



Babeljs & Webpack

- Concepts
- Quizes - Recap
- Assignment

BabelJS



What is Babel?

- Babel is a JavaScript transpiler
- Babel is a toolchain that is used to convert ES2015+ code into a backwards compatible version of JavaScript in current and older browsers or environments.



What can Babel do for you?

- Transform syntax
- Polyfill features that are missing in your target environment
- Source code transformations
- Can convert JSX syntax
- Can strip out type annotations



Installation

```
npm install --save-dev @babel/core @babel/cli
```

<https://babeljs.io/setup#installation>



Usage

Babel comes with a built-in CLI which can be used to compile files from the command line.

Compile Files

```
babel script.js --out-file script-compiled.js
```

Compile Directories

```
babel src --out-dir build
```

npm scripts

```
"scripts": {  
  "build": "babel src --out-dir lib"  
},
```



Config Files

Configuration File Types

- Project-wide configuration
 - `babel.config.json` files, with the different extensions
- File-relative configuration
 - `.babelrc.json` files, with the different extensions

```
{  
  "presets": [  
    [  
      "@babel/preset-env",  
      {  
        "useBuiltIns": "usage",  
        "corejs": 3  
      }  
    ]  
  ]  
}
```



Plugins

- Out of the box Babel doesn't do anything.
- It basically acts like `const babel = code => code`.
- You will need to add plugins for Babel to do anything.

ES2015

`arrow-functions`

`block-scoped-functions`

`block-scoping`

`classes`

`Destructuring`

ES2018

`async-generator-functions`

`dotall-regex`

`named-capturing-groups-regex`

`object-rest-spread`

`optional-catch-binding`

`unicode-property-regex`



Presets

Don't want to assemble your own set of plugins? No problem! Presets can act as an array of Babel plugins or even a sharable options config

Official Presets

- @babel/preset-env
- @babel/preset-react
- @babel/preset-flow
- @babel/preset-typescript



@babel/preset-env

@babel/preset-env is a smart preset that allows you to use the latest JavaScript without needing to micromanage which syntax transforms (and optionally, browser polyfills) are needed by your target environment(s). This both makes your life easier and JavaScript bundles smaller!

Installation

```
npm install --save-dev @babel/preset-env
```



Options

Both plugins and presets can have options specified by wrapping the name and an options object in an array inside your config.

- **targets**
- **debug**
- **include**
- **exclude**
- **useBuiltIns**

```
{
  "presets": [
    ["@babel/preset-env", {
      "loose": true,
      "modules": false
    }]
  ]
}
```

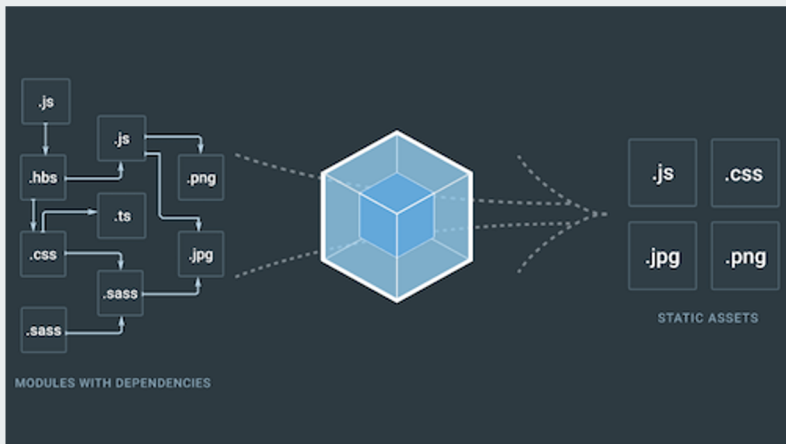


Browserslist Integration

- Targets Browsers - See browserlist coverage > 0.25%
- You can read about how it works in the browserlist documentation
(<https://github.com/browserslist/browserslist>) not dead

Webpack

What is Webpack?



- Webpack is a module bundler. It takes modules with dependencies and generates static assets representing those modules
- You can use webpack for both your client side JS files as well as your server side node JS files.



What can Webpack do for you?

