

创建 Vue.js 工程 动手实验

主讲人：和凌志

主要内容

- 如何创建 Vue 工程？
- 熟悉页面路由跳转

安装 Vue 开发环境

Vue 运行环境——安装 node.js 软件

- Vue 运行是基于 node 环境的,构建 vue 框架, 需先安装 node
- 安装 node.js <https://nodejs.org>

node.js 安装之后, npm 也会自动安装。查询版本命令:

```
$ node --version
```

最新稳定版本: v14.15.0

```
$ npm --version
```

最新稳定版本: 6.14.8

Vue 安装——全局安装脚手架工具 vue-cli

打开终端窗口。 注意： 全局安装，以管理员身份安装。

Mac 下执行： `sudo npm install -g @vue/cli`

说明： mac 下 sudo 表示以管理员身份操作

查看 vue 版本：

\$ `vue --version`

最新版本： @vue/cli 4.5.8



创建并运行 Vue 工程

命令行工具 CLI: Command Line Interface

- Vue 提供了一个官方的 CLI，为单页面应用 (SPA) 快速搭建繁杂的脚手架。它为现代前端工作流提供了 batteries-included 的构建设置。只需要几分钟的时间就可以运行起来并带有热重载、保存时 lint 校验，以及生产环境可用的构建版本。

创建 vue 工程

先创建一个工程文件， 打开终端窗口， 进入所创建的文件夹下

```
$ vue create project-name
```

【说明】 工程名字中不能有大写字母， 可用短接线

Vue CLI v4.5.8

? Please pick a preset: (Use arrow keys)

> Default ([Vue 2] babel, eslint)

Default (Vue 3 Preview) ([Vue 3] babel, eslint)

Manually select features

选择默认的 Vue 2.x 版本， 回车

安装 依赖包

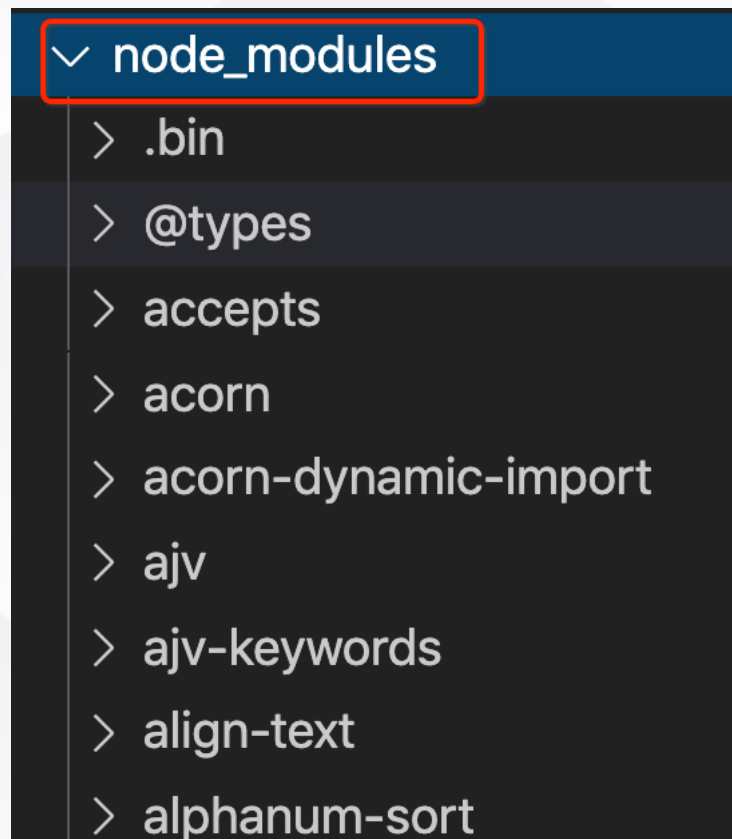
工程创建完毕后，给出以下提示：

```
$ cd project-name  
$ npm run serve
```

进入 所创建的工程 目录下，安装 package.json 包中的依赖

```
$ cd xx (进入到所创建的工程路径)
```

```
$ npm install (安装依赖包)
```



运行 Vue 工程

```
$ npm run serve
```

App running at:

