

DINT  
Bootstrap  
- webpack

Curso: 2023-2024

## Índice

Bootstrap and Webpack .....	3
Instalación de Bootstrap con webpack clonando el repositorio .....	3
Estructura de un proyecto con Bootstrap y webpack.....	4
Archivo package.json .....	5
Fichero webpack.config.js.....	5
Verificar que SASS funciona .....	7

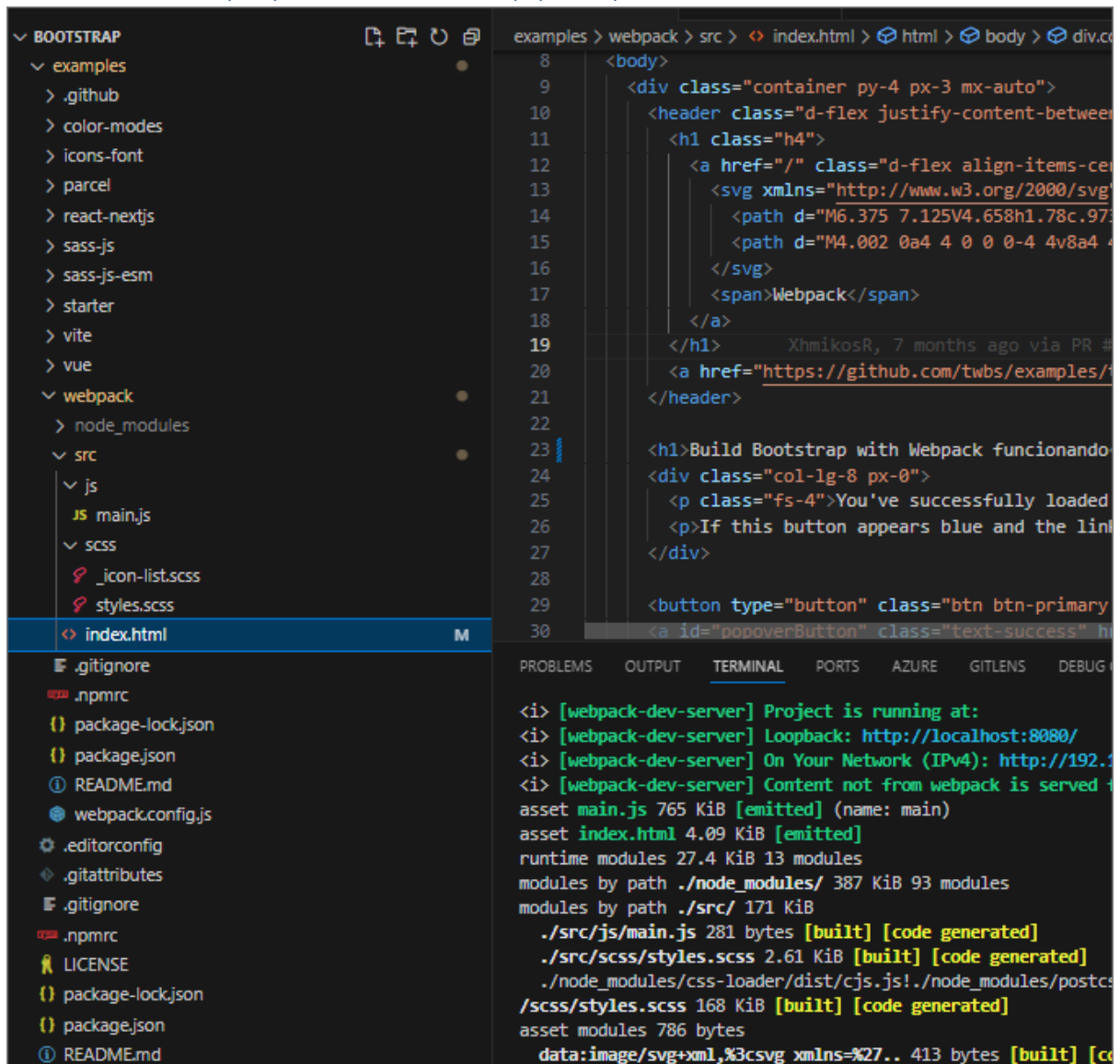
## Bootstrap and Webpack



<https://getbootstrap.com/docs/5.3/getting-started/webpack/>

Instalación de Bootstrap con webpack clonando el repositorio

## Estructura de un proyecto con Bootstrap y webpack



## Archivo package.json

```
{
  "name": "webpack",
  "description": "Include Bootstrap's source Sass and individual JavaScript
plugins with Webpack.",
  "version": "0.0.0",
  "private": true,
  "repository": "twbs/examples",
  "license": "MIT",
  "stackblitz": {
    "startCommand": "npm start"
  },
  "scripts": {
    "start": "webpack serve",
    "build": "webpack build --mode=production",
    "test": "npm run build"
  },
  "dependencies": {
    "@popperjs/core": "^2.11.8",
    "bootstrap": "^5.3.0"
  },
  "devDependencies": {
    "autoprefixer": "^10.4.14",
    "css-loader": "^6.8.1",
    "html-webpack-plugin": "^5.5.3",
    "postcss-loader": "^7.3.3",
    "sass": "^1.64.1",
    "sass-loader": "^13.3.2",
    "style-loader": "^3.3.3",
    "webpack": "^5.88.2",
    "webpack-cli": "^5.1.4",
    "webpack-dev-server": "^4.15.1"
  }
}
```

