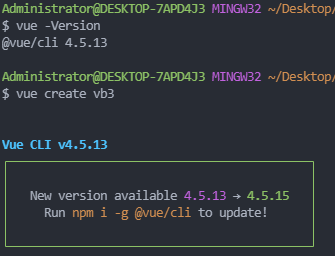
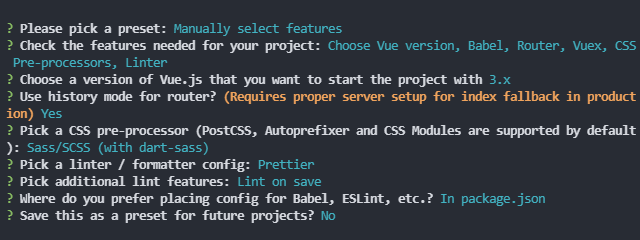
1、创建新的vue3项目

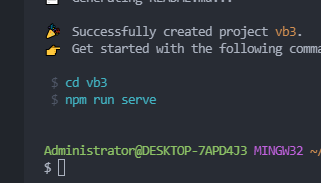
升级/安装脚版本较高的手架  
 npm update -g @vue/cli  
2、创建项目



终端输入 vue create vb3

创建一个名为 vb3 的项目

  
 选择手动选择，并配置其他内容



创建完成

1. 2.x 过渡到3.x 及其区别
   1. vue3.x 和 vue2.x 目录结构无太大变化
   2. 取消了 beforeCreate 和 create 改为使用 setup 方法；

