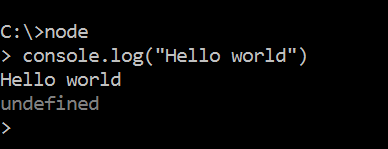
**Magenic Masters React: Environment Setup**

**Required Installations**

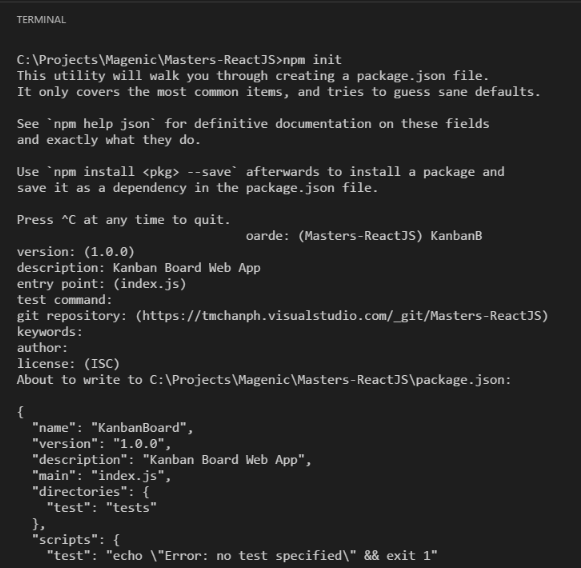
1. Install [NodeJs](https://nodejs.org/en/)
   1. Download and install the current version (v7.4.0)
   2. To test if Node is working, open command and enter “node”
   3. Inside node, you can enter various JS expressions and node should evaluate it (see screenshot below)



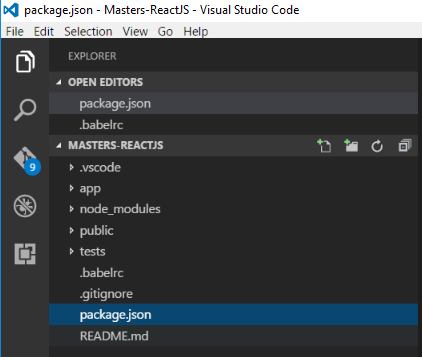
1. You may use your favorite JS editor application (Atom, Sublime Text or Visual Studio). I recommend using Visual Studio code for stability, simplicity and JSX support
   1. Download [Visual Studio Code](https://code.visualstudio.com/)

**Creating New Project**

1. Open command and go to target project folder (ex. D:\React\HelloWorld)
2. Type “npm init” and enter project settings according to your requirements. Just hit enter for default values



1. Once done, a package.json should be present on your project folder.



Installing Project Dependencies (all command line commands)

1. Install webpack packages

npm install --save-dev webpack

npm install --save-dev webpack-dev-server

npm install --save-dev babel-loader'

