Back to Week 4 | Lessons | Prev | Next

Exercise (Instructions): NPM Scripts Part 2

Objectives and Outcomes

In this exercise you will learn to build a distribution folder containing the files that can be deployed on a web server hosting your project. This distribution folder would be built from your project files using various NPM packages and scripts. At the end of this exercise, you will be able to:

- Clean out a folder using the clean NPM module.
- Copy files from one folder to another
- Prepare a minified and concatenated css file from all the css files used in your project
- Prepare an uglified and concatenated JS file containing all the JS code used in your project

Cleaning up a Distribution Folder

- Install the *rimraf* npm module by typing the following at the prompt:

```
npm install --save-dev rimraf
```

- Then, set up the following script:

```
    "clean": "rimraf dist",
```

Copying Fonts

- Your project uses font-awesome fonts. These need to be copied to the distribution folder. To help us do this, install the *copyfiles* NPM module globally as follows:

```
npm -g install copyfiles
```

Remember to use *sudo* on mac and Linux.

- Then set up the following script:

```
    "copyfonts": "copyfiles -f node_modules/font-awesome/fonts/* dist/fonts",
```

Compressing and Minifying Images

- We use the *imagemin-cli* NPM module to help us to compress our images to reduce the size of the images being used in our project. Install the *imagemin-cli* module as follows:

```
npm -g install imagemin-cli
```

Remember to use *sudo* on mac and Linux.

- Then set up the following script:

```
    "imagemin": "imagemin img/* -o dist/img",
```

Preparing the Distribution Folder

- Open *.gitignore* and update it as follows. We do not want the dist folder to be checked into the git repository.

```
node_modules
dist
```

- Then, install the *usemin-cli, cssmin, uglifyjs* and *htmlmin* NPM packages as follows:

```
npm install --save-dev usemin-cli cssmin uglifyjs htmlmin
```

- Add the following two scripts to the package.json file:

```
    "usemin": "usemin contactus.html -d dist --htmlmin -o dist/contactus.html && usemin aboutus.html -d dist --htmlmin -o dist/aboutus.html && usemin index.html -d dist --htmlmin -o dist/index.html",
    "build": "npm run clean && npm run imagemin && npm run copyfonts && npm run usemin"

```

- Open *index.html* and surround the css links inclusion code as follows:

```
    <!-- build:css css/main.css -->
    <link rel="stylesheet" href="node_modules/bootstrap/dist/css/bootstrap.min.css">
    <link rel="stylesheet" href="node_modules/font-awesome/css/font-awesome.min.css">
    <link rel="stylesheet" href="css/bootstrap-social.css">
    <link href="css/styles.css" rel="stylesheet">
    <!-- endbuild -->
```

- Do the same change in *aboutus.html* and *contactus.html*

- Similarly, open *index.html* and surround the js script inclusion code as follows:

```
    <!-- build:js js/main.js -->
    <script src="node_modules/jquery/dist/jquery.min.js"></script>
    <script src="node_modules/tether/dist/js/tether.min.js"></script>
    <script src="node_modules/bootstrap/dist/js/bootstrap.min.js"></script>
    <script src="js/scripts.js"></script>
    <!-- endbuild -->
```

- Do the same change in *aboutus.html* and *contactus.html*
- To build the distribution folder, you can type the following at the prompt:

```
npm run build
```

- This will build the *dist* folder containing the files that are a self-contained version of your project. You can now copy the contents of this folder to a web server that hosts your website.
- After verifying that the dist folder is built correctly, you can now do a git commit with the message "NPM Scripts Part 2"

Conclusions

In this exercise, you learnt the various steps to build the project for deployment using NPM scripts.

Mark as completed

👍 👎 ⚑