Before we begin...

- Videos On!

Welcome

Agenda

- Node
- NPM
- HTTP Server

Before Node

JavaScript Engines

- JavaScript Engines are pieces of software that execute JavaScript code
- They are typically created by Web Browser Vendors:
  - V8 is for Google Chrome
  - SpiderMonkey is for Firefox
  - JavaScriptCore is for Safari
  - Chakra is for Internet Explorer (and a fork of it was for Edge)

Why are we talking about this?

- Node.js is the JavaScript Engine (V8)
  - Taken out of the browser, with a little bit extra
- It's the same programming language as the JavaScript you know, it just has extra back-end related functionality
  - e.g. Work with file system, work with databases etc.

Node

What is Node?

- A back-end programming language
- Open-source and cross-platform
- Created in 2009 by Ryan Dahl while he was working at Joyent
- Can be run interactively (as a REPL) or as files
- Very popular (Uber, Netflix, LinkedIn, Twitter, Paypal, eBay etc.)

When we install Node

We also install command line tools too:

- `node`
  - We use this to open a REPL and to run files
- `npm`
  - We use this to manage our projects

How to run it

```
# To open up a REPL (CTRL + C to quit)
node

# To run a file
node main.js
```

NPM

What is NPM?

- A command line tool
- The Node Package Manager
- The largest software library (package registry)
  - With over 1.5 million packages
- A tool for managing software projects. We use it to:
  - Install dependencies (e.g. libraries and frameworks)
  - Manage dependencies
  - Structure our applications
  - Organise scripts for the app

NPM Commands

npm init

Used to set up a new Node/NPM project (we run this once at the start)

```
npm init
```

This will create a `package.json` file, which describes our project, its scripts and its dependencies

npm install

How we install packages into our projects

```
# Install all required packages
npm install

# Install and save a package as a dependency
npm install --save package_name

# Install and save a package as a dependency for development
npm install --save-dev package_name
```

npm run

In our `package.json` file, we can add scripts for common tasks we need to perform. We can run these using `npm run`

```
npm run script_name

# Run the "start" script in package.json
npm run start
```

HTTP Server

What is HTTP Server?

- It is a command line tool to create simple servers
  - It doesn't require any configuration
- We install it on our computer using NPM

Why do we need it?

- Because we can only speak to APIs and AJAX using URLs
  - So far, everything has just been based on files in the browser
- HTTP Server will create a server using `localhost`
  - Which is a server only available on our computers giving us access to special browser features (AJAX, Location etc.)

Let's install it!

```
npm install -g http-server
```

This will install the `http-server` command line tool. It'll be available everywhere on our computer

Let's run it!

```
# Create a server from the current directory
http-server .

# Create a server on a different port
http-server . -p 8081
```

We close down the server using CTRL + C

Review

That's all for tonight!

Homework

- NodeSource NPM Guide
- An Introduction to NPM
- Read JavaScript.info's Promise pages
  - And their promise chaining page and error handling page

What's Next?

- Asynchronous Programming
- Synchronous Programming
- Promises
- Modules

Thank You!