

FRONTEND DEVELOPMENT

WEBPACK 4

WEBPACK 4

- ▶ Narzędzie do „pakowania” plików różnego typu (skrypty, arkusze stylów, obrazki) do jednego lub kilku pliku wynikowego .js
- ▶ Zachowuje kolejność ładowania łączonych plików
- ▶ Pozwala na zminifikowanie oraz zoptymalizowanie plików
- ▶ Pozwala na dzielenie pliku wynikowego na mniejsze kawałki
- ▶ Zwiększenie wydajności przez mniejszą liczbę plików do pobrania

INSTALACJA WEBPACK 4

- ▶ `npm install --save-dev webpack`
- ▶ `npm install --save-dev webpack@<version>`
- ▶ `npm install --save-dev webpack-cli`
- ▶ `"scripts": {`
- ▶ `„start”: "webpack --config webpack.config.js"`
- ▶ `}`

WEBPACK CONFIG

- ▶ Tworzymy plik o nazwie webpack.config.js w głównym katalogu projektu

```
const path = require('path');
module.exports = {
  entry: './src/home.js',
  output: {
    path: path.resolve(__dirname, './dist'),
    filename: 'bundle.js'
  }
};
```

WIELE PLIKÓW WEJŚCIOWYCH

```
const path = require('path');
module.exports = {
  entry: {
    home: './src/home.js',
    post: './src/post.js',
    contact: './src/contact.js'
  },
  output: {
    path: path.resolve(__dirname, './dist'),
    filename: '[name].bundle.js'
  }
};
```

LOADERY

- ▶ Biblioteki dokonujące transformacji na kodzie źródłowym
- ▶ Biblioteki te pozwalają na wczesne przetwarzanie importowanych plików. Dzięki temu możemy w trakcie kompilacji dokonywać kompresji, translacji języków itp.
- ▶ Pozwalają na „pakowanie” plików innych niż JavaScript, np. TypeScript to JavaScript, CSS (LESS/SCSS), pliki JSON, obrazki

INSTALACJA I SPOSOBY UŻYCIA

- ▶ `npm install --save-dev <nazwa_loader>`
- ▶ `require('css-loader!./styles.css');`
- ▶ `webpack --module-bind 'css=css-loader'`

PLIKI KONFIGURACYJNE

```
const path = require('path');
module.exports = {
  entry: {
    home: './src/home.js',
    post: './src/post.js',
    contact: './src/contact.js'
  },
  output: {
    path: path.resolve(__dirname, './dist'),
    filename: '[name].bundle.js'
  },
  module: {
    rules: [
      {
        test: /\.js$/,
        exclude: /node_modules/,
        use: {
          loader: 'babel-loader',
          options: {
            presets: ['env']
          }
        }
      },
      {
        test: /\.css$/,
        use: [ 'style-loader', 'css-loader' ]
      }
    ]
  }
};
```

OBRAZKI W PLIKACH KONFIGURACYJNYCH

```
module: {
  rules: [
    {
      test: /\.?(png|jpg|gif)$/,
      use: [
        {
          loader: 'url-loader',
          options: {
            limit: 8192
          }
        }
      ]
    }
  ]
}
```

PLUGINS

- ▶ Wbudowany lub zewnętrzny kod dodający dodatkowe funkcjonalności
- ▶ W przypadku narzędzia Webpack, jesteśmy w stanie dołożyć funkcjonalności podczas „pakowania” skryptów
- ▶ Wtyczki są dopełnieniem „loaderów”

KONFIGURACJA

```
const HtmlWebpackPlugin = require('html-webpack-plugin'); //installed via npm
const webpack = require('webpack'); //to access built-in plugins
const path = require('path');
const config = {
  entry: './path/to/my/entry/file.js',
  output: {
    filename: 'my-first-webpack.bundle.js',
    path: path.resolve(__dirname, 'dist')
  },
  module: {
    rules: [
      {
        test: /\.js$/,
        use: 'babel-loader'
      }
    ]
  },
  plugins: [
    new webpack.optimize.UglifyJsPlugin(),
    new HtmlWebpackPlugin({template: './src/index.html'})
  ]
};
module.exports = config;
```

COMMONSCHUNKPLUGIN

- ▶ Wtyczka służąca do eksportu części wspólnej z plików wejściowych do osobnego pliku wyjściowego. W wersji 4 zastąpione przez SplitChunkPlugin

```
module.exports = {  
  // ...  
  plugins: [  
    new webpack.optimize.CommonsChunkPlugin({  
      name: 'vendor',  
      filename: 'vendor.bundle.js'  
    })  
  ]  
};
```

HTMLWEBPACKPLUGIN

- ▶ Wtyczka generująca plik html, który zawiera wszystkie wygenerowane pliki

```
const HtmlWebpackPlugin = require('html-webpack-plugin')
module.exports = {
  entry: 'index.js',
  output: {
    path: __dirname + '/dist',
    filename: 'index_bundle.js'
  },
  plugins: [
    new HtmlWebpackPlugin()
  ]
}
```

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Webpack App</title>
  </head>
  <body>
    <script src="index_bundle.js"></script>
  </body>
</html>
```

ŹRÓDŁA

- ▶ <https://www.webpack.js.org/concepts/>
- ▶ <https://www.nafrontendzie.pl/>
- ▶ <https://webpack.js.org/loaders/>
- ▶ <https://webpack.js.org/plugins/>
- ▶ <https://github.com/webpack-contrib/awesome-webpack#webpack-plugins>