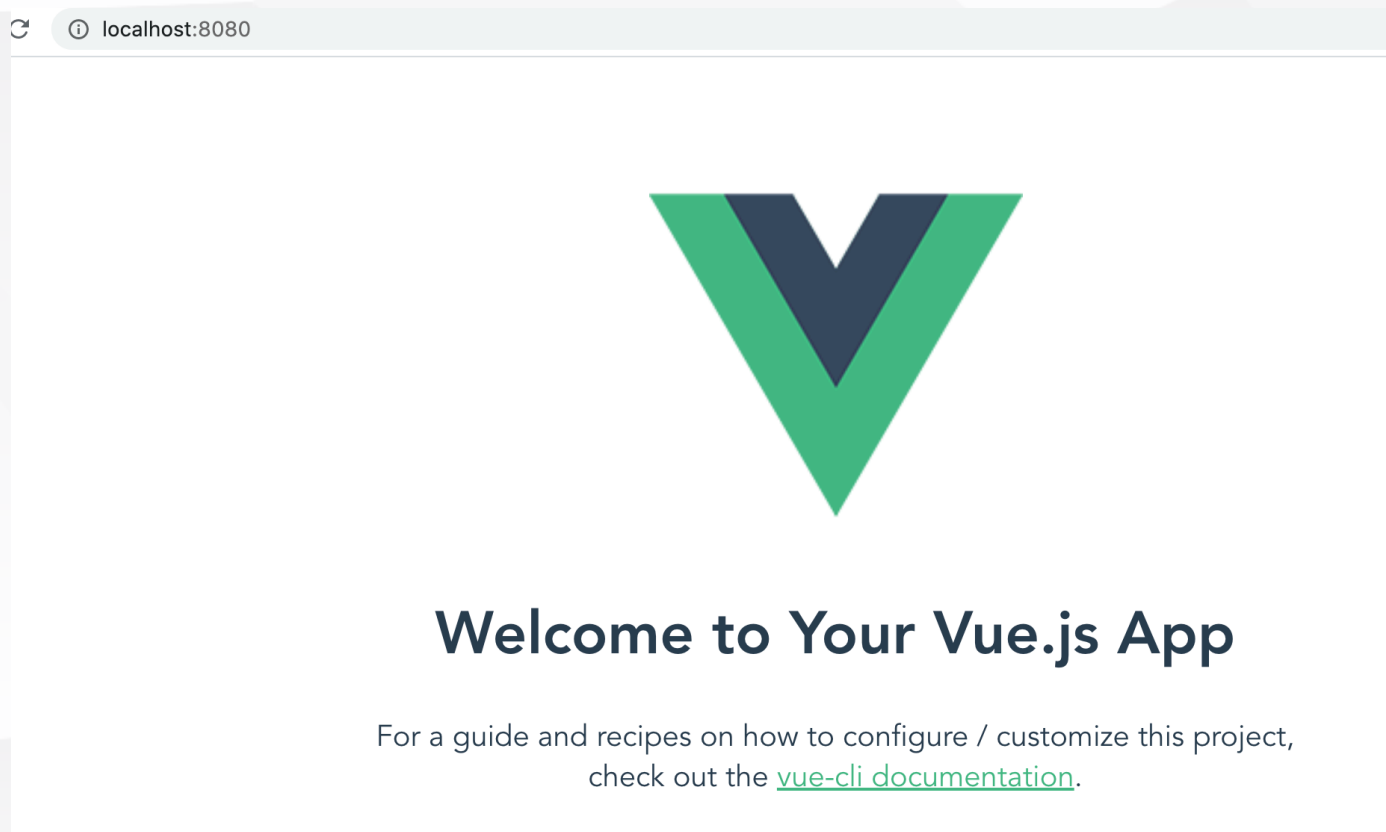
- Local: <http://localhost:8080/>
- Network: <http://192.168.3.41:8080/>

Note that the development build is not optimized.

To create a production build, run `npm run build`.

