

Vue小白课（二）——项目结构解析（Vue2.x）



豆包君 Lv3

2019年01月15日 13:54 · 阅读 2397

关注



@[Vue项目结构解析](#)

create by db on 2019-1-10 16:28:10

Recently revised in 2019-1-15 11:46:47

Hello 小伙伴们，如果觉得本文还不错，麻烦点个赞或者给个 star，你们的赞和 star 是我前进的动力！[GitHub 地址](#)

查阅网上诸多资料，并结合自己的学习经验，写下这篇Vue学习笔记，以记录自己的学习心得。现分享给大家，以供参考。

作为一只前端菜鸟，这是我掘金分享的第五篇文章。如有不足，还请多多指教，谢谢大家。

在上一篇项目搭建文章中，我们已经下载安装了node环境以及vue-cli，并且已经成功构建了一个vue-cli项目——见[Vue小白课（一）——CLI搭建项目（Vue2.x）](#)。那么接下来，我们来梳理一下vue-cli项目的结构。

参考文献：

- [vue-cli入门（二）——项目结构 | 思否-Reachel](#)
- [Vue-cli创建vue项目以及配置文件梳理 | 思否-AshleyLv](#)
- [Vue系列之Webpack与Eslint | 充电实践](#)

正文

Vue项目结构一览

		复制代码
└─ build	----- 项目构建(webpack)相关配置文件，配置参数什么的，一般不用动	
│ └─ build.js	----- webpack打包配置文件	
│ └─ check-versions.js	----- 检查npm,nodejs版本	
│ └─ dev-client.js	----- 设置环境	
│ └─ dev-server.js	----- 创建express服务器，配置中间件，启动可热重载的服	
│ └─ utils.js	----- 配置资源路径，配置css加载器	
│ └─ vue-loader.conf.js	----- 配置css加载器等	
│ └─ webpack.base.conf.js	----- webpack基本配置	
│ └─ webpack.dev.conf.js	----- 用于开发的webpack设置	
│ └─ webpack.prod.conf.js	----- 用于打包的webpack设置	
└─ config	----- 配置目录，包括端口号等。我们初学可以使用默认的。	
│ └─ dev.env.js	----- 开发环境变量	
│ └─ index.js	----- 项目配置文件	
│ └─ prod.env.js	----- 生产环境变量	
│ └─ test.env.js	----- 测试环境变量	
└─ node_modules	----- npm 加载的项目依赖模块	
└─ src	----- 我们要开发的目录，基本上要做的事情都在这个目录里。	
│ └─ assets	----- 静态文件，放置一些图片，如logo等	
│ └─ components	----- 组件目录，存放组件文件，可以不用。	
│ └─ main.js	----- 主js	
│ └─ App.vue	----- 项目入口组件，我们也可以直接将组件写这里，而不使用 compone	
│ └─ router	----- 路由	
└─ static	----- 静态资源目录，如图片、字体等。	
└─ .babelrc	----- babel配置文件	
└─ .editorconfig	----- 编辑器配置	
└─ .gitignore	----- 配置git可忽略的文件	
└─ index.html	----- 首页入口文件，你可以添加一些 meta 信息或统计代码啥的。	

在webpack的配置文件里，设置了main.js是入口文件，我们的项目默认访问index.html，这个文件里面 `<div id="app"></div>` 和App.vue组件里面的容器完美的重合了，也就是把组件挂载到了index页面，然后我们只需要去建设其他组件就好了，在App组件中我们也可以引入，注册，应用其他组件，可以通过路由将其他组件渲染在App组件，这样我们就只需要去关注每个组件的功能完善。

就是说vue的默认页面是index.html，index中的 `<div id="app"></div>` 挂载了App.vue这个大组件，然后所有的其他子组件（hello.vue等）都归属在App.vue这个主组件下。

主要文件详解

src——[项目核心文件]

在vue-cli的项目中，其中src文件夹是必须要掌握的，因为基本上要做的事情都在这个目录里。

index.html——[主页]

index.html如其他html一样，但一般只定义一个空的根节点，在main.js里面定义的实例将挂载在根节点下，内容都通过vue组件来填充

[复制代码](#)

