

Tooling

npm



Gestor de dependencias de ~~Node.js~~ Javascript

- Compartir
- Reutilizar
- Instalación automática
- Gestión de versiones
- Enlazado con Git

Paquetes npm

`package.json`

Sirve:

- Documentación de las dependencias de tu proyecto
- Para especificar la versión de código de tu proyecto
- Indicar repositorio
- Unificación de desarrollo y entregables
- Scripts de proyecto

Eslint

Eslint

- Un *analizador sintáctico estático*
- Revisa el código de nuestro proyecto en cada build
- Nos avisa si no seguimos el conjunto de **reglas** que tenemos activas

Eslint

- Entiende la sintaxis, pero no la semántica
- Que toda la base de código siga un **estilo homogéneo**
- Puede detectar errores simples
- NO es ninguna garantía de funcionamiento

Eslint

```
npm install eslint
```


Eslint

```
./node_modules/.bin/eslint --init
```

Eslint

```
./node_modules/.bin/eslint .
```

Eslint

```
console.log('hola')
```

Eslint

```
./node_modules/.bin/eslint test.js
```

Eslint

```
/home/redradix/projects/jspro/test.js  
1:1   warning  Unexpected console statement  no-console  
1:20  error      Missing semicolon            semi
```

✖ 2 problems (1 error, 1 warning)

Eslint

- Nuestro código **no cumple las reglas** de estilo
- Tenemos dos opciones:
 - modificar el código
 - modificar las reglas

Eslint

```
/home/redradix/projects/jspro/test.js
```

```
1:1    warning  Unexpected console statement  no-console  
1:20   error     Missing semicolon            semi
```

✖ 2 problems (1 error, 1 warning)

Eslint

/home/redradix/projects/jspro/test.js

1:1	warning	Unexpected console statement	no-console
1:20	error	Missing semicolon	semi

✖ 2 problems (1 error, 1 warning)

Eslint

```
/home/redradix/projects/jspro/test.js
```

1:1	warning	Unexpected console statement	no-console
1:20	error	Missing semicolon	semi

✖ 2 problems (1 error, 1 warning)

Eslint

```
/home/redradix/projects/jspro/test.js
```

```
1:1    warning
```

```
Unexpected console statement
```

```
no-console
```

```
1:20   error
```

```
Missing semicolon
```

```
semi
```

✖ 2 problems (1 error, 1 warning)

Eslint

```
/home/redradix/projects/jspro/test.js
```

```
1:1    warning  Unexpected console statement
```

```
1:20   error     Missing semicolon
```

```
no-console  
semi
```

✖ 2 problems (1 error, 1 warning)

Eslint

```
console.log('hola');
```

Eslint

```
/home/redradix/projects/jspro/test.js
```

```
1:1  warning  Unexpected console statement  no-console
```

✖ 1 problems (1 warning)

Eslint

```
/* eslint-disable */  
console.log('hola');
```

Eslint

```
/* eslint-disable */  
console.log('hola');  
/* eslint-enable */
```

Eslint

```
// eslint-disable-next-line  
console.log('hola');
```


Eslint

```
/* eslint-disable no-console */  
console.log('hola');
```

Eslint

- Eslint sigue las reglas definidas en el fichero `.eslintrc`
- Podemos modificar el fichero para introducir modificaciones permanentes

Eslint

```
{  
  "extends": "airbnb-base",  
  "plugins": [  
    "import"  
  ]  
}
```

Eslint

```
{  
  "extends": "airbnb-base",  
  "plugins": [  
    "import"  
  ]  
}
```

Eslint

```
{  
  "extends": "airbnb-base",  
  "plugins": [  
    "import"  
  ],  
  "rules": {  
    "no-console": "off"  
  }  
}
```

Eslint

```
{  
  "extends": "airbnb-base",  
  "plugins": [  
    "import"  
  ],  
  "rules": {  
    "no-console": "off"  
  }  
}
```