Vue 工程运行结果

在浏览器地址栏输入：`http://localhost:8080`



结束项目运行

- CTRL + C

停止项目运行



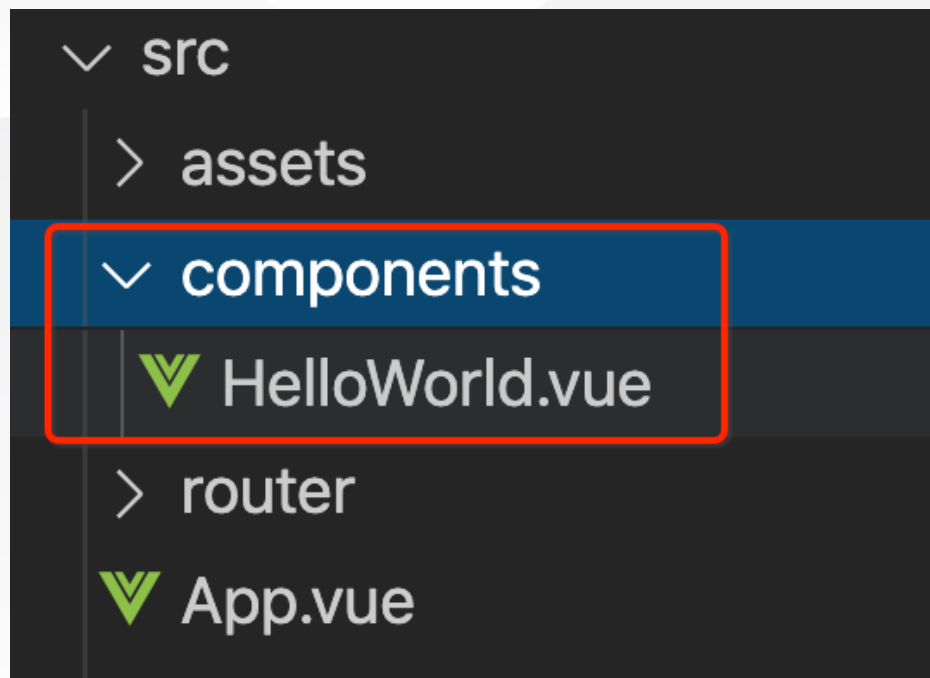
Vue 工程框架结构

vue 工程目录

```
> build
> config
> node_modules
✓ src
  > assets
  > components
  > router
  ▼ App.vue
  JS main.js
  > static
```

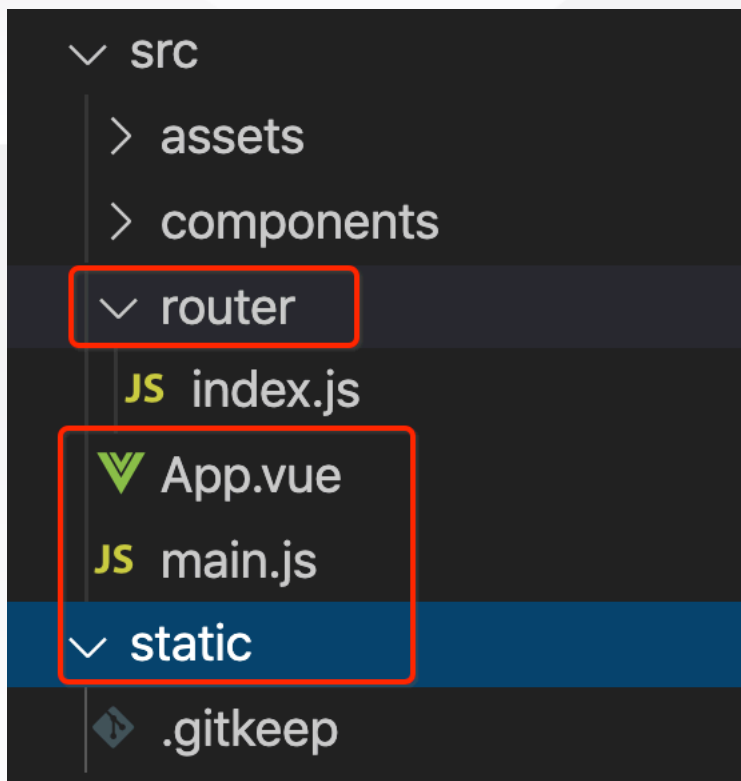
- build: 项目构建(webpack)相关代码
- config: 配置目录, 包括端口号等
- node_modules: npm加载的项目依赖块
- src: 这里是我们要开发的目录, 基本上要做的事情都在这个目录里。里面包含了几个目录及文件:
- assets: 放置一些图片, 如logo等

