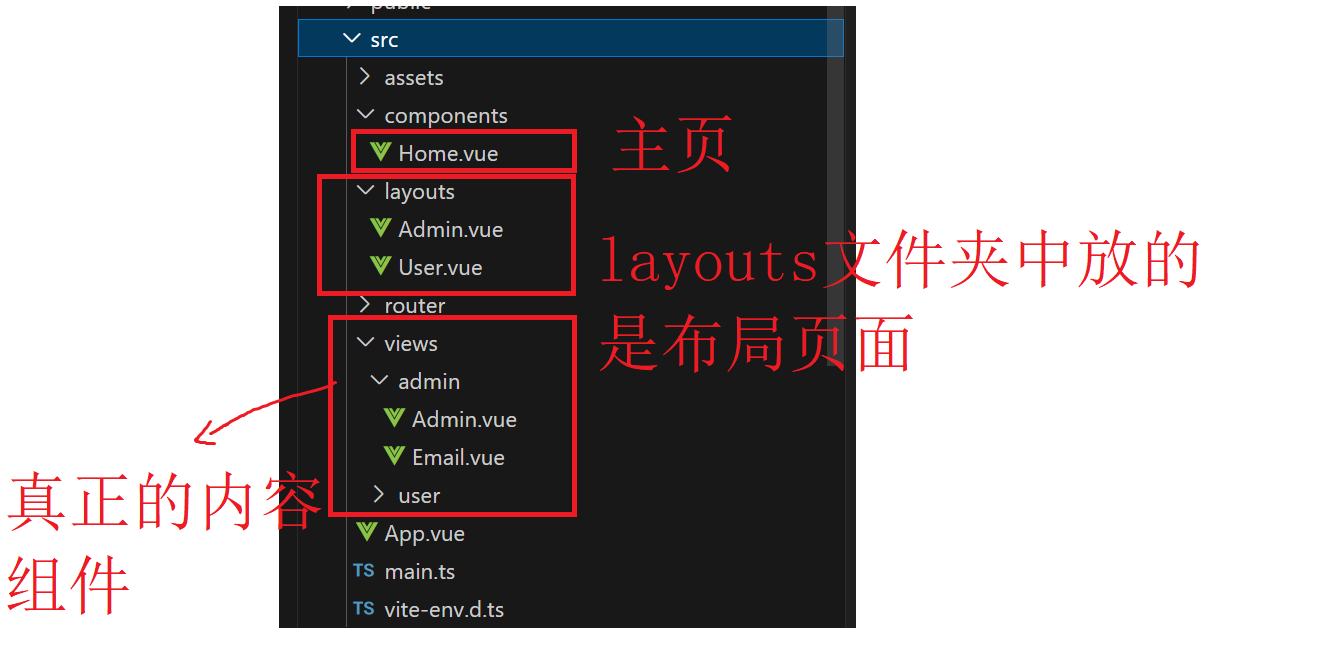
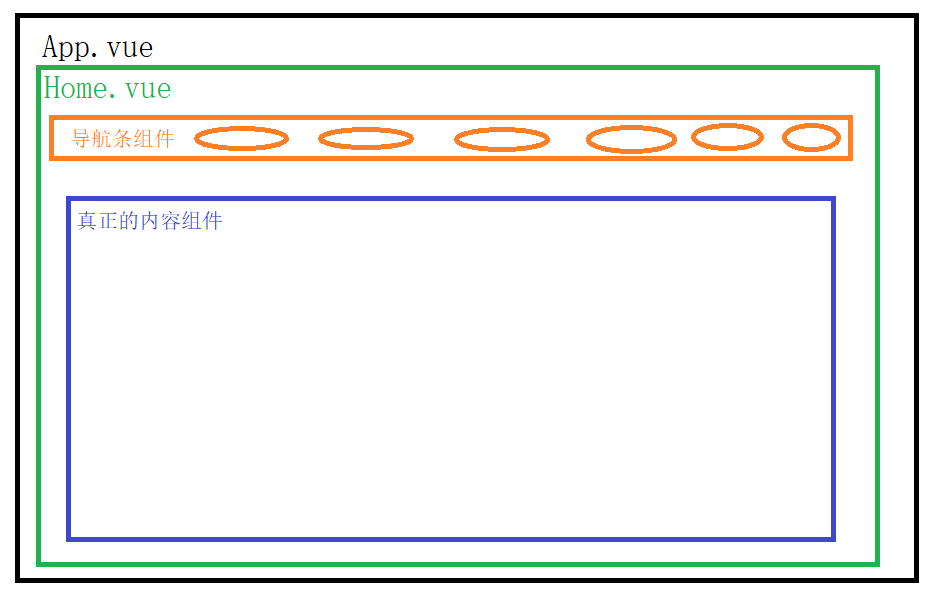
前言：在使用vue-router前端路由的时候，注册路由和使用路由本身不是一件很复杂的事。但在开发web应用的前期，如何快速的注册路由，提升开发效率是一件不可忽视的事。这篇文章将介绍才能实现前端路由的自动注册。

