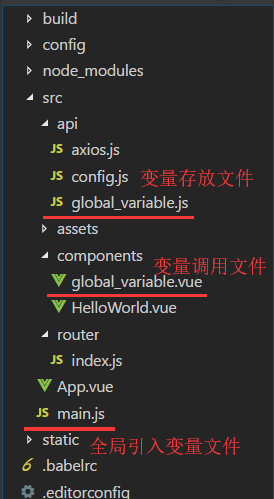
**vue项目中，定义并使用 全局变量，全局函数**

**一、定义变量，并全局使用**

**原理：**  
1. 单独新建一个全局变量模块文件，模块中定义一些变量初始状态，用export default 暴露出去。  
2. 在main.js中引入，并通过Vue.prototype挂载到vue实例上面。供其他模块文件使用；  
3. 或者直接引入到需要的模块文件中使用；

**项目目录**  


**步骤1、新建 global\_variable.js文件，用于存放变量，示例如下：**

const baseURL = 'www.baidu.com'

const token = '123456'

const userSite = '林花落了春红，太匆匆'

export default {

baseURL,

token,

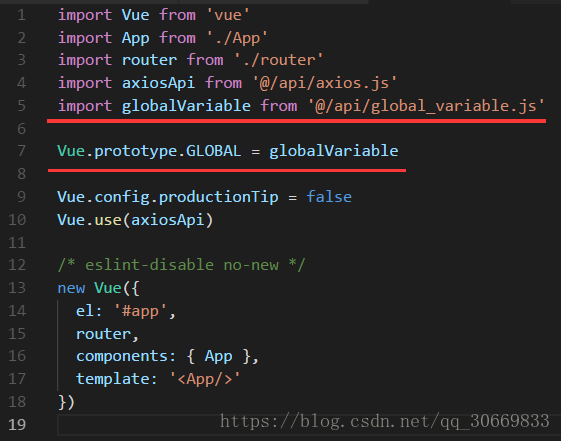
userSite

}

* **方法1：在需要使用的模块文件中使用（局部引用），示例如下**



* **方法2：全局使用，示例如下：**

1、将global\_variable.js文件引入main.js文件，并使用Vue.prototype挂在至vue实例上，示例如下：  


2、在需要使用的模块文件中使用(无需引入，直接通过this使用)，示例如下：



**二、定义函数，并全局使用**

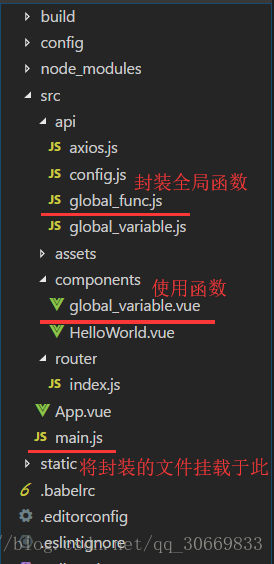
**原理：**

新建一个模块文件，然后在main.js里面通过Vue.prototype将函数挂载到Vue实例上面，通过this.输出的函数名，来运行函数。

* **方法1. 在main.js里面直接写函数**

简单的函数可以直接在main.js里面直接写，示例如下：  


* **方法2. 新建一个模块文件，挂载到main.js上面**

项目目录如下  


1、global\_func.js文件中代码示例如下：

*// param为传入参数*

function packageFunc (param) {

alert(param)

}

export default {

*// Vue.js的插件应当有一个公开方法 install。这个方法的第一个参数是 Vue 构造器，第二个参数是一个可选的选项对象。*

install: function (Vue) {

Vue.prototype.global\_func = (param) => packageFunc(param)

}

}

2、main.js文件中代码示例如下：  


3、在需要调用的模块文件中使用this.输出的函数名，调用，代码示例如下：  