组件



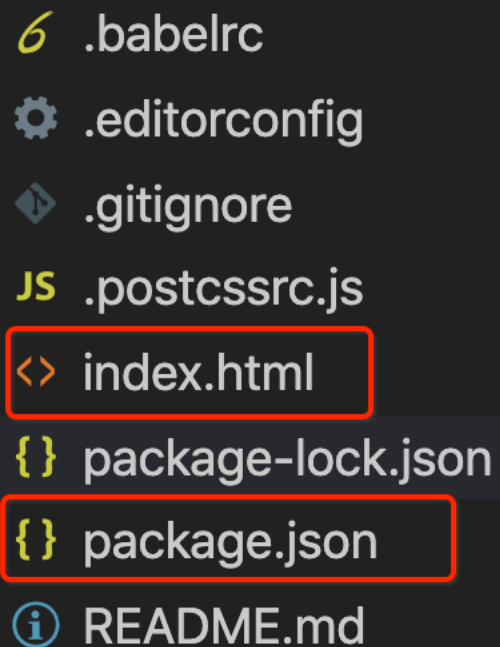
components: 该目录里存放的我们的开发文件组件，主要的开发文件都存放在这里了

常用文件



- App.vue : 项目入口文件
- main.js : 项目的核心文件
- router : 路由配置目录
- static : 放置一些静态资源文件

网页入口文件 index.html



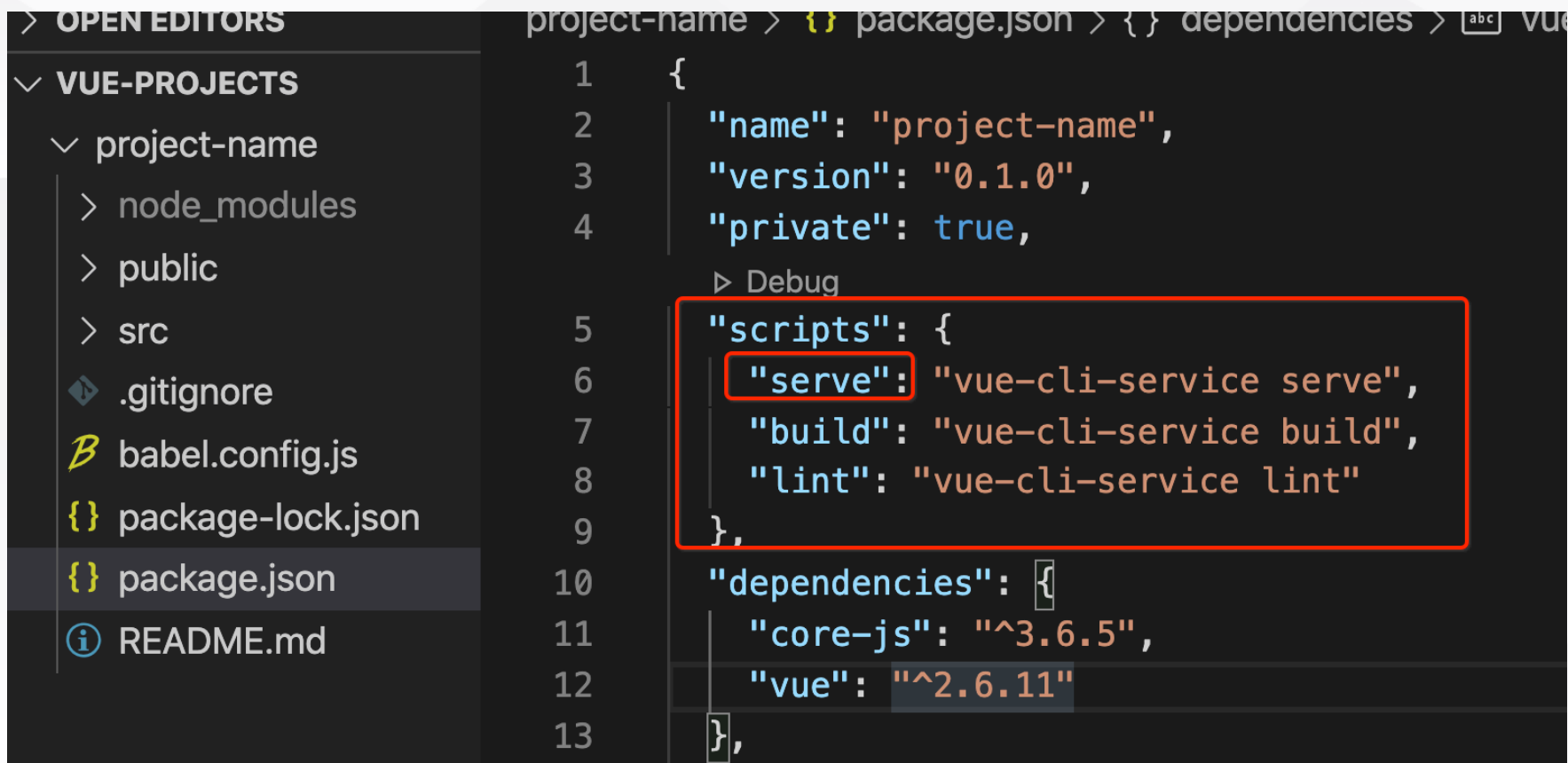
`.babelrc`
`.editorconfig`
`.gitignore`
`.postcssrc.js`
`index.html`
`package-lock.json`
`package.json`
`README.md`

