

```
export default (text = 'Hello world') => {  
  const element = document.createElement('div');  
  
  element.innerHTML = text;  
  
  return element;  
};
```

```
import component from './component';  
  
document.body.appendChild(component());
```

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

const PATHS = {
  app: path.join(__dirname, 'app'),
  build: path.join(__dirname, 'build'),
};

module.exports = {
  // Entries have to resolve to files! They rely on Node
  // convention by default so if a directory contains *index.js*,
  // it resolves to that.
  entry: {
    app: PATHS.app,
  },
  output: {
    path: PATHS.build,
    filename: '[name].js',
  },
  plugins: [
    new HtmlWebpackPlugin({
      title: 'Webpack demo',
    }),
  ],
};
```

```
module.exports = {  
  devServer: {  
    host: process.env.HOST, // Defaults to `localhost`  
    port: 80, // Defaults to 8080  
    // overlay: true captures only errors  
    overlay: {  
      errors: true,  
      warnings: true,  
    },  
  },  
},
```

```
module: {  
  rules: [  
    {  
      test: /\.css$/,  
      include,  
      exclude,  
  
      use: ['style-loader', 'css-loader'],  
    },  
  ],  
},
```

```
{  
  loader: 'css-loader',  
  
  options: {  
    modules: true,  
  },  
},
```



```
{
  // Conditions
  test: /\.js$/,
  include: PATHS.app,

  // Actions
  loader: 'babel-loader',
  options: {
    cacheDirectory: true,
    presets: ['react', 'es2015'],
  },
},
```

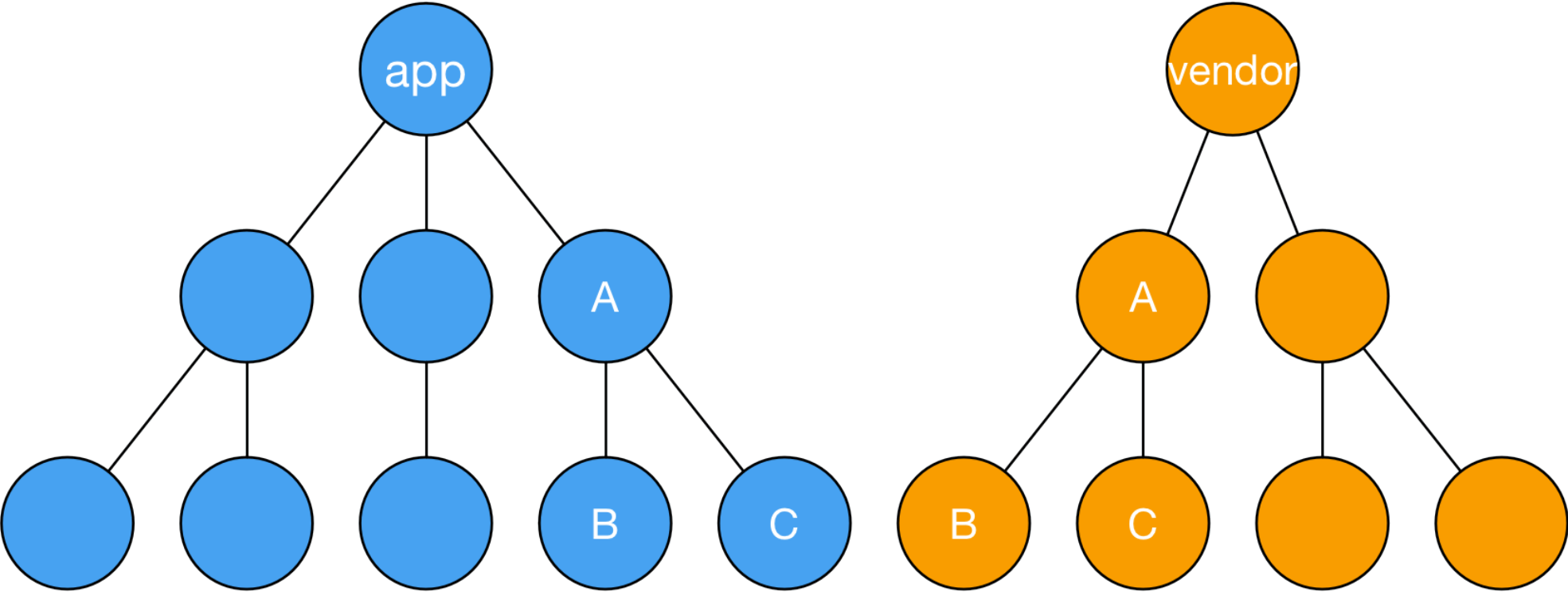
```
{  
  // Conditions  
  test: /\.js$/,  
  include: PATHS.app,  
  
  // Actions  
  use: {  
    loader: 'babel-loader',  
    options: {  
      cacheDirectory: true,  
      presets: ['react', 'es2015'],  
    },  
  },  
},  
},
```

```
{  
  // Conditions  
  test: /\.js$/,  
  include: PATHS.app,  
  
  // Actions  
  use: 'babel-loader?cacheDirectory,presets[ ]=es2015',  
},
```

```
{
  performance: {
    hints: 'warning', // 'error' or false are valid too
    maxEntrypointSize: 100000, // in bytes
    maxAssetSize: 450000, // in bytes
  },
},
```

devtool	build speed	rebuild speed	production supported	quality
eval	+++	+++	no	generated code
cheap-eval-source-map	+	++	no	transformed code (lines only)
cheap-source-map	+	o	yes	transformed code (lines only)
cheap-module-eval-source-map	o	++	no	original source (lines only)
cheap-module-source-map	o	-	yes	original source (lines only)
eval-source-map	—	+	no	original source
source-map	—	—	yes	original source

```
entry: {  
  app: PATHS.app,  
  vendor: ['react'],  
},
```



```
new webpack.optimize.CommonsChunkPlugin({  
  name: 'vendor',  
}),
```