1. Install babel packages and setup .babelrc file

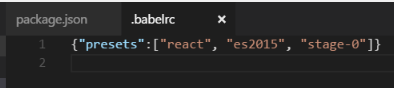
npm install --save-dev babel-cli

npm install --save-dev babel-preset-es2015

npm install --save-dev babel-preset-react

npm install --save-dev babel-preset-stage-0

echo {"presets":["react", "es2015", "stage-0"]} > .babelrc



1. Install react packages

npm install --save react

npm install --save react-dom

1. Install flux package (for Flux application architecture implementation)

npm install --save flux

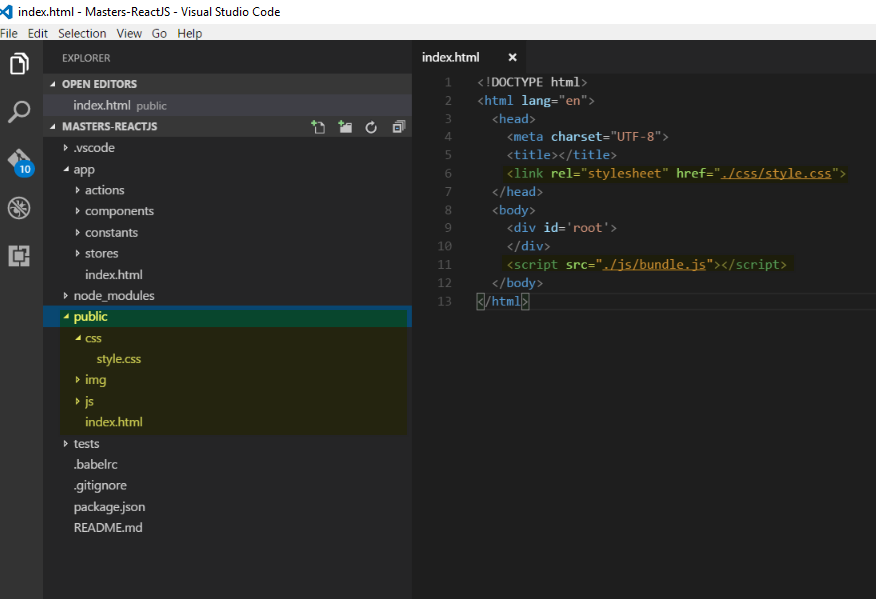
1. Install other useful libraries, Bootstrap (and jQuery) and lodash

npm install –save bootstrap jquery lodash

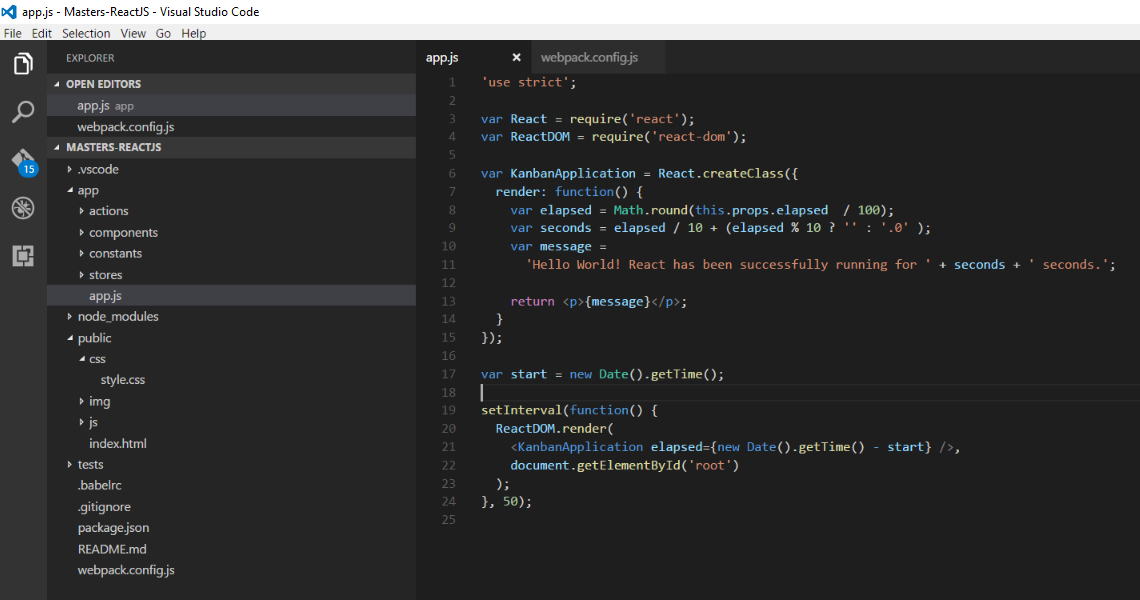
1. Your package.json should look like this after these installations



