**BROWSER COMPATIBILITY AND TRANSPILATION**

**Review**

In this lesson, you learned about browser compatibility and transpilation. Let’s review the key concepts:

* ES5 — The old JavaScript version that is supported by all modern web browsers.
* ES6 — The new(er) JavaScript version that is *not* supported by all modern web browsers. The syntax is more readable, similar to other programming languages, and addresses the source of common bugs in ES5.
* caniuse.com — a website you can use to look up HTML, CSS, and JavaScript browser compatibility information.
* Babel — A JavaScript package that transpiles JavaScript ES6+ code to ES5.
* npm init — A terminal command that creates a **package.json** file.
* **package.json** — A file that contains information about a JavaScript project.
* npm install — A command that installs Node packages.
* babel-cli — A Node package that contains command line tools for Babel.
* babel-preset-env — A Node package that contains ES6+ to ES5 syntax mapping information.
* **.babelrc** — A file that specifies the version of the JavaScript source code.
* "build" script — A **package.json** script that you use to tranpsile ES6+ code to ES5.
* npm run build — A command that runs the build script and transpiles ES6+ code to ES5.

For future reference, here is a list of the steps needed to set up a project for transpilation:

1. Initialize your project using npm init and create a directory called **src**
2. Install babel dependencies by running
3. npm install babel-cli -D

npm install babel-preset-env -D

1. Create a **.babelrc** file inside your project and add the following code inside it:
2. {
3. "presets": ["env"]

}

1. Add the following script to your scripts object in **package.json**:

"build": "babel src -d lib"

1. Run npm run build whenever you want to transpile your code from your **src** to **lib** directories.