## Fichero webpack.config.js

```
'use strict'

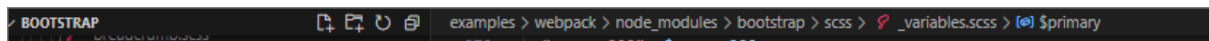
const path = require('path')
const autoprefixer = require('autoprefixer')
const HtmlWebpackPlugin = require('html-webpack-plugin')

module.exports = {
  mode: 'development',
  entry: './src/js/main.js',
  output: {
```

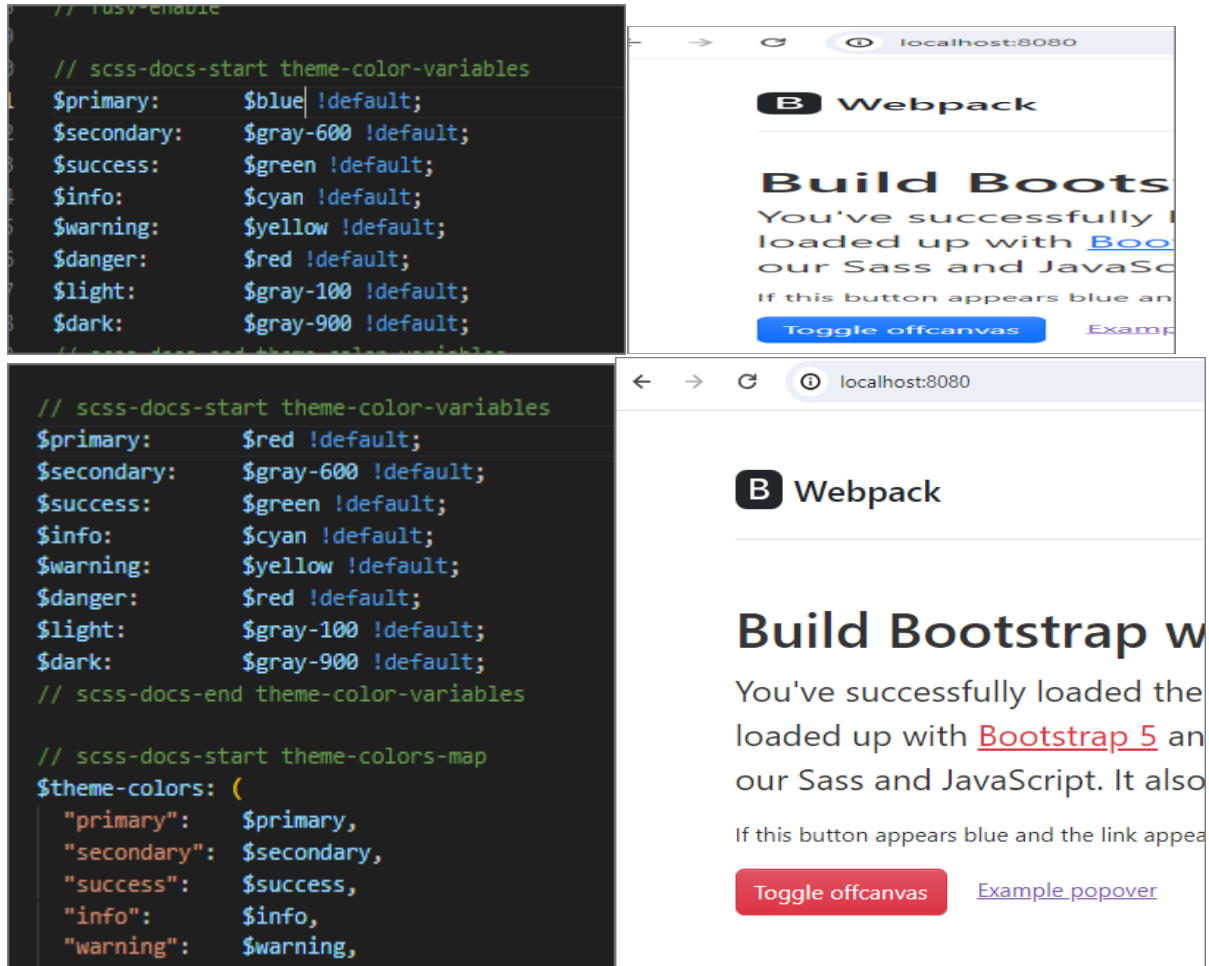
```
    filename: 'main.js',
    path: path.resolve(__dirname, 'dist'),
  },
  devServer: {
    static: path.resolve(__dirname, 'dist'),
    port: 8080,
    hot: true
  },
  plugins: [
    new HtmlWebpackPlugin({ template: './src/index.html' })
  ],
  module: {
    rules: [
      {
        test: /\.scss$/,
        use: [
          {
            // Adds CSS to the DOM by injecting a <style> tag
            loader: 'style-loader'
          },
          {
            // Interprets `@import` and `url()` like `import/require()` and
            will resolve them
            loader: 'css-loader'
          },
          {
            // Loader for webpack to process CSS with PostCSS
            loader: 'postcss-loader',
            options: {
              postcssOptions: {
                plugins: [
                  autoprefixer
                ]
              }
            }
          },
          {
            // Loads a SASS/SCSS file and compiles it to CSS
            loader: 'sass-loader'
          }
        ]
      }
    ]
  }
}
```

## Verificar que SASS funciona

Vamos al directorio siguiente que es donde esta instalado scss y modificamos por ejemplo la variable \$primary



Vemos que inicialmente el botón es azul por que en



Automaticamente cambia el color en la web