Webpack:

npm i -D webpack webpack-cli

{

  "name": "webpack",

  "version": "1.0.0",

  "description": "",

  "main": "index.js",

  "scripts": {

    "build": "webpack production"

  },

  "keywords": [],

  "author": "",

  "license": "ISC",

  "devDependencies": {

    "webpack": "^5.90.0",

    "webpack-cli": "^5.1.4"

  }

}

Version 1:

const path = require("path");

module.exports = {

  mode: "development",

  entry: path.resolve(\_\_dirname, "src/index.js")

  output: {

    path: path.resolve(\_\_dirname, "dist"),

    filename: "bundle.js",

  },

};

Version 2

const path = require("path");

module.exports = {

  mode: "development",

  entry: { bundle: path.resolve(\_\_dirname, "src/index.js") },

  output: {

    path: path.resolve(\_\_dirname, "dist"),

    filename: "[name].js",

  },

};

For loading sass

npm i -D sass style-loader css-loader sass-loader

plugins

npm I -D html-webpack-plugin

with this plugin we were able to generate the html file

index.html with a reference to the bundle script with defer attribute

What is the problem is if we change anything in the index.html and then if we run npm run build the old index.html which we edited is replaced by the index.html generated by the above plugin

So for this purpose

We use a template in the src folder and give the reference in webpack config file as below

  plugins: [

    new HtmlWebpackPlugin({

      title: "Webpack App",

      filename: "index.html",

      template: "src/template.html",

    }),

  ],

Also as next step for caching the tempalate

We did a small change to the file name , we included the [contenthash]

  output: {

    path: path.resolve(\_\_dirname, "dist"),

    filename: "[name][contenthash].js",

  },

Webpack dev server for auto reload

    "build": "webpack",

    "dev": "webpack serve"

  },

In package.json

* We do npm run dev
* It will ask if we need to install webpack-dev-server
* After clicking yes it will install the webpack-dev-server
* App opens at <http://localhost:8080/>

Customize the webpack-dev-server with the below devServer config in webpack.config.js

  output: {

    path: path.resolve(\_\_dirname, "dist"),

    filename: "[name][contenthash].js",

  },

  devServer: {

    static: {

      directory: path.resolve(\_\_dirname, "dist"),

    },

    port: 3000,

    open: true,

    hot: true,

    compress: true,

    historyApiFallback: true,

  },

Hot->hot reloading

Compress->compress

Open🡪open automatically

historyApiFallback- > ?

Now if you do any change in the index.js file in the src and then do npm run build

Another build file gets generated

So we add clean:true

  output: {

    path: path.resolve(\_\_dirname, "dist"),

    filename: "[name][contenthash].js",

    clean:true

  },

In the output in the webpack.config.js

Enable source maps

Good for debugging

Show line number where the problem is

  devtool: "source-map",

  devServer: {

    static: {

      directory: path.resolve(\_\_dirname, "dist"),

    },

This source-map will generate a bundle.map file and this can be used for debugging in the browser chrome dev tools 🡪sources

For backward compatible for old browsers

Use babel-loader

--install as npm I -D babel-loader @babel/core @babel/preset-env

    {

        test: /\.js$/,

        exclude: /node\_modules/,

        use: {

          loader: "babel-loader",

          options: {

            presets: ["@babel/preset-env"],

          },

        },

      },

To load the images:

 {

        test: /\.(png|svg|jpg|jpeg|gif)$/i,

        type: "asset/resource",

      },

 output: {

    path: path.resolve(\_\_dirname, "dist"),

    filename: "[name][contenthash].js",

    clean: true,

    assetModuleFilename: "[name][ext]",

  },