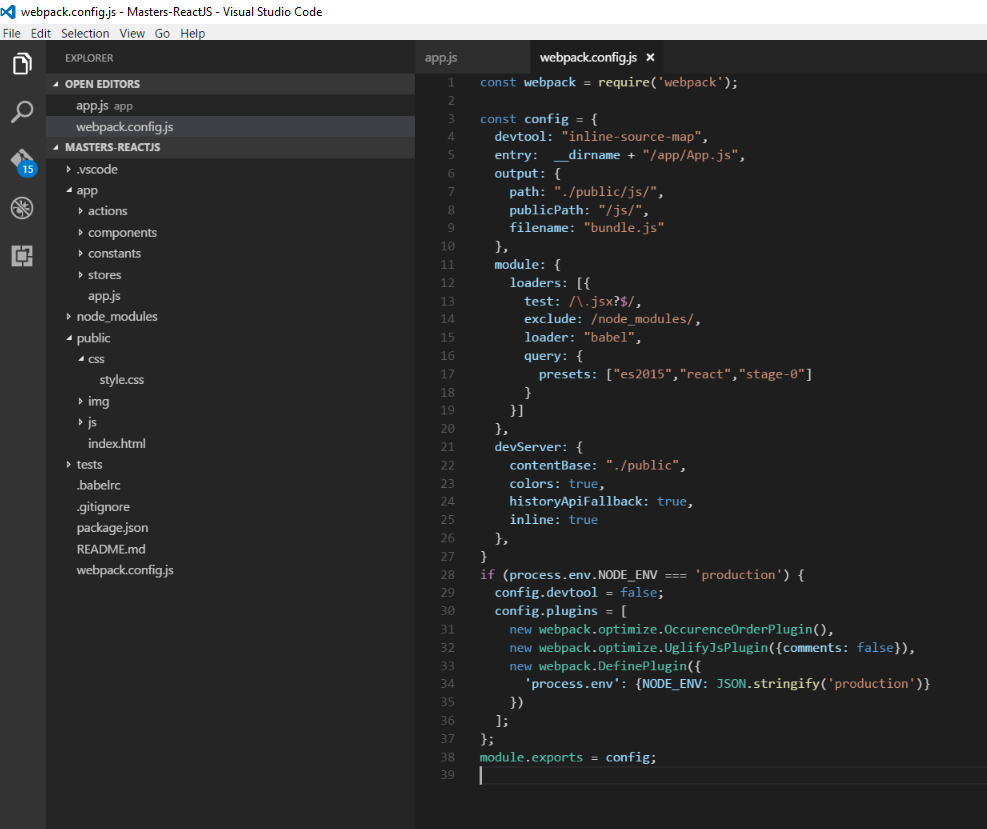
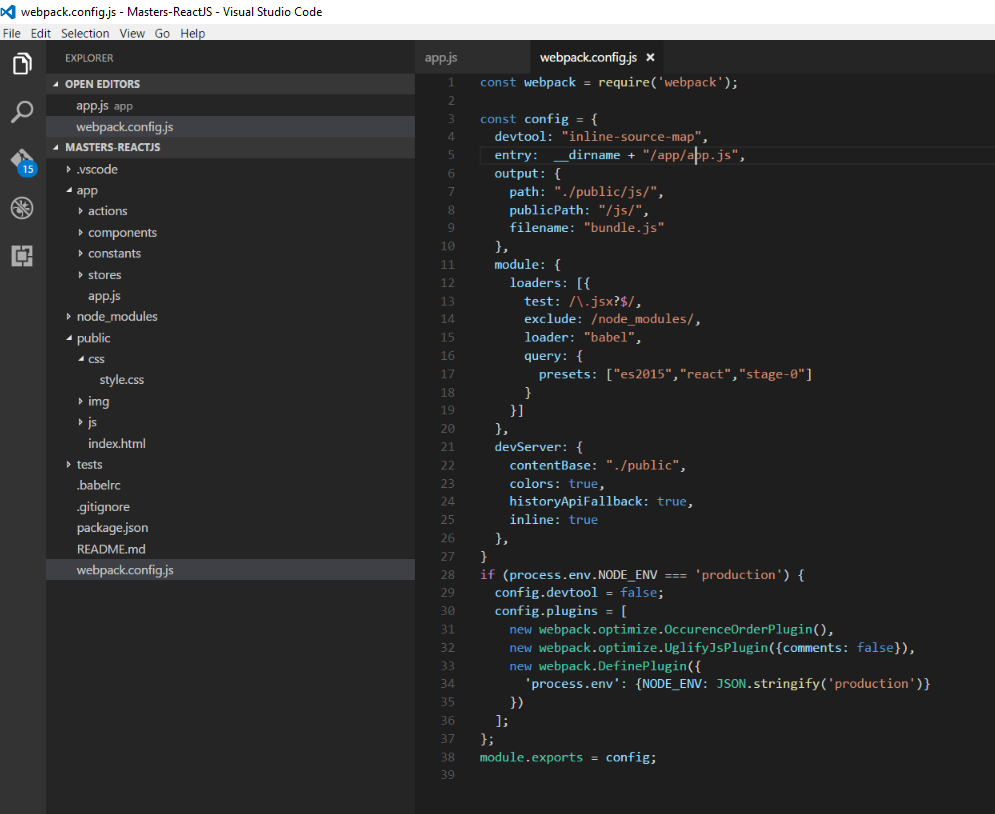
1. Create your public directories and public assets (index.html, style.css).