在正式介绍如何实现前端路由的自动注册之前，我们先来看一下项目文件的结构。

注：这个小demo的构建工具是使用vite+ts



我们大概就想实现下图这样的一个页面

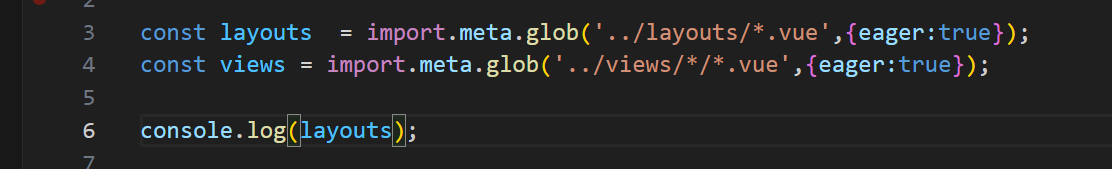


但是由于我们这篇文章不是重点将布局的，所以页面不会布局的很精美，大概是那么个意思就可以。

使用vite构建工具提供的import.meta.glob()方法实现文件的批量引入。

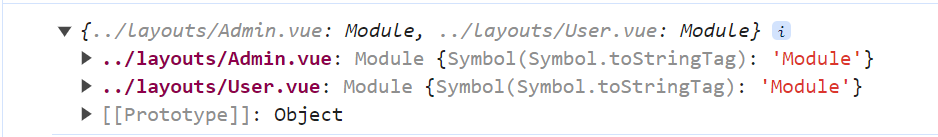
什么是vite给我们提供的import.meta.glob()方法？

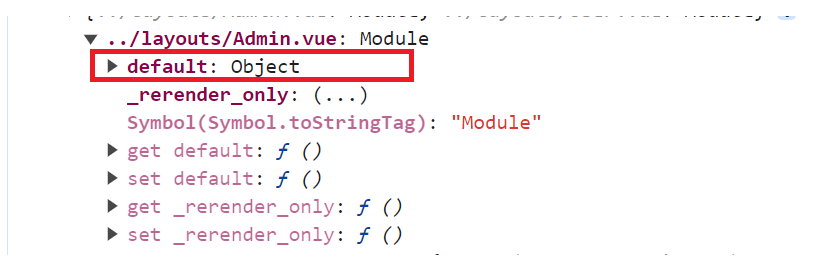
Import.meta.glob()支持从你给定的文件路径上一次性抓取多个模块。它的返回值是一个对象，对象中的属性名，就是抓取到的module的路径，与之匹配的则是抓取到的module。



后面的参数就是指定是否同步导入，eager设置为true就是同步导入，如果不设置默认为false，就是异步导入。这个看你个人，如果异步导入的话，它返回的值就是一个promise，后续我们需要用then来接受模块导出的内容，我感觉这样有点太麻烦了，所以我直接使用同步导入。