- Manage dependencies
 - Require
 - Import (e.g. CJS, AMD, ES6, ...)
- Build tasks - convert and preprocess
 - Minify
 - Combine
 - sass/less conversion
 - Babel transpile
- It combines the build system and module bundling



Installation

```
npm install --save-dev webpack webpack-cli
```




Core Concepts

- Entry
- Output
- Loaders
- Plugins
- Mode



Entry

- Define a module for webpack to begin with.
- Generates Dependency graph.
- By default, ./src/index.js

```
// webpack.config.js
```

```
module.exports = {  
  entry: './path/to/my/entry/file.js'  
};
```



Output

- Defines where to emit the bundles it creates and how to name these files
- By default, ./dist/main.js

```
// webpack.config.js
```

```
module.exports = {  
  mode: 'development',  
  entry: './src/index.js',  
  output: {  
    path: path.resolve(__dirname,  
      'dist'),  
    filename: 'app.bundle.js'  
  }  
};
```



Loaders

- Webpack understands js & json files
- Loaders allow webpack to process other files

```
// ...  
module: {  
  rules: [  
    {  
      test: /\.txt$/,  
      use: 'raw-loader'  
    }  
  ]  
}  
// ...
```



Plugins

- Bundle optimization
- Asset management
- Inject of new variables

```
// ...  
plugins: [  
    new HtmlWebpackPlugin({template:  
        './src/index.html'})  
]  
// ...
```



Mode

- development
- production
- none

```
// ...  
mode: 'production'  
// ...
```

webpack-dev-server



What is development

- `npm i webpack-dev-server`
- The `webpack-dev-server` provides you with a simple web server and the ability to use live reloading. Let's set it up:
- `npm install --save-dev webpack-dev-server`
- `"start": "webpack serve"`

Quiz



Quiz

- <https://create.kahoot.it/share/javascript-intermediate/61bb5352-538b-49af-adaa-e9213dd0b613>
- <https://create.kahoot.it/share/asynchronous-javascript/e9d58fd1-8106-4376-b2a4-fb0dfa4fb99b>
- <https://codingfantasy.com/games/flexboxadventure>

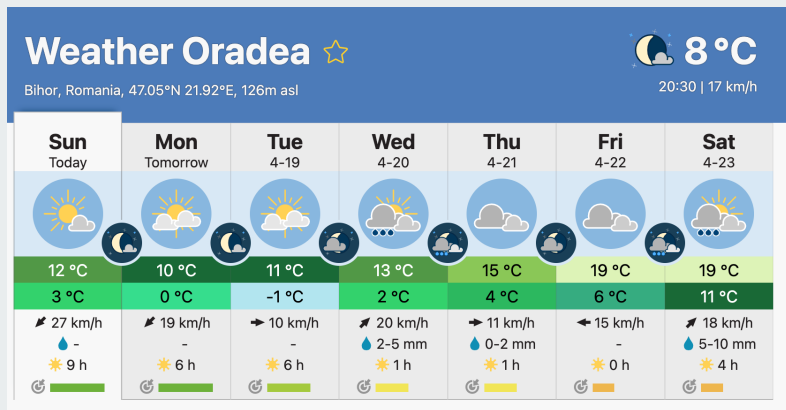


Interview Questions

- <https://www.interviewbit.com/javascript-interview-questions/#basic-js-interview-questions>

Assignment

Assignment



- Using the following API
[https://weatherdbi.herokuapp.com/data/w
eather/oradea](https://weatherdbi.herokuapp.com/data/weather/oradea)
- Create an interface to display the weather forecast in Celsius
- The interface should have the possibility to switch to Fahrenheit
- Show the min/max temperatures, the picture for the situation and the day name of course
- API Documentation here:
[https://weatherdbi.herokuapp.com/docum
entation/v1](https://weatherdbi.herokuapp.com/documentation/v1)
- You should use fetch (promises), display flex and anything you learned and want to apply



Assignment (Girls)

- Using the following API <https://makeup-api.herokuapp.com/api/v1/products.json?brand=maybelline>
- Create an interface to display the Makeup Products for 3 Brands of your liking (e.g. Maybelline and two others)
- On item click redirect to buy page
- Display item price
- Display items in a grid(as a layout) of 5 in a row
- You should use fetch (promises), display flex and anything you learned and want to apply
- Documentation API here: <https://makeup-api.herokuapp.com/>



Assignment (Boys)

- Using the following API
<https://vpic.nhtsa.dot.gov/api/>
- Create an interface to display 3 brands of your favorite USA made Car
- Each brand should have 3 models displayed in their respective category
- on car click you should redirect to a google search wiki with the car brand and model
- Documentation API here:
<https://vpic.nhtsa.dot.gov/api/>