beforeMount 改为 onBeforeMount

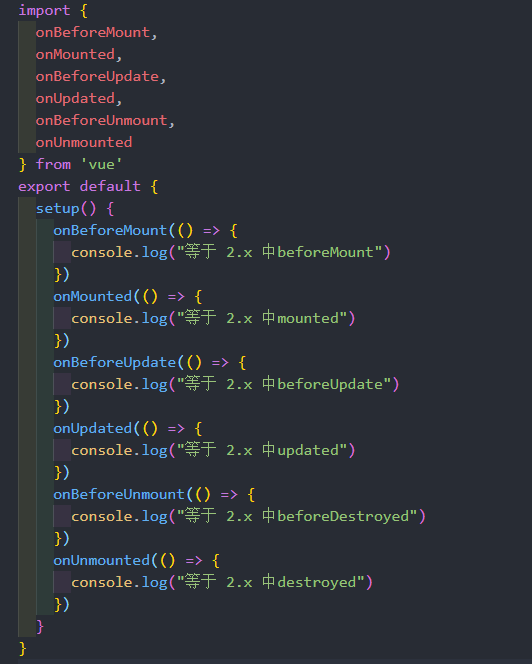
mounted改为 onMounted

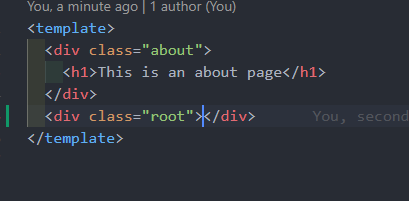
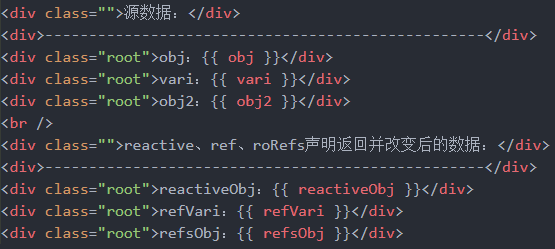
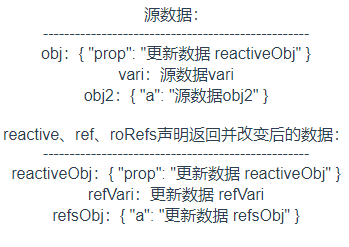
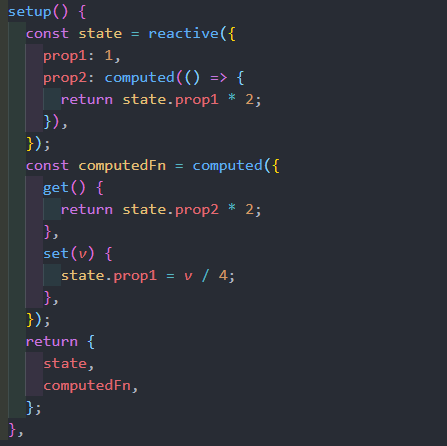
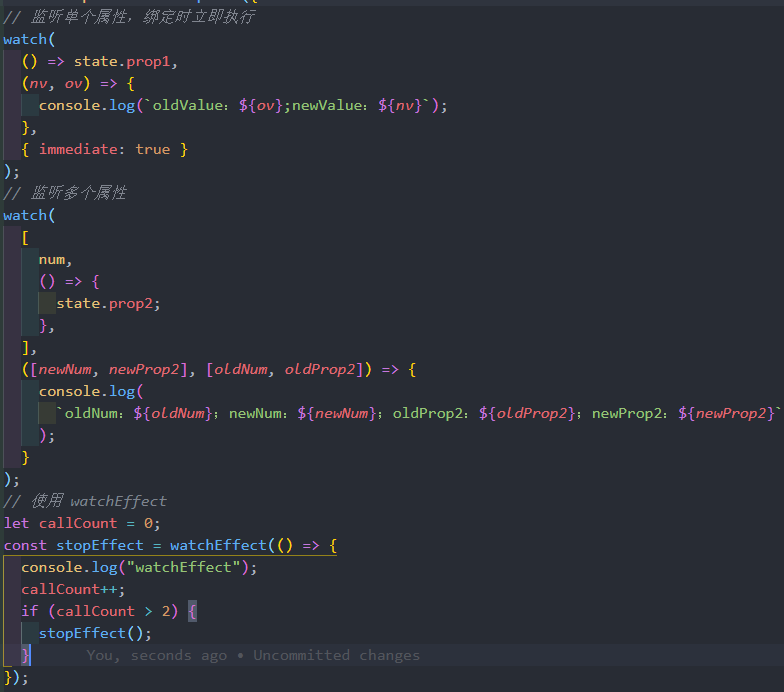
beforeUpdate 改为 onBeforeUpdate

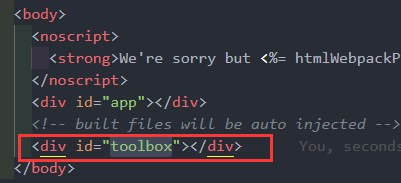
updated改为 onUpdated

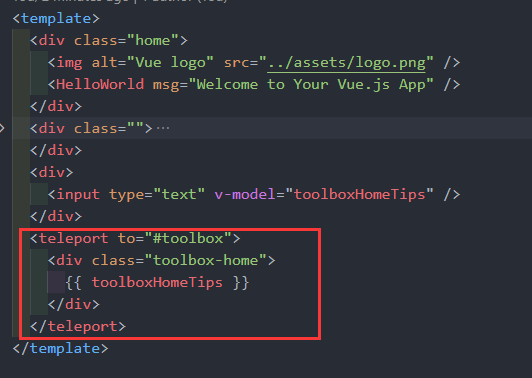
beforeDestroy改为 onBeforeDestroy

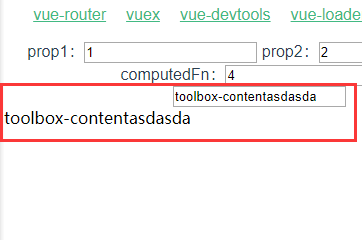
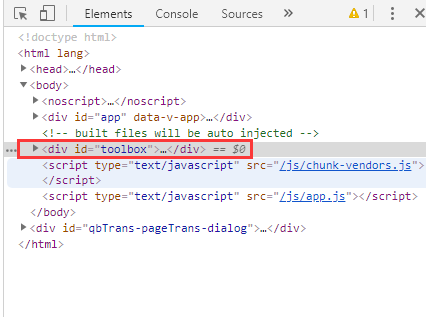
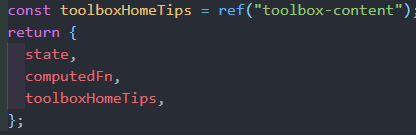
destroyed改为 onUnmounted