- `index.html` : 首页入口文件
- `package.json` : 项目配置文件
- `README.md` : 项目的说明文档 ,
markdown 格式



Vue 工程启动流程

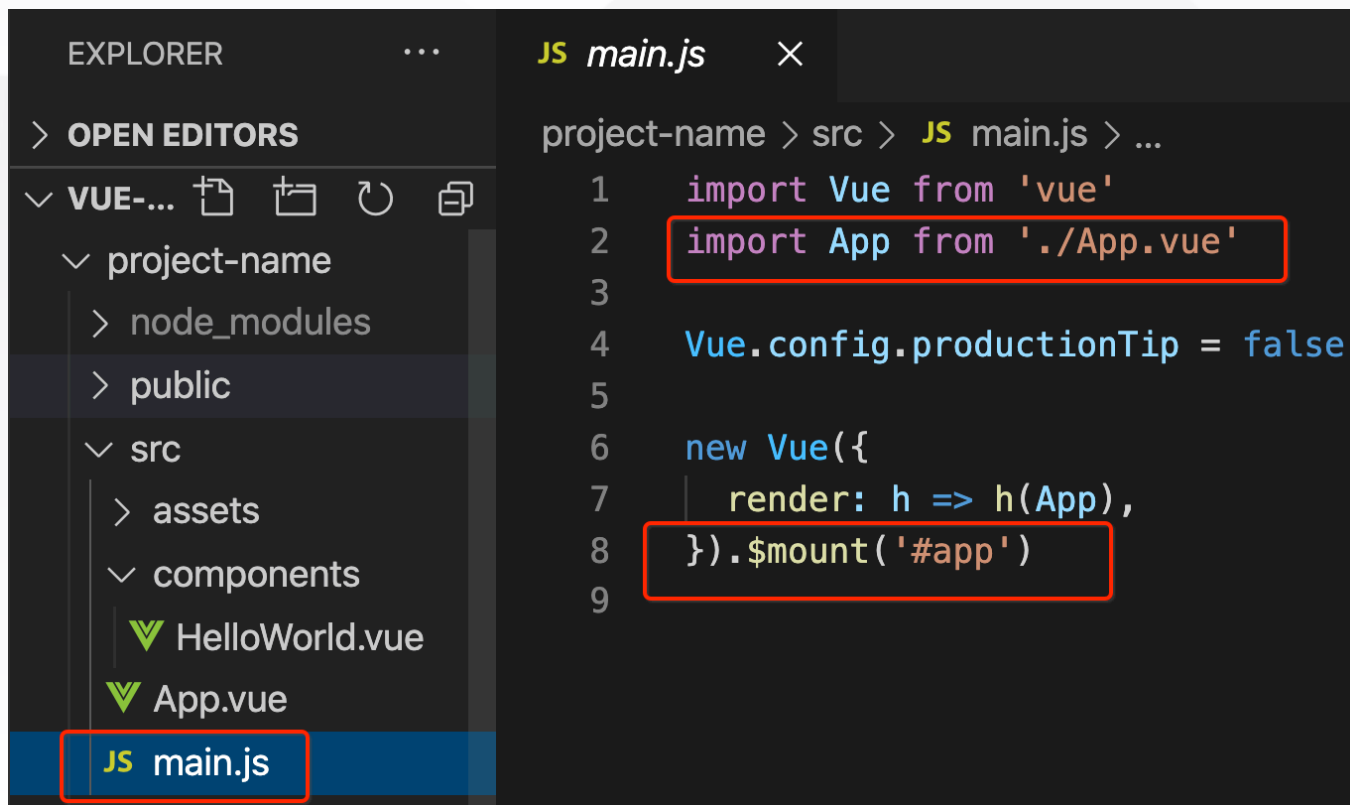
Vue 程序启动 -- package.json



```
project-name > {} package.json > {} dependencies > [abc] Vue
1  {
2    "name": "project-name",
3    "version": "0.1.0",
4    "private": true,
5    "scripts": {
6      "serve": "vue-cli-service serve",
7      "build": "vue-cli-service build",
8      "lint": "vue-cli-service lint"
9    },
10   "dependencies": {
11     "core-js": "^3.6.5",
12     "vue": "^2.6.11"
13   },
```

