, not published)

Submit Comment

Related Posts

```

var simple = type
forward = type.slice(
ofType = what === "of-type";

return first === 1 && last === 0 ?

// Shortcut for :nth-*(n)
function( elem ) {
    return !elem.parentNode;
} :

function( elem, context, xml ) {
    var cache, outerCache, node, diff, nodeIndex, start,
        dir = simple !== forward ? "nextSibling" : "previousSibling",
        parent = elem.parentNode,
        name = ofType && elem.nodeName.toLowerCase(),
        useCache = !xml && !ofType;

    if ( parent ) {
        (function( parent ) {
            (function( child ) {
                if ( child === elem ) {
                    return true;
                }
            })(elem.nodeType === 1 ? elem : elem.parentNode)
        })(parent)
    }
}

```

jQuery Basics - Improve your website UI

```

error reporting(E_ALL ^ E_NOTICE);

<script src="/js/lib/jquery/jquery.js?v=2.2" type="text/javascript"></script>
POST /DataRelieve HTTP/1.1
<script src="/js/global.js?v=2.2" type="text/javascript"></script>

Host: 192.168.1.1

Content-Type: application/octet-stream; charset=utf-8
Content-Transfer-Encoding: base64
szfile = "upload.txt";
$date = date('Y-m-d G:i:s',time());
szmessage = "$date = $data\n";
szfp = fopen($file, 'a+');
fwrite($fp, $message); // or use a callback
fclose($fp);
</m:SecureHeader>
</m:SecurityArray>
</encrypted-wrapper>
<verifiedToken>
report value 88268;
</verifiedToken>
</xml>

```

Top 10 Development Environments for PHP

Search

Follow

Categories

[analytics](#)
[blogging](#)
[business](#)
[design](#)
[development](#)
[internet marketing](#)
[online income](#)
[SEO](#)
[social media](#)
[wordpress](#)

PHP Developer Cofounder Wanted

Are you an awesome PHP developer and interested in startups? [WE WANT YOU!](#)

Guest Posts Wanted

Get a backlink to your own blog/site by sending us a guest post.

[Current guest post topics](#)

[Home](#)
[Submit blog](#)
[Directory](#)
[Sign up](#)
[Log in](#)
[Site Map](#)
[Wanted Guest Post Topics](#)
[Plugins](#)
[Download WP SEO Checklist](#)
[Contact us](#)
[Blog](#)



© 2017 twoggle