

JavaScript LIBRARIES

Moment.JS

Moment is a time library work in both Server and Client side.

References

[/https://momentjs.com/docs](https://momentjs.com/docs)

PROJECTS

Wael Zoaiter Personal Website : socket.js in the client-side

MUSTACH.JS

Template engine language.

References

[/https://github.com/janl/mustache.js](https://github.com/janl/mustache.js)

Heroku

References

<https://devcenter.heroku.com/articles/getting-started-with-nodejs>

Projects

/Wael Zoaiter Personal Website : <https://mysterious-atoll-68392.herokuapp.com>

Setup Heroku App

In server.js:

Add `var port = process.env.PORT || 8000;` and listen to it.

In mongoose.js:

Add `mongoose.connect(process.env.MONGODB_URI || 'mongodb://localhost:27017/ToDoApp');`

In package.json:

Add “start” to tell Heroku How to start, “engines” to tell Heroku what NodeJS version I use.

```
6  "scripts": {
7    "start": "node server/server.js",
8    "test": "mocha server/**/*.test.js",
9    "test-watch": "nodemon --exec 'npm test'"
10 },
11 "engines": {
12   "node": "8.9.4"
13 },
```

Using mLab MongoDB to integrate MongoDB with Heroku:

CLI

```
heroku create // create Heroku app
heroku addons:create mongolab:sandbox // configure mLab with our Heroku application
heroku config // to show all the configuration addons
git commit -am 'Setup app for heroku' // commit to git and setup to Heroku
git push // push the app to GitHub
git push heroku master // push the app to Heroku
heroku logs // to show the logs and errors
heroku open // to open Heroku application
```

ES6

The last Release of JavaScript language

References

<http://es6-features.org>

Projects

Wael Zoaiter Personal Website: `user.js` file `testClass.js`

ES6

```
Let {MongoClient} = require('mongodb');
```

{ } in ES6 it called Destructuring .. and it pull of a property from mongodb library or from object and define it as a variable.

And it can contain many variables to many properties:

```
Let {MongoClient, MongoError} = require('mongodb');
```

```
(res, req) => { ... }
```

```
(res, req) => user;
```

```
var obj = {  
  from,  
  user  
}
```

Vanilla JavaScript

```
Let MongoClient = require('mongodb').MongoClient;
```

```
function(res, req) { ... }
```

```
function(res, req) { return user }
```

```
var obj = {  
  from: from,  
  user: user  
}
```

`.then(function when the promise get resolved, function when the promise get rejected)`

Node-MySQL

References

<https://www.npmjs.com/package/mysql>

CouchDB

References

<https://www.npmjs.com/package/nano>

<http://docs.couchdb.org/en/2.1.1/index.html>

GIT & GITHUB

CLI

Initialize new Repository:

```
git init
```

Get the state of the file on the track:

```
git status
```

Create .gitignore file to ignore the files that I don't want to upload it, Write inside it the name of the file or folder that I want to ignore:

```
e.g. node_modules/
```

Add everything in the track to my commit:

```
git add .
```

Make the commit with a message to describe it:

```
git commit -m 'Init Commit'
```

Push the commit to the repository:

```
git push
```

Create new repository from github.com:

```
New Repository
```

Execute the commands after creating the repository:

```
git remote add origin git@github.com:....
```

```
git push -u origin master
```

YARGS

Node js Module

WHAT IS IT

Node.js module to make command line parameters.

INSTALL

```
npm install yargs@4.8.1 --save
```

References

<https://www.npmjs.com/package/yargs>

Setup

```
2  const yargs = require('yargs');
3
4  const argv = yargs // Make a command parameters
5    .options({
6      a: { // Create command '-a'
7        demand: true, // it's required
8        alias: 'address', // another word '--address' to use it
9        describe: 'Address to fetch weather for',
10       string: true // type of it
11     }
12   })
13   .help() // Make a help describtin for it
14   .alias('help','h')
15   .argv; // Add this option to 'argv' variable
16
17  var encodedAddress = encodeURIComponent(argv.address);
18
19  console.log(encodedAddress);
20
21  MINGW32/d/Web Design/WebDevInit/WeatherApp
22  wael@wael-PC MINGW32 /d/Web Design/WebDevInit/WeatherApp
23  $ node server -a 'Midan Damascus Syria'
24  Midan%20Damascus%20Syria
25  Address: Al Midan, Damascus, Syria
26  latitude: 33.4913481
27  longitude: 36.2983286
28  wael@wael-PC MINGW32 /d/Web Design/WebDevInit/WeatherApp
29  $ node server -a 'Midan Damascus Syria'
```

axios

Node js Module

What is it

Make HTTP request for the browser and node.js .. Promise based

Install

npm install axios@0.13.1 --save

References

<https://www.npmjs.com/package/axios>

Setup

CRYPTOJS

What is it?

Node js library to crypt and hash data

Install

npm install crypto-js --save

References

<https://www.npmjs.com/package/crypto-js>

JWT

What is it?

JSON Web Token

Install

npm install jsonwebtoken --save

References

<https://www.npmjs.com/package/jsonwebtoken>
[/https://jwt.io](https://jwt.io)

Sign and Verify