

Webpack



- [webpack.config.js](#)
- [entry](#)
- [output](#)
- [externals](#)
- [module](#)
- [resolve](#)
- [plugins](#)
- [require\(\)](#)
- [require.ensure\(\)](#)
- [Loader](#)
- [Plugin](#)
- [Node API](#)
- [webpack](#)

Webpack  bundler


Webpack 

-  chunk
- 



Webpack 

```
$ webpack ./src/App.js ./build/bundle.js
```

App.js

bundle.js

App.js

App.js

```
let $ = require('jquery');  
$("p").css({ color: 'red'});
```

bundle.js

App.js

jquery

--watch -w

```
$ webpack -w js/profile.js dist/bundle.js  
$ webpack --watch
```

--devtool

Source map

```
$ webpack -w --devtool source-map js/profile.js dist/bundle.js
```

--config

```
$ webpack --config ./path/to/webpack.dev.js
```

webpack.config.js

Webpack

webpack.config.js

CommonJS

```
module.exports = {  
  entry: '...',  
  output: '...',  
  // ...  
};
```

Webpack

entry

entry

```
entry: 'app.js',

// 
entry: [
  'react-hot-loader/patch',
  'webpack-dev-server/client?http://localhost:8080',
  'webpack/hot/only-dev-server',
  './src/client.jsx'
],
```

entry

output

```
module.exports = {
  entry: {
    desktop: './src/desktop.js',
    mobile: './src/mobile.js'
  },
  output: {
    path: './dist',
    filename: '[name].bundle.js',
    // bundle.js 
    chunkFilename: '[id].common.js'
  }
};
```

output

output

```
output: {
  // 
  path: './dist',
  // 
  filename: 'bundle.js',
  // 
```

```
// 配置公共路径
publicPath: '/',
// 配置目标库
libraryTarget: "var",
// 配置库名称
library: "Foo"
},
```

Webpack 配置

47a0cc1b840cb310842cb85fb5b6116c.js

```
module.exports = {
  entry: './src/main.js',
  output: {
    path: './dist',
    filename: '[hash].js'
  }
};
```

externals

externals

```
// 配置 jquery 为外部依赖
externals: {
  "jquery": "jQuery"
},
```

module

Babel

module.rules

```
module: {
  loaders: [
    { test: /\.css$/, loader: 'style!css' }
  ]
},
```

resolve

resolve

1 resolve.modules

resolve.modules

```
resolve: {
  modules: ['node_modules', 'src']
}
```

src/dir1/foo.js

```
// src/dir1/foo.js
import Bar from '../dir2/bar';
```

src/dir2/bar.js resolve.modules

```
import Bar from 'dir2/bar';
```

2 resolve.extensions

resolve.extensions Webpack

```
resolve: {
  extensions: ['.js', '.jsx']
},
```

3 resolve.mainFields

package.json main
Node

package.json

- browser
- module ES6 CommonJS
- main index.js

d3 package.json

```
{
  ...
  main: 'build/d3.Node.js',
  browser: 'build/d3.js',
  module: 'index',
  ...
}
```

resolve.mainFields Webpack

☐ Webpack target webworker ☐ web
 resolve.mainFields

```
mainFields: ["browser", "module", "main"]
```

Webpack package.json browser
 module module main
 target node ☐ resolve.mainFields

```
mainFields: ["module", "main"]
```

plugins ☐

plugins

require()

require

```
require(['animals'], function(Animal) {
  var bear = new Animal('bear');
});
```

require.ensure()

[illegible]

require.ensure

```
require('./a');

if (condition) {
  require.ensure([], function(require){
    require('./b');
    console.log('done!');
  });
}
```

☐ ☐ ☐ ☐ ☐ `a.js` ☐ ☐ `b.js` ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ `bundle.js` ☐ `0.bundle.js` ☐ ☐ ☐ ☐ ☐ ☐

`bundle.js` ☐ ☒ ☐ ☐ `bundel.js` ☐ ☐ `condition` ☐ ☐ ☐ ☐ `0.bundle.js` ☐

`require.ensure`

0.bundle.js

0.bundle.js require

[illegible]

require.ensure

```
require.ensure([], function(require){
  require('./a');
}, 'foo');

require.ensure(['foo'], function(require){
  require('./b');
});
```

`require.ensure`

```
require.ensure
```

```
require.ensure(['./b.js'], function(require) {
  require('./c.js');
});
```

0.bundle.js
c.js b.js b.js

Loader

Webpack Loader JavaScript

```
var css = require('css!./css/style.css');  
//  
var css = require('css-loader!./css/style.css');
```

css-loader CSS

Loader

```
require('style!css!./css/style.css');
```

style-loader CSS link

loader

loader

```
module.exports = {  
  module: {  
    loaders: [  
      { test: /\.css$/, loader: 'style!css' }  
    ]  
  }  
};
```

```
{ test: /\.js$/,  
  exclude: /(node_modules|bower_components)/,  
  loader: 'babel-loader',  
  query: {  
    presets: ['es2015']  
  }  
},
```

■■■■■■■

.css ■■■■■■

```
require('./css/style.css');
```

■■■■

preloader

```
preLoaders: [
  { test: /\.js?$/, exclude: /node_modules/, loader: 'eslint-loader', include: __dirname +
    '/' }
],
```

Plugin

■■■■■■■■■

■■■■

```
plugins: [
  new webpack.optimize.UglifyJsPlugin({
    compress: {
      warnings: false
    }
  })
]
```

■■■■

```
module.exports = {
  . . . . .,
  module: {
    loaders: [
      // for less files
      { test: /\.less$/,
        exclude: /(node_modules|bower_components)/,
        loader: ExtractTextPlugin.extract("style-loader", "css-loader", "less-loader")
      }
    ]
  },
}
```

```

plugins: [
  new ExtractTextPlugin("style.css")
]
}

```

Node API

Webpack  Node 

Webpack

```

var webpack = require('webpack');
var WebpackDevServer = require('webpack-dev-server');
var config = require('./webpack.config');

var server = new WebpackDevServer(webpack(config), {
  contentBase: './dev',
  publicPath: config.output.publicPath,
  hot: true
});

server.listen(3000, 'localhost', function (err, result) {
  if (err) {
    console.log(err);
  }
  console.log('Listening at localhost:3000');
});

```

```

// require the webpack Node.js library
var webpack = require('webpack');

webpack({
  // The first argument is your webpack config
  entry: './src/entry.js',
  output: {
    path: './dist',
    filename: 'bundle.js'
  }
}, function(err, stats) {

```

```
// The second argument is a callback function that returns
// more information about your bundle when it is complete
});
```

□□□□□

```
var webpack = require('webpack');

// Create an instance of the compiler
var compiler = webpack({ /* webpack config */ });

// Run the compiler manually
compiler.run(function(err, stats) { });
```

watch □□□□□□□□□□□□□□□□

```
compiler.watch(/* watchDelay */ 200, function(err, stats) { });
```

□□□□

- [Module Bundling with Webpack: Getting Started Guide](#) by Chimeremeze Ukah

□□ #1
□ □□ □□ 2017-04-09 10:10:41 UTC
□ □□ □□ 2017-06-09 13:16:30 UTC