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>vuedemo</title>
  </head>
  <body>
    <!-- 定义的vue实例将挂载在#app节点下 -->
    <div id="app"></div>
  </body>
</html>
```

App.vue——[根组件]

一个vue页面通常由三部分组成:模板(template)、js(script)、样式(style)

```
<!-- 模板 -->
<template>
  <div id="app">
    
    <router-view></router-view>
  </div>
</template>

<!-- script -->
<script>
export default {
  name: 'app'
}
</script>

<!-- 样式 -->
<style>
#app {
  font-family: 'Avenir', Helvetica, Arial, sans-serif;
  -webkit-font-smoothing: antialiased;
  -moz-osx-font-smoothing: grayscale;
  text-align: center;
  color: #2c3e50;
  margin-top: 60px;
}
</style>
```

【template】

其中模板只能包含一个父节点，也就是说顶层的div只能有一个（例如上面代码，父节点为#app的div，其没有兄弟节点）

`<router-view></router-view>` 是子路由视图，后面的路由页面都显示在此处

打一个比喻吧，`<router-view>` 类似于一个插槽，跳转某个路由时，该路由下的页面就插在这个插槽中渲染显示

【script】

vue通常用es6来写，用 `export default` 导出，其下面可以包含数据data，生命周期(mounted等)，方法(methods)等，具体语法请看vue.js文档。

样式通过style标签包裹，默认是影响全局的，如需定义作用域只在该组件下起作用，需在标签上加scoped。

```
<style scoped></style>
```

如要引入外部css文件，首先需给项目安装css-loader依赖包，打开cmd，进入项目目录，输入 `npm install css-loader` ,回车。

安装完成后，就可以在style标签下import所需的css文件，例如：

[复制代码](#)

```
<style>
  import './assets/css/public.css'
</style>
```

main.js——[入口文件]

main.js主要是引入vue框架，根组件及路由设置，并且定义vue实例，下面的 `components: {App}` 就是引入的根组件App.vue

后期还可以引入插件，当然首先得安装插件。

[复制代码](#)

```
/*引入vue框架*/
import Vue from 'vue'
/*引入根组件*/
import App from './App'
/*引入路由设置*/
import router from './router'

/*关闭生产模式下给出的提示*/
Vue.config.productionTip = false

/*定义实例*/
new Vue({
  el: '#app',
  router,
  template: '<App/>',
  components: { App }
})
```

vue-router是Vue.js官方的路由插件，它和vue.js是深度集成的，适合用于构建单页面应用。vue的单页面应用是基于路由和组件的，路由用于设定访问路径，并将路径和组件映射起来。

router文件夹下，有一个index.js，即为路由配置文件。

[复制代码](#)

```
/*引入vue框架*/
import Vue from 'vue'
/*引入路由依赖*/
import Router from 'vue-router'
/*引入页面组件，命名为Hello*/
import Hello from '@/components/Hello'

/*使用路由依赖*/
Vue.use(Router)

/*定义路由*/
export default new Router({
  routes: [
    {
      path: '/',
      name: 'Hello',
      component: Hello
    }
  ]
})
```

这里定义了路径为 '/' 的路由，该路由对应的页面是Hello组件，所以当我们在浏览器url访问 <http://localhost:8080/#/> 时就渲染的Hello组件

类似的，我们可以设置多个路由，'/index','/list'之类的，当然首先得引入该组件，再为该组件设置路由。

其他配置文件

主要包括webpack的配置，项目配置，项目依赖等等。

详情可参考以下文章：

- [Vue-cli创建vue项目以及配置文件梳理 | 思否-AshleyLv](#)

这是我自己做的一个vue模板文件，符合Eslint规则

[复制代码](#)

```
<!-- -->
<template>
  <div/>
</template>

<script>
// 这里可以导入其他文件（比如：组件，工具js，第三方插件js，json文件，图片文件等等）
// 例如：import 《组件名称》 from '《组件路径》'

export default {
  // import引入的组件需要注入到对象中才能使用
  components: {},
  data() {
    // 这里存放数据
    return {

    }
  },
  // 监听属性 类似于data概念
  computed: {},
  // 监控data中的数据变化
  watch: {},
  // 生命周期 - 创建完成（可以访问当前this实例）
  created() {