* 1. 3.x 新增钩子函数 onRenderTracked、onRenderTriggered。用于页面内容更新后，跟踪新值和旧值
  2. 相比与2.x ，3.x 代码中 template 支持多个根节点  
        
     （3.x 中使用多个根节点， eslint不报错；注意：如果当前工作区跟项目根目录不是一个目录，eslint同样会报只允许一个根节点的错误。需要把项目根目录当做vscode的工作空间来打开）
  3. 响应式区别  
      3.x 中响应式属性或对象需要使用reactive，ref，toRef，toRefs显式声明  
       
      直接声明的obj ；使用reactive声明的reactiveObj；  
      input1文本框绑定obj.prop ，input2文本框绑定reactiveObj.prop   
      改变 input1 的值，只改变obj.prop 且不更新视图内容；  
      改变 input2 的值，改变 reactiveObj.prop会同时改变 obj.prop 且更新视图  
       
      直接声明的vari ；使用ref声明的refVari；  
      input1文本框绑定vari ，input2文本框绑定refVari   
      改变 input1 的值，只改变vari 且不更新视图内容；  
      改变 input2 的值，改变 refVari的值 更新视图  
       
      直接声明的obj2 ；使用reactive声明的refsObj；  
      input1文本框绑定obj2.a，input2文本框绑定refsObj.a  
      改变 input1 的值，只改变ob2j.a且不更新视图内容；  
      改变 input2 的值，只改变refsObj.a且不更新视图内容；
  4. 3.x 计算属性  
      
  5. 3.x 监听
     1. 相比于2.x 不支持 “obj.obj2.obj3”的监听方式
     2. 能监听多个属性
     3. 多了一个watchEffect属性，watchEffect注册后立即调用；watchEffect可以停止监听  
        
  6. 3.x 新增 <teleport> 组件
     1. 用于在 <div id=”app”></div> 标签之外渲染其他的响应元素
     2. 在index <body> 标签内 添加自定义id 的div或其他标签
     3. 在vue页面template 标签内部新增 <teleport>标签，并且属性to应等于之前在index中定义的id

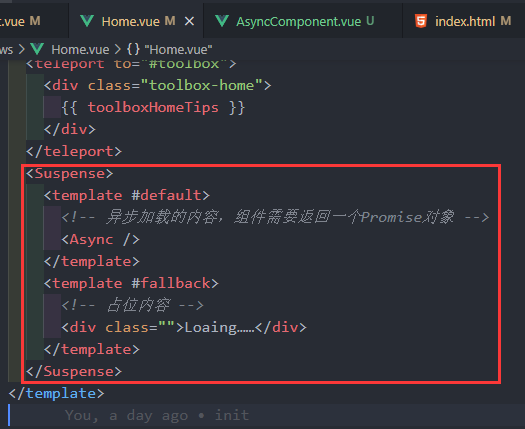




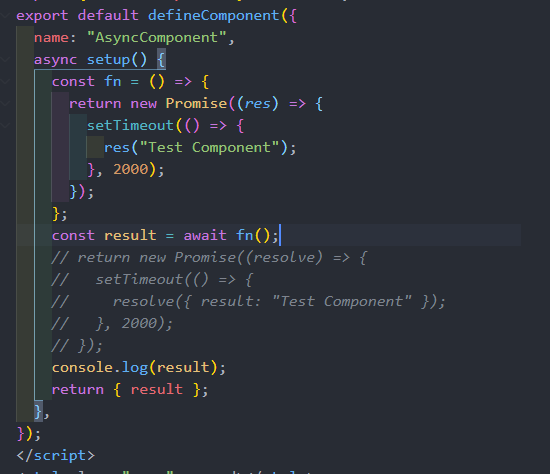