Eslint

`http://eslint.org/docs/rules/`

Eslint

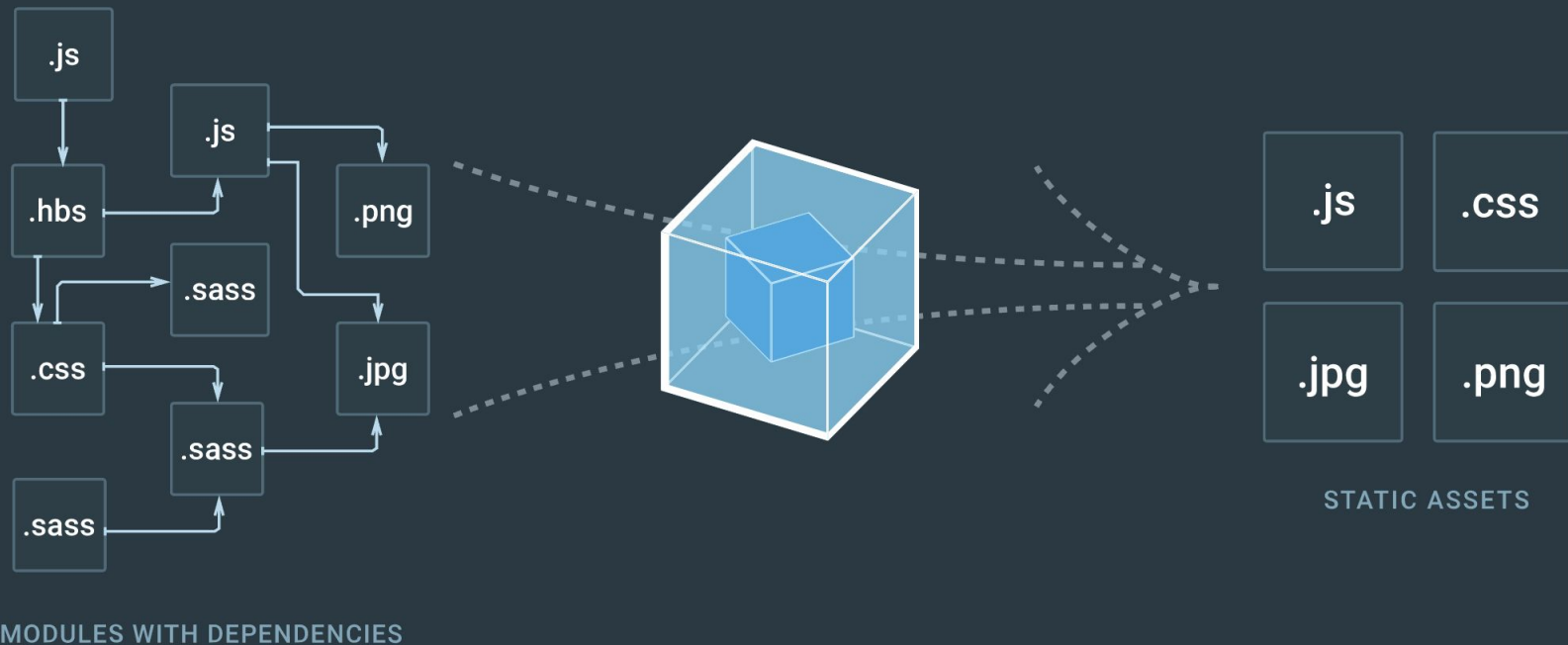
¡Intégralo con tu **editor**!

Webpack

Webpack

- Un empaquetador de módulos
- Analiza el árbol de código y sus dependencias
 - código
 - assets (estilos, imágenes, json, ...)
- “Empaqueta” lo que puede

Webpack



Webpack

- Conceptos fundamentales
 - Entry
 - Output
 - Loaders (test && use)
 - Plugins

Webpack

- Loaders para muchos formatos
 - Sass / stylus / less / ...
 - JSON
 - glsl
 - txt

Webpack

- Excelente soporte para código
 - Concatena, minifica, comprime...
 - Dead code elimination
 - Múltiples puntos de entrada

Webpack

```
npm install webpack
```


Webpack

```
module.exports = {  
  entry: path.join(__dirname, 'src', 'main.js'),  
  output: {  
    path: path.join(__dirname, 'dist'),  
    filename: 'bundle.js',  
    publicPath: '/'  
  },  
  module: {rules: [{ test: /\.txt$/, use: 'raw-loader' }]},  
  plugins: [  
    new webpack.optimize.UglifyJsPlugin(),  
    new HtmlWebpackPlugin({template: './src/index.html'})  
  ]  
}
```

```
rules: [  
  { test: /\.js$/,  
    exclude: [/node_modules/],  
    use: 'babel-loader',  
    query: {  
      presets: ['es2015']  
    },  
    include: [path.join(__dirname, 'src')]  
  },  
  { test: /\.json$/,  
    exclude: [/node_modules/],  
    use: 'json-loader'  
  }  
]
```

```
rules: [  
  { test: /\.js$/,  
    exclude: [/node_modules/],  
    use: 'babel-loader',  
    query: {  
      presets: ['es2015']  
    },  
    include: [path.join(__dirname, 'src')]  
  },  
  { test: /\.json$/,  
    exclude: [/node_modules/],  
    use: 'json-loader'  
  }  
]
```

```
rules: [  
  { test: /\.js$/,  
    exclude: [/node_modules/],  
    use: 'babel-loader',  
    query: {  
      presets: ['es2015']  
    },  
    include: [path.join(__dirname, 'src')]  
  },  
  { test: /\.json$/,  
    exclude: [/node_modules/],  
    use: 'json-loader'  
  }  
]
```

```
rules: [  
  { test: /\.js$/,  
    exclude: [/node_modules/],  
    use: 'babel-loader',  
    query: {  
      presets: ['es2015']  
    },  
    include: [path.join(__dirname, 'src')]  
  },  
  { test: /\.json$/,  
    exclude: [/node_modules/],  
    use: 'json-loader'  
  }  
]
```

```
rules: [  
  { test: /\.js$/,  
    exclude: [/node_modules/],  
    use: 'babel-loader',  
    query: {  
      presets: ['es2015']  
    },  
    include: [path.join(__dirname, 'src')]  
  },  
  { test: /\.json$/,  
    exclude: [/node_modules/],  
    use: 'json-loader'  
  }  
]
```

Webpack

webpack

Webpack

webpack -w

Todo junto

Vamos a crear nuestro primer paquete npm

Usaremos

- npm
- Webpack
- Eslint
- Mocha

Todo junto

- npm
- npm init
- Configuramos package.json
- Empezamos a instalar dependencias
- Añadimos fichero de configuración webpack
- Añadimos dependencias (webpack, loaders...)
- Setear linter