**ES6+ Env Setup ( webpack & babel )**

* For better developer experience
* Cross browser support and compatibility
* Fast & professional code experience
* Use modern syntax

**Webpack**

* Webpack is a module bundler
* It builds your apps dependency graph
* Create a modules bundle
* Bundle your assets, scripts, images , styles

**Babel**

* Babel is a JavaScript Compiler
* Use next generation JS today
* Don't worry, babel will convert it to a vanilla Js code for all browser support.
* Babel is also called transpiler

**Setup Environment**

* First create a project ( Folder ) for your apps
* Initialize npm through it   
  npm init
* Or if you skip npm init setup   
  npm init -y
* Now install webpack and babel ( old babel )  
  npm install babel-core --save-dev / -D  
  npm install babel-preset-env --save-dev / -D   
    
  New version of babel - use it not old   
  npm install @babel/core --save-dev / -D   
  npm install @babel/preset-env --save-dev / -D   
    
    
  npm install webpack --save-dev / -D  
  npm install webpack-cli --save-dev / -D   
  npm install webpack-dev-server --save-dev / -D
* Now create a entry js file for your apps in src folder   
  app.js / index.js

**loaders for environment**

* sass
* style-loader
* css-loader
* sass-loader
* babel-loader

**Webpack configuration**

* Create a webpack.config.js file in root patch
* Now set configuration in this file   
    
  const path = require(‘path’);  
    
  Module.exports = {   
   mode : ‘development’,  
   entry : path.resolve(\_\_dirname, ‘src/app.js’) ,  
   output : {  
   path : path.resolve(\_\_dirname, ‘dist’),  
   filename : ‘app.bundle.js’,  
   clean : true,  
   assetModuleFielname: ‘[name][ext]’  
   },  
   module : {   
   rules : [  
   {  
   test : /\.js$/,  
   exclude : /node\_modules/,  
   use : {  
   loader : ‘babel-loader’,  
   options : {  
   Presets : [‘@babel/preset-env’]  
   }  
   }  
   },  
   {  
   test : /\.scss$/,  
   use : [‘style-loader’, ‘css-loader’, ‘sass-loader’]  
   },  
   {  
   test : /\.(png|jpg|jpeg|gif|webp)$/,  
   type : ‘asset/resource’  
   },  
   ]   
   },  
   devServer : {  
   static : {  
   directory : path.resolve(\_\_dirname, ‘dist’);  
   },  
   port : 3000,   
   open : true,   
   hot : true,   
   compress : true,   
   historyApiFallback : true   
   }  
    
  }

**Build Scripts**

* Open package.json file
* Type a new scripts for webpack into scripts   
  “build” : “webpack”  
  “start” : “webpack server”

**Run Environment**

* Now just hit for build app  
  npm run build
* And run app   
  npm run start

**Plugins for HTML Template**

* Install **html webpack plugin** for templating   
  npm i -D html-webpack-plugin
* now set this plugins in webpack config  
  const HtmlWebPack = require(‘html-webpack-plugin’);  
    
   plugins : [  
   new HtmlWebPack({  
   title : ‘Website Title’,  
   filename : ‘index.html’,  
   template : ‘public/index.html’  
   })  
   ],