

WEBPACK

Webpack es una herramienta open source que utilizan los desarrolladores para empaquetar y exportar todos los ficheros necesarios para que un proyecto funcione con las dependencias frontend

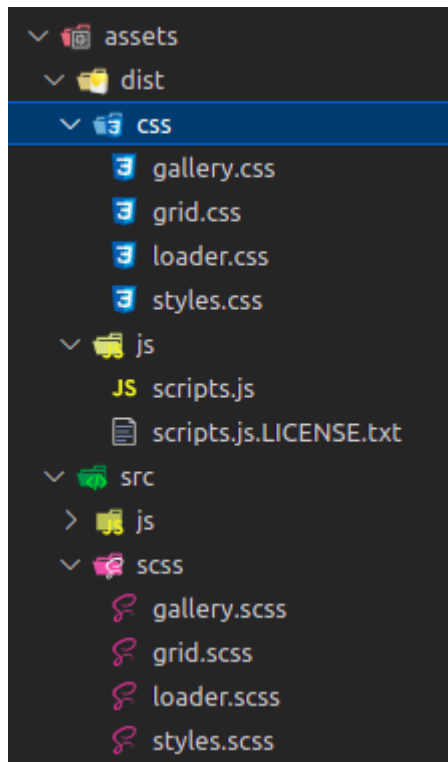
Para empezar a usar Webpack, lo tenemos que hacer en un entorno npm por lo que tenemos que poner en terminal en la raíz de nuestro proyecto el comando **npm init**, el cual nos creará un archivo package.json

Una vez generado el package.json procedemos a instalar las dependencias utilizando el siguiente comando **npm install bootstrap @popperjs/core autoprefixer clean-webpack-plugin node-sass nodemon postcss-cli webpack webpack-cli webpack-merge --save-dev**. El comando **--save-dev** sirve para garantizar que el paquete se utilice para fines de desarrollo y se use el package.json. También se nos crea un archivo package.lock.json

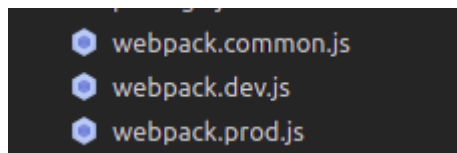
```
{
  "name": "4.2componentes",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "build": "npm run css && webpack --config webpack.prod.js",
    "css-compile": "node-sass --include-path node_modules --output-style",
    "css-prefix": "postcss --replace assets/dist/css/styles.css --use a",
    "css": "npm run css-compile && npm run css-prefix",
    "watch": "nodemon -e scss -x \"npm run css && webpack --config webp",
  },
  "author": "",
  "license": "ISC",
  "devDependencies": {
    "@popperjs/core": "^2.11.2",
    "autoprefixer": "^10.4.2",
    "bootstrap": "^5.1.3",
    "clean-webpack-plugin": "^4.0.0",
    "node-sass": "^7.0.1",
    "nodemon": "^2.0.15",
    "postcss-cli": "^9.1.0",
    "webpack": "^5.69.1",
    "webpack-cli": "^4.9.2",
    "webpack-merge": "^5.8.0"
  },
  "dependencies": {
    "generator-sassdoc-theme": "^0.2.3",
    "sassdoc": "^2.7.3"
  }
}
```

El comando usado anteriormente, genera en el package.json las dependencias que podemos observar, en el apartado de devDependencies.

Una vez hecho esto procedemos a crear la estructura del proyecto, que sería tal que así.



Después de esto hay que crear tres ficheros.js para poder usar el webpack, con su correspondiente configuración.



- **webpack.common.js**

```
const path = require('path');
const { CleanWebpackPlugin } = require('clean-webpack-plugin');
module.exports = {
  entry: {
    scripts: './assets/src/js/scripts.js'
  },
  plugins: [
    new CleanWebpackPlugin(),
  ],
  output: {
    path: path.resolve(__dirname, 'assets/dist/js/'),
    filename: '[name].js',
  }
};
```

- **webpack.dev.js**

```
const { merge } = require('webpack-merge');
const common = require('./webpack.common.js');
module.exports = merge(common, {
  mode: 'development',
  watch: true,
  watchOptions: {
    ignored: /node_modules/
  }
});
```

- webpack.prod.js

```
const { merge } = require('webpack-merge');
const common = require('./webpack.common.js');
module.exports = merge(common, {
  mode: 'production',
});
```

Una vez tenemos listo nuestro entorno de trabajo, ya podemos usar el comando **npm run build** para que nuestro proyecto empiece a funcionar. Esto completará nuestro archivos scss y los convertirá en archivos css.

Para poder usar los scripts que hemos configurado en la terminal tenemos que poner el comando **npm run nombre_del_script**.