我们可以打印出来看一下，

  
module对象上有一个default属性，它就是模块导出内容。

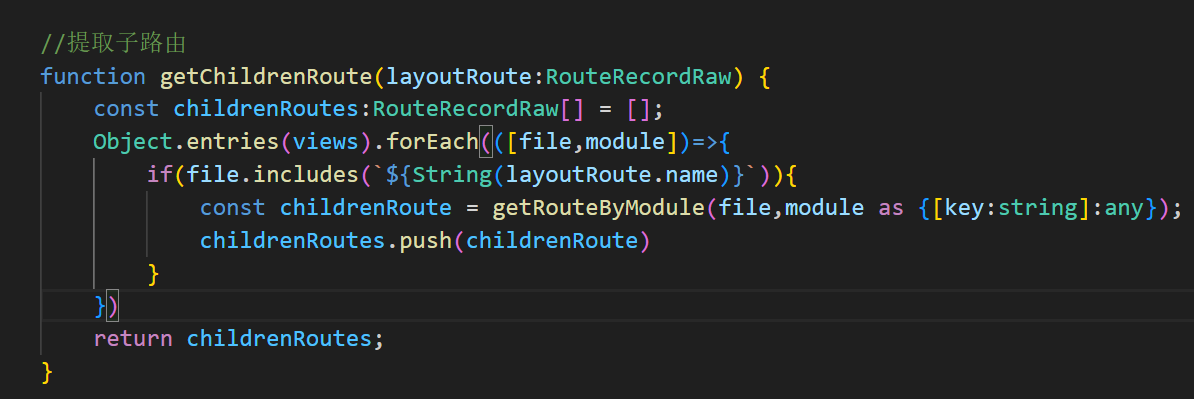


根据路径自动生成路由对象。

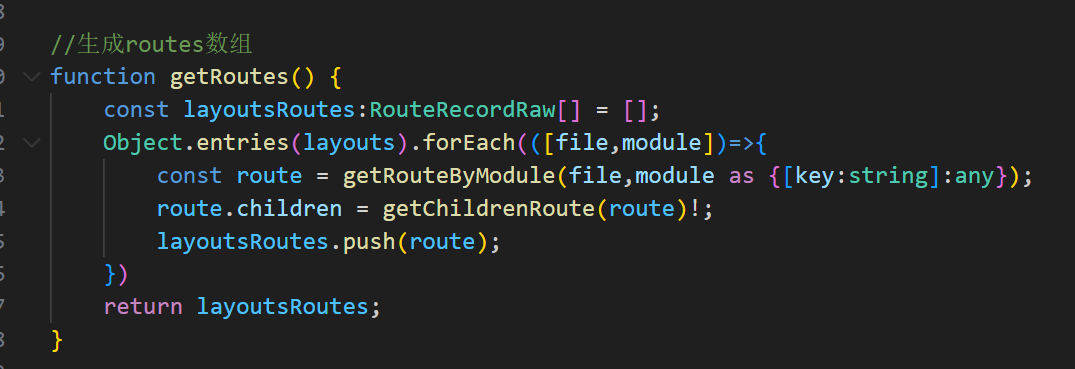
根据路径自动生成路由对象的处理很简单，我们可以截取路径上的文件名字符串作为路由的名称，同时也用它作为路由对象path属性的地址。路由对象上的component属性的值则使用module.default的值。我们可以创建一个单独的方法用于处理提取路由的操作。



