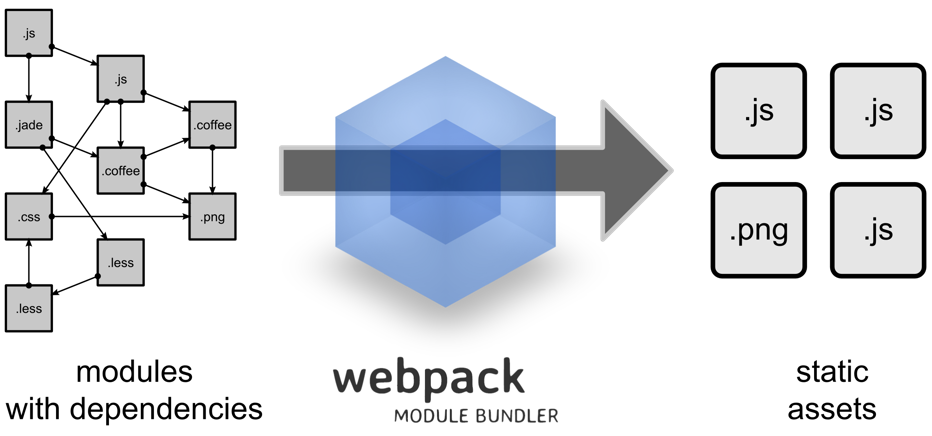
1. 首先需要node.js环境，使用npm安装webpack

网络原因下载cnpm：npm install cnpm –g

node.js是javascript的一种运行环境，是对Google V8引擎进行的封装。是一个服务器端的javascript的解释器。node.js包含npm

npm是node.js的包管理器，引入优秀的库，避免重复造轮子。



webpack是一个打包工具，可以帮你把你的项目(**这里的项目其实就是指通过模块化开发的项目)**的复杂依赖关系的静态文件打包为简洁版的浏览器可识别的静态资源；可以将一些浏览器尚不支持的新特性进行转换，提高兼容性

1. 安装脚手架vue-cli
2. cnpm install --global vue-cli 安装脚手架
3. vue init webpack my-project 创建项目；默认回车

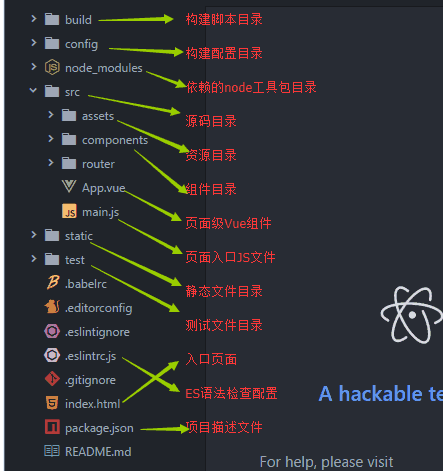
ESLint：是否使用ESLint管理代码，ESLint是个代码风格管理工具，是用来统一代码风格的，一般项目中都会使用。

1. $cd my-project

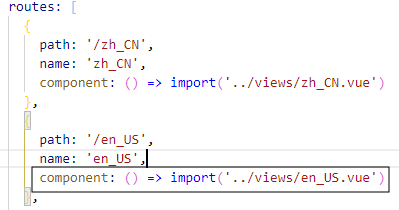
$ cnpm install

$ cnpm run dev

进入项目，安装并运行



添加路由：



当有多个页面需要添加路由的时候，可以用上图的方式通过箭头函数进行声明；

代替：

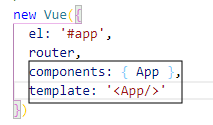


组件化开发：

关于单页面开发：

每个页面都是一个组件，每个公共样式也是一个组件。

主页面index对应的跟组件是一个app.vue的单文件组件；<template>部分配置<route-view/>，规定其他分页面展示的地方；<style>可以规定公用样式；而它根组件的地位是由页面入口main.js文件定义的



页面入口js文件：main.js不仅定义了根组件，也可以在其内定义全局组件和挂载路由。在渲染的时候会把名为App.vue的文件（跟组件）渲染到index中的id=app的模块中。

路由：其他分页面不能同时在一个页面显示，需要通过列表中的链接点击等方式选择性的显示，该项目便是通过一个菜单栏进行<route-link>多个分页面，通过点击选择需要渲染哪个分页面显示。

其中router目录下的index.js就是用来定义路由、创建路由实例的。

分页面和全局组件: 则是页面的真正内容，每个分页面代表不同的内容，全局组件的定义让每个页面有相同的部分可以直接引用(便于维护)

引用第三方资源：通过cnpm install xxx下载后引用

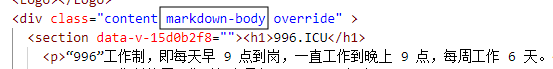
1. 调用的github-markdown-css 样式：进行格式优化

链接：https://github.com/sindresorhus/github-markdown-css

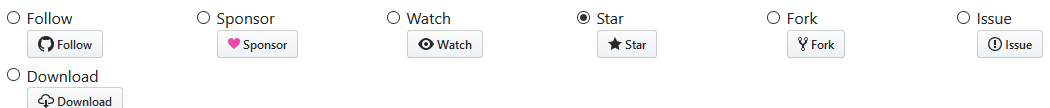
npm install github-markdown-css安装后

在入口js文件中import 'github-markdown-css'

在需要的地方通过**markdown-body类**调用



1. 调用的vue-github-button样式：获取github上面的按钮样式



链接：<https://buttons.github.io/>

需要创建一个单文件组件，分别填上template模块和局部组件定义，然后再去入口js文件把该单文件组件定义为全局组件。

开发完成之后进行打包：vue run build

在此之前需要配置：

1. 在路由组件中定义一个base: "/996icu" 一个基础路径
2. 在该项目名(/996icu)路径下创建一个vue.config.js的文件

并在里面填写：