- 在执行 `npm run serve` 时，会去在当前文件夹下的项目中找 `package.json` 文件,启动开发服务器，默认端口是8080；

Vue 程序 main.js



The screenshot shows a code editor with a dark theme. On the left, the 'EXPLORER' sidebar is open, showing a file tree. The tree structure is as follows:

- > OPEN EDITORS
 - ✓ VUE-...
 - ✓ project-name
 - > node_modules
 - > public
 - ✓ src
 - > assets
 - ✓ components
 - ✓ HelloWorld.vue
 - ✓ App.vue
 - JS main.js (highlighted with a red box)

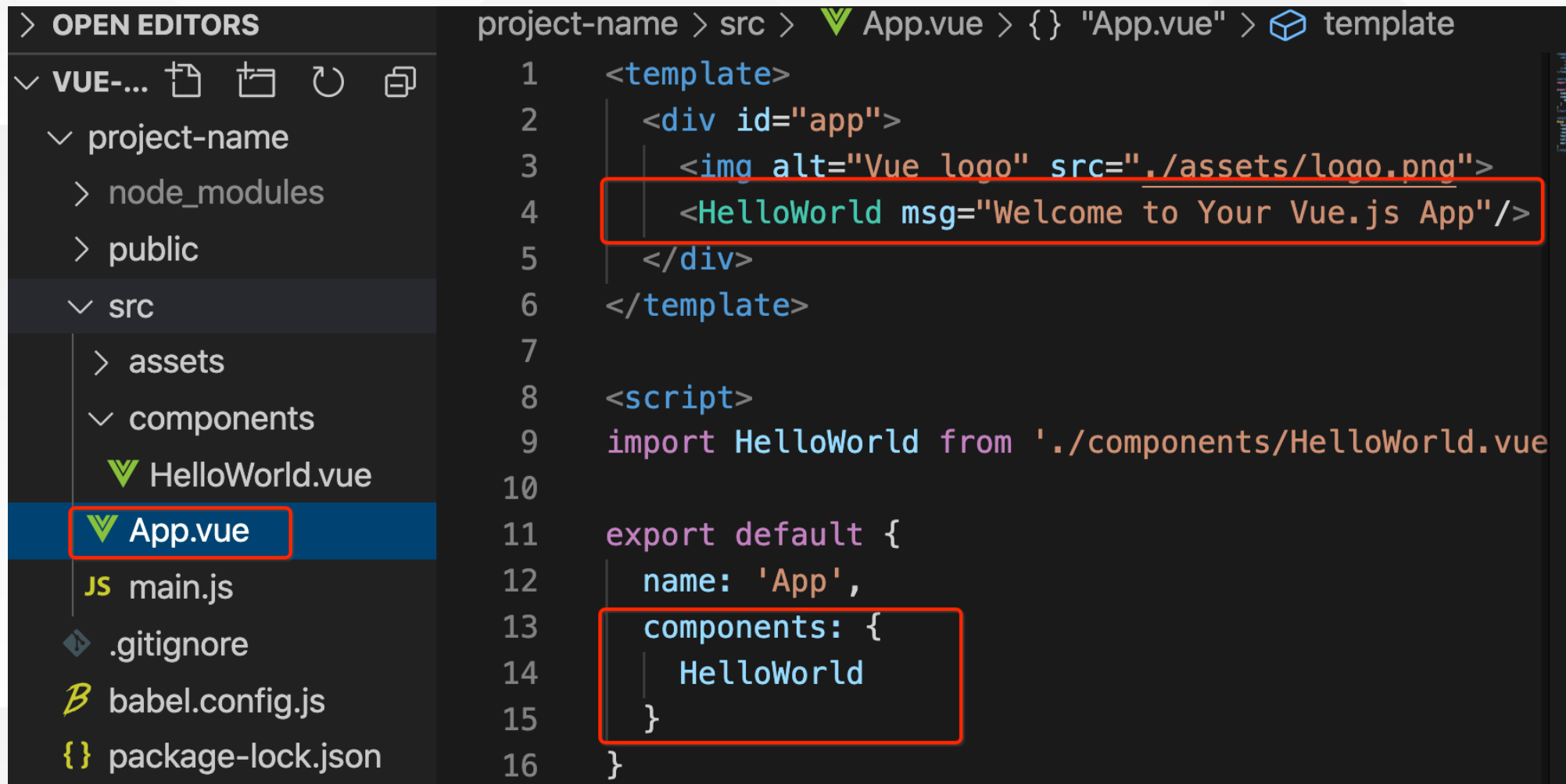
The main editor area shows the content of 'main.js' with the following code:

```
project-name > src > JS main.js > ...  
1  import Vue from 'vue'  
2  import App from './App.vue'  
3  
4  Vue.config.productionTip = false  
5  
6  new Vue({  
7    render: h => h(App),  
8  }).$mount('#app')  
9
```

Red boxes highlight the 'import App from './App.vue'' line and the 'new Vue' block. The 'new Vue' block contains the render function and the \$mount call.

- 找到 src -> main.js 文件，在该文件中 **new Vue** 的实例，要加载的模板内容 App ；

App.vue —— WEB 页面的入口



The image shows a code editor interface with a sidebar on the left and a main editor area on the right. The sidebar, titled 'OPEN EDITORS', shows a file tree for a project named 'project-name'. The tree includes folders like 'node_modules' and 'public', and files like 'main.js', '.gitignore', 'babel.config.js', and 'package-lock.json'. The 'src' folder is expanded, showing 'assets', 'components', and 'App.vue'. 'App.vue' is selected and highlighted with a red box. The main editor area shows the content of 'App.vue', which is a Vue.js component template. The template section (lines 1-6) contains a root element with an id of 'app', which includes a Vue logo and a 'HelloWorld' component. The script section (lines 8-16) imports 'HelloWorld' from './components/HelloWorld.vue' and exports a default object with the component's name and a components array containing 'HelloWorld'. Two red boxes highlight the 'HelloWorld' component usage in the template and its definition in the components array.

```
project-name > src > App.vue > {} "App.vue" > template

1  <template>
2    <div id="app">
3      
4      <HelloWorld msg="Welcome to Your Vue.js App"/>
5    </div>
6  </template>
7
8  <script>
9    import HelloWorld from './components/HelloWorld.vue'
10
11    export default {
12      name: 'App',
13      components: {
14        HelloWorld
15      }
16    }
```

动手实验

如何创建一个 Vue 工程？

- 安装 node.js
- 安装 Vue-CLI
- 创建 Vue 工程
- 运行Vue 工程

小结

- ✓ 搭建 Vue 开发环境
- ✓ 创建 Vue 工程，并运行
- ✓ 熟悉 Vue 应用 框架