根据路径自动创建子路由对象。

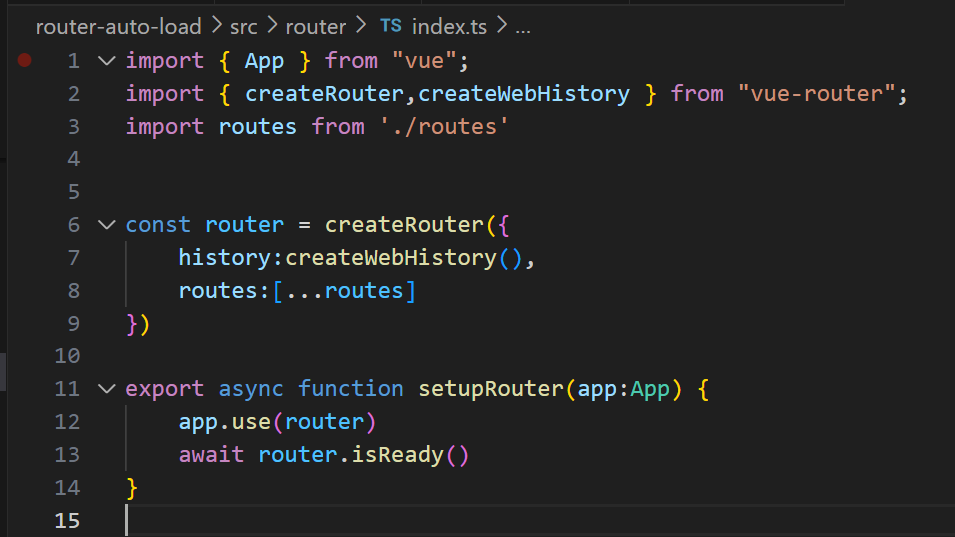


生成routes数组



最后我们将这个生成routes数组的方法导出即可。

封装路由注册文件，实现路由的自动注册。

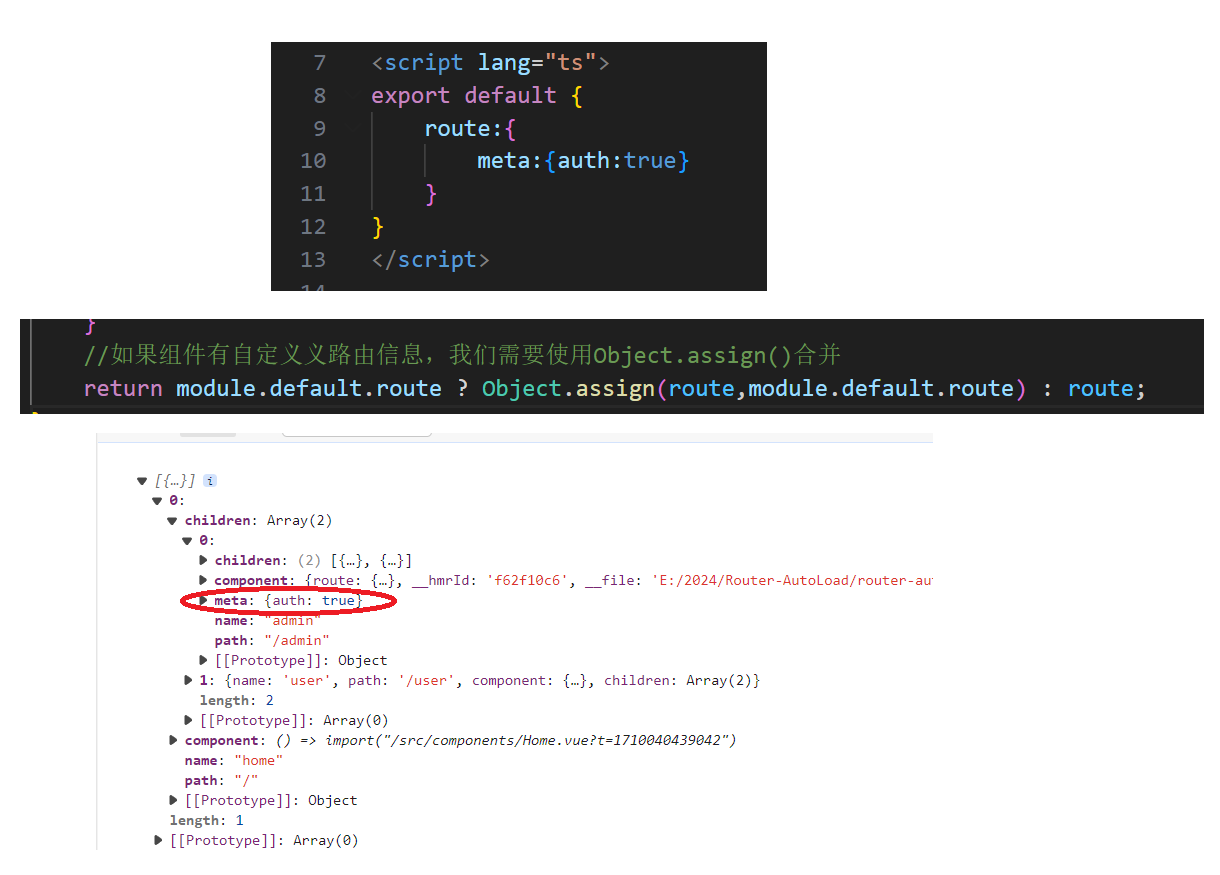


最终的效果：



在组件中定义路由的元信息，实现路由对象的自定义。

我们知道vue组件里的script标签中的默认导出，是挂载在组件实例上的。所以我们在组件的script标签中，默认导出一个，自定的路由对象，然后使用Object.assign()来完成，对象”{}”的合并。因为Object.assign()合并对象会把第二个对象中的属性添加到第一个属性中，如果重复了，就会拿第二个覆盖掉第一个（有关Object.assigon()的细节，可以去mdn官网搜索阅读）。

**