NODE PROGRAM

2014

PRESENTER

Azat Mardan

@azat_co
webapplog.com

AGENDA DAY 1

1. Introductions
2. JavaScript Fundamentals
3. Break
4. Node.js Fundamentals
5. Debugging and other tools
6. NPM: utils, fs, buffer, url, path
7. Learn You Node Workshop
8. Lunch
9. Hello World Server
10. Heroku
11. Stream Adventure Workshop
12. Functional JavaScript Workshop

AGENDA DAY 2

1. NoSQL and MongoDB
2. REST API
3. Break
4. Express.js Fundamentals
5. Jade and Stylus
6. Express REST API
7. ExpressWorks Workshop
8. Lunch
9. Production
10. Elective personal projects

JAVASCRIPT FUNDAMENTALS

EXPRESSIVNESS

LOOSE TYPING

OBJECT LITERAL NOTATION

FUNCTIONS

ARRAYS

PROTOTYPAL NATURE

CONVENTIONS

- Names
- Indentation
- Semi-colons
- Comma-first style

NO MODULES

CLOSURES

GLOBAL AND PRIVATE VARS

IMMEDIATELY-INVOKED FUNCTION EXPRESSIONS

```
;(function(){
    ...
}())
```

KEYWORD 'THIS'

PITFALLS

- Leaking vars into global space
- Using wrong "this" context
- Using wrong == or ===
- Missing break in a switch case
- Not using "return" when needed
- Dealing with pseudo-classical inheritance

NODE.JS FUNDAMENTALS

EVENT LOOP

NON BLOCKING I/O

READ-EVAL-PRINT LOOP (A.K.A. CONSOLE) IN NODE.JS

LAUNCHING NODE.JS SCRIPTS

NODE.JS PROCESS INFORMATION

ACCESSING GLOBAL SCOPE IN NODE.JS

EXPORTING AND IMPORTING MODULES

require()

module.exports

BUFFER IS A NODE.JS SUPER DATA TYPE

__DIRNAME VS. PROCESS.CWD

CORE MODULES

PATH

FS

NET

HTTP

URL

STRINGDECODER

CRYPTO

DEBUGGING AND OTHER TOOLS

- Node Inspector
- WebStorm
- nodemon/supervisor
- Sublime Text Plugins

YOULEARNNODE

```
$ sudo npm install -g learnyounode
```

WORKSHOP TIME

NPM

NPM INSTALL

```
$ npm install express
$ npm install express@4.2.0
$ npm install express --save
$ npm install express --save-dev
```

PACKAGE.JSON

```
 {
  "name": "stream-adventure",
  "version": "2.5.0",
  "description": "an educational stream adventure",
  "bin": {
    "stream-adventure": "bin/cmd.js"
  },
  "dependencies": {
    "hyperquest": "~0.1.6",
    "ws": "~0.4.25"
  },
  "devDependencies": {
    "tape": "~2.3.0"
  }
}
```

NODE_MODULES

NPM INSTALL -G

HELLO WORLD SERVER

Page 153 in Rapid Prototyping with JS (PDF)
6.1 Node.js

DEPLOYING TO HEROKU

Page 158 in Rapid Prototyping with JS (PDF
6.1.4 Deploying "Hello World" to PaaS

SSH
Procfile

STREAM

STREAM ADVENTURE

```
$ sudo npm install -g  stream-adventure
$ stream-adventure
```

WORKSHOP TIME

FUNCTIONAL JAVASCRIPT

```
$ sudo npm install -g functional-javascript-workshop
$ functional-javascript-workshop
```

WORKSHOP TIME!

Q&A

CHAT SERVER

test.js
mb-server.js

Page 146 (print)

MONGODB REST API

(memory store)

MESSAGE BOARD

(memory store)

http://bit.ly/1usviBi

ENDPOINTS

POST /messages.json
GET /messages.json

MB-SERVER.JS

POST /messages/create.json
GET /messages/list.json

util.inspect
querystring.parse
exports.getMessages
exports.addMessage

CURL

```
$ curl http://127.0.0.1:1337
$ curl -X POST -d 'name=azat&message=hi' http://127.0.0.1:1337
```

DAY 2

AGENDA DAY 2

- Front-end and back-end overview
- NoSQL and MongoDB
- REST API
- Break
- Express.js Fundamentals
- Jade and Stylus
- Express REST API
- ExpressWorks Workshop
- Lunch
- Production
- Elective personal projects

FRONT-END AND BACK-END

- Traditional web
- Thick client / XHR web

MONGODB BASICS

Commands:

- mongo
- mongod

Methods:

- find
- insert
- save
- remove
- update

MONGO SHELL

```
> use dbname
> show collections
```

var uri = process.env.MONGO

NOSQL AND MONGODB

Page 168 in Rapid Prototyping with JS (PDF)
Page 154 in print
6.4 MongoDB

CHAT REST API

(with MongoDB)

http://bit.ly/1AkobeN

MONGOHQ

```
var uri = process.env.MONGOHQ_URL || 'mongodb://@127.0.0.1:27017'
```

```
 $ heroku addons:add mongohq:sandbox
```

EXPRESS.JS FUNDAMENTALS

APP STRUCTURE

1. Includes
2. Instantiations
3. Configurations
4. Middleware
5. Routes
6. Boot-up

MIDDLEWARE

Almost always use:

- static
- body-parser
- express-session
- compression
- all from cheatsheet*

* http://bit.ly/Us2qbP

EXPRESS.JS STACK

- Jade
- Stylus
- Mongoose or Mongoskin

GENERATOR

```
$ npm install -g express-generator
```

OTHER FRAMEWORKS

- Hapi
- Sails
- Derby

many others at
http://nodeframeworks.com

EXPRESS.JS HELLO WORLD

REST API PROJECT

https://github.com/azat-co/rest-api-express

Description: http://bit.ly/1jy30tn

JADE AND STYLUS

EXPRESSWORKS WORKSHOP

PRODUCTION

NODE.JS STACK

- Heroku
- AWS with Nginx + Varnish Cache + Upstart scripts
- MongoDB or MongoHQ/MongoLab: Mongoskin, Mongoose
- Logging with Winston, Elastic Kibana or Papartrailapp

TIPS

- Deployment scripts with Salt (saltstack.com)
- Increase MaxSockets
- Lock versions
- Don't trust user input
- Have good error handling in place
- Use upstart or forever

ELECTIVE PERSONAL PROJECTS

- Full-stack*
- Integration
- Deployment

* Come up with an idea, e.g., todo app

INTEGRATION

Introduction to OAuth

DEPLOYMENT

Practical Node.js chapter 11

END

If it's not fun, it's not Node.js!

@azat_co