

### 1、创建一个基于 webpack 模板的新项目

```
$ vue init webpack myvue
```

### 2、在当前目录下，安装依赖

```
$ cd myvue  
$ npm install
```

### 3、安装sass的依赖包

```
npm install --save-dev sass-loader  
//sass-loader依赖于node-sass  
npm install --save-dev node-sass
```

### 4、在build文件夹下的webpack.base.conf.js的rules里面添加配置

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

如下图所示：

```
module: {  
  rules: [  
    {  
      test: /\.vue$/,  
      loader: 'vue-loader',  
      options: vueLoaderConfig  
    },  
    {  
      test: /\.js$/,  
      loader: 'babel-loader',  
      include: [resolve('src'), resolve('test')]  
    },  
    {  
      test: /\.scss$/,  
      loaders: ['style', 'css', 'sass']  
    }  
  ]  
}
```

<http://blog.csdn.net/lily2016n>

5、在APP.vue中修改style标签

```
<style lang="scss">
```

6、然后运行项目

```
$ npm run dev
```

7、修改APP.vue的样式，可以看下效果

```
App.vue  package.json  webpack.base.conf.js
10     name: 'app'
11   }
12 </script>
13
14 <style lang="scss">
15   /* #app {
16     font-family: 'Avenir',
17     -webkit-font-smoothing:
18     -moz-osx-font-smoothing
19     text-align: center;
20     color: #2c3e50;
21     margin-top: 60px;
22   } */
23   $background:#cccccc;
24   body{
25     background:$background;
26   }
27 </style>
```

8、运行结果背景变成灰色吗，说明你已成功配置好sass



Welcome to Your Vue.js App

## Essential Links

[Core Docs](#)   [Forum](#)   [Gitter Chat](#)   [Twitter](#)  
[Docs for This Template](#)

## Ecosystem

[vue-router](#)   [vuex](#)   [vue-loader](#)   [awesome-vue](#)

<http://blog.csdn.net/lily2016n>

## jQ

```
cnpm install jquery --save-dev
```

找到build文件夹下的webpack.base.conf.js          加入var

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

在module.exports加入:

```
1. resolve: {
2.   extensions: ['.js', '.vue', '.json'],
3.   alias: {
4.     '@': resolve('src'),
5.     'assets': path.resolve(__dirname, '../src/assets'),
6.     'jquery': path.resolve(__dirname, '../node_modules/jquery/src/jquery')
7.   }
8. }
```

```
7.    }  
8.  }
```

在main.js中加入import 'jquery'

## 或 (推荐)

1.NPM 安装 jQuery，项目根目录下运行以下代码

?

```
1  npm install jquery --save
```

2.webpack配置

在项目根目录下的build目录下找到webpack.base.conf.js文件，在开头使用以下代码引入webpack，因为该文件默认没有引用。

?

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

然后在module.exports中添加一段代码，

?

```
1  // 原有代码  
2  resolve: {  
3    extensions: ['.js', '.vue', '.json'],  
4    alias: {  
5      'vue$': 'vue/dist/vue.esm.js',  
6      '@': resolve('src')  
7    }  
8  },  
9  // 添加代码  
10 plugins: [  
11   new webpack.ProvidePlugin({  
12     $: "jquery",  
13     jQuery: "jquery",  
14     jquery: "jquery",  
15     "window.jQuery": "jquery"  
16   })  
17 ],  
18 // 原有代码  
19 module: {  
20   rules: [  
21     // .....  
22   ]  
23 }
```

然后许多其他解决办法到此就说在main.js里导入就可以了，然而题主照着做了。

main.js里导入jQuery

?

```
1 import 'jquery'
```

在Vue组件里使用 \$ or jQuery 写了操作dom的代码

接着启动项目

?

```
1 npm run dev
```

但是编译却报错了：

?

```
1 http://eslint.org/docs/rules/no-undef '$' is not defined or
2 http://eslint.org/docs/rules/no-undef 'jQuery' is not defined
```

咋回事呢？？？

### 3.eslint 检查

机智的朋友肯定想到跟eslint有关，没错，这时候需要做的下一步就是要修改根目录

下.eslintrc.js文件了，在改文件的module.exports中，为env添加一个键值对 **jquery: true** 就可以了，也就是：

?

```
1 env: {
2   // 原有
3   browser: true,
4   // 添加
5   jquery: true
6 }
```

再次 npm run dev ，OK了，没报错，赶紧去组件里试一下吧，`console.log($('选择器'))`，你会发现成功使用jQuery获取到了DOM。