1. Create you app directory and first React app (app.js)



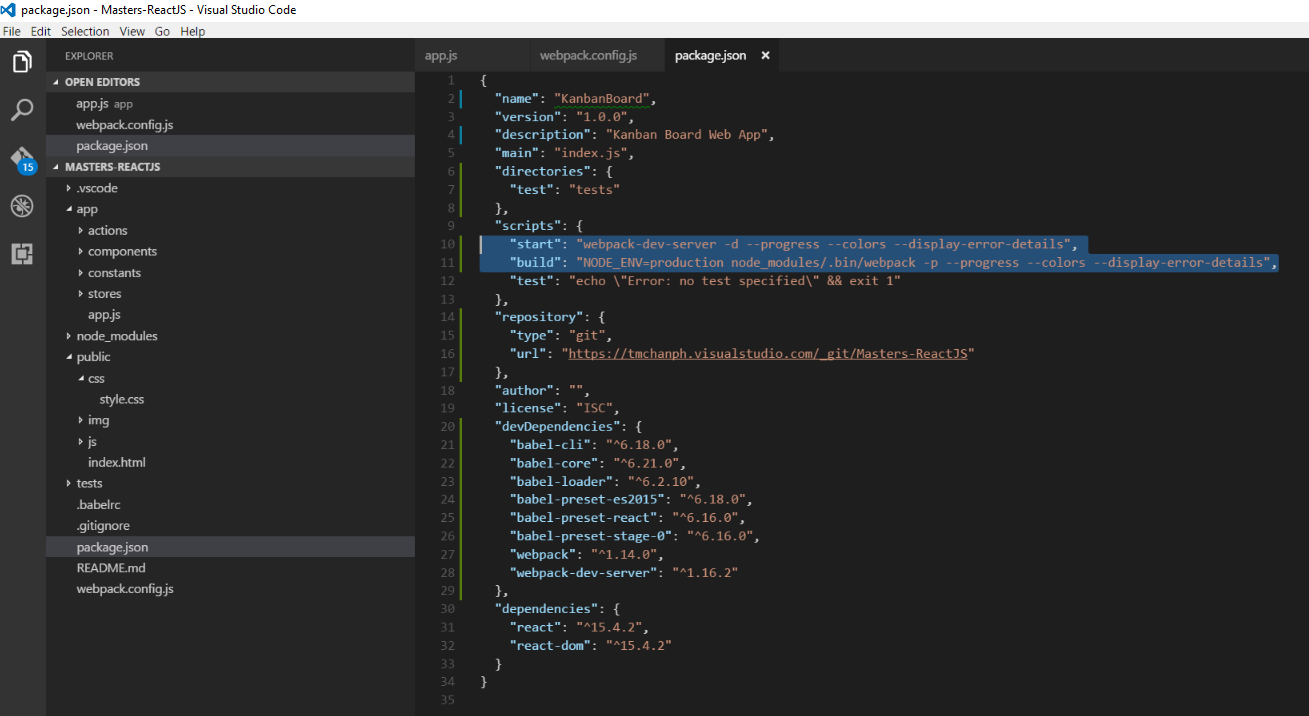
1. Create your webpack.config.js file based on your directories