  },
  // 生命周期 - 挂载完成（可以访问DOM元素）
  mounted() {

  },
  beforeCreate() {}, // 生命周期 - 创建之前
  beforeMount() {}, // 生命周期 - 挂载之前
  beforeUpdate() {}, // 生命周期 - 更新之前
  updated() {}, // 生命周期 - 更新之后
  beforeDestroy() {}, // 生命周期 - 销毁之前
  destroyed() {}, // 生命周期 - 销毁完成
  activated() {}, // 如果页面有keep-alive缓存功能，这个函数会触发
  // 方法集合
  methods: {

  }
}

</script>
```

</style>

总结

vue-cli给创建vue项目提供了很大的便利。但是同时大量的第三方库的使用，会让打包后的js变的很大，所以还是要熟悉配置,熟悉第三方插件的使用，才可以开发更高效的开发web应用。这里把vue-cli的一些相关内容给自己做一个总结，便于以后查阅。也是希望对其他开发者有帮助。有不足之处请指正。

路漫漫其修远兮，与诸君共勉。

祝大家2019更上一层楼！

后记：Hello 小伙伴们，如果觉得本文还不错，记得点个赞或者给个 star，你们的赞和 star 是我编写更多更丰富文章的动力！[GitHub 地址](#)



db 的文档库 由 db 采用 [知识共享 署名-非商业性使用-相同方式共享 4.0 国际 许可协议](#) 进行许可。

基于github.com/danygitgit上的作品创作。

本许可协议授权之外的使用权限可以从 creativecommons.org/licenses/by... 处获得。

分类： 前端 标签： [Vue.js](#)

文章被收录于专栏：



vue相关

vue相关文章

[关注专栏](#)

评论



探 Q

登录

全部评论 1

最新

最热

洛笙11510月前

感谢大佬

点赞

回复

相关推荐

Mottle9月前前端Vue.js

2x1 VueRouter - 进阶 - 路由元信息

6082评论

Leecason1月前Vue.jsJavaScript

来自 Vue 官方团队的 57 个技术分享

5.4w83142

前端胖头鱼2月前前端JavaScriptVue.js

11+ chrome高级调试技巧，学会效率直接提升666%

4.6w1736124

mapengfei2月前前端Vue.js

大菠萝？Pinia已经来了，再不学你就out了

6.9w916174

杰出D11月前前端算法

面试了十几个高级前端，竟然连（扁平数据结构转Tree）都写不出来

18.0w29831564

前端进击者1年前Vue.js

绝对干货~！学会这些Vue小技巧，可以早点下班和女神约会了

字节一面，面试官问我Vue3源码，我说.....

11.8w 1555 219

doodlewind 1年前 Vue.js 面试

他写出了 Vue，却做不对这十道 Vue 笔试题

7.1w 773 233

海明月 5月前 面试 前端

三十七个常见Vue面试题，背就完事了。

7.3w 1330 80

Mottle 11月前 前端

2x1 精读Vue官方文档 - 状态过度

599 3 评论

IDuxFE 5月前 前端 Vue.js

浅析我们为什么要上Vue3？

5.7w 422 151

monkeysoft 1年前 Vue.js

史上最强vue总结---面试开发全靠它了

14.1w 3194 115

Sunshine_Lin 11月前 前端 面试 Vue.js

「自我检验」熬夜总结50个Vue知识点，全都会你就是神！！！！

10.9w 3020 205

杭州程序员张张 4年前 Vue.js JavaScript 前端

Vue.js中 watch 的高级用法

10.6w 931 44

漆黑之牙_ 1年前 前端

Vue、Node全栈项目~面向小白的博客系统~

codexu 1年前 Vue.js

基于Vue的前端架构，我做了这15点

5.9w 2441 279

前端论道 2年前 Vue.js CSS

【建议收藏】90%的前端都会踩的坑，你中了吗？

5.1w 1640 80

金色小芝麻 1年前 Vue.js

30张脑图带你从零开始学VUE | VUE基础知识篇

3.5w 886 119

黄轶 8月前 Vue.js 前端框架 JavaScript

是什么事让尤大如此生气？

7.7w 640 155

chenhongdong 4年前 Vue.js JavaScript 前端

不好意思！耽误你的十分钟，让MVVM原理还给你

7.7w 2112 291



10



1



收藏