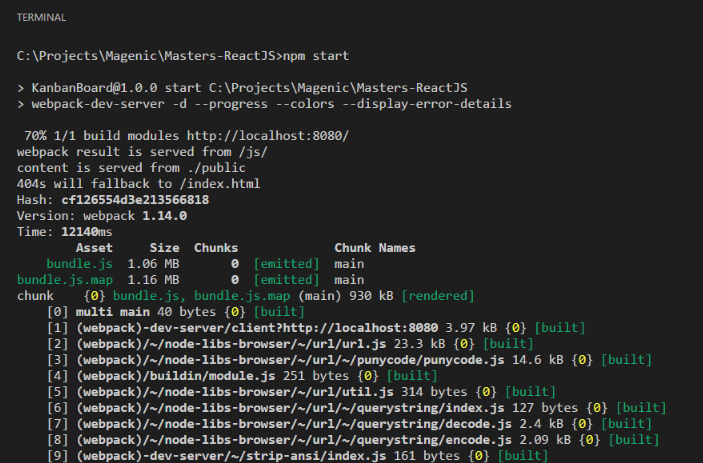
1. Add start and build npm scripts in your package.json

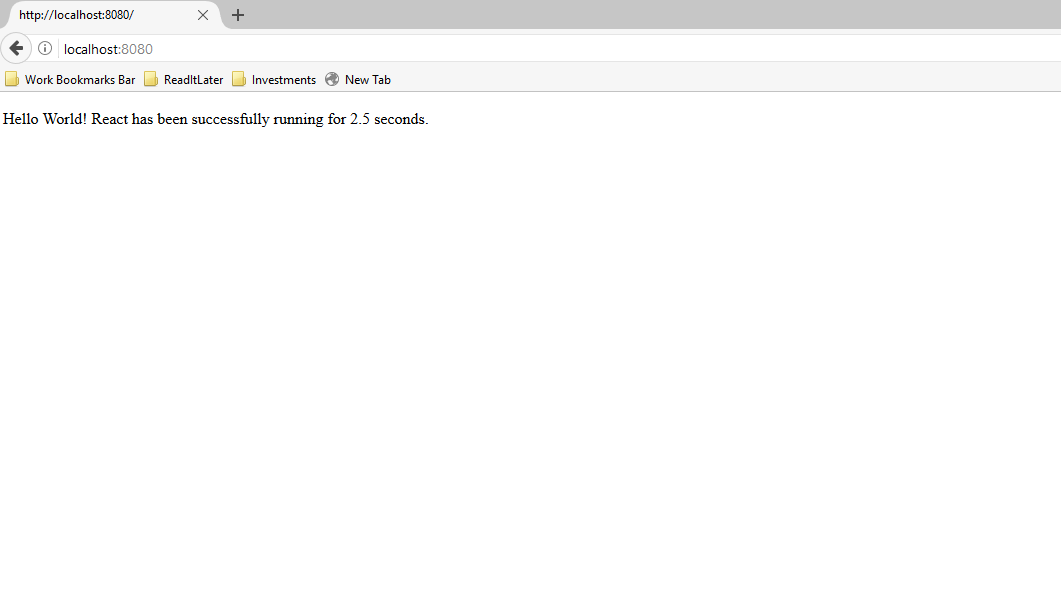
"start": "webpack-dev-server -d --progress --colors --display-error-details",

"build": "NODE\_ENV=production node\_modules/.bin/webpack -p --progress --colors --display-error-details",



1. Start your webpack-dev-server and browse your first React application.





**Note**: Each library and its use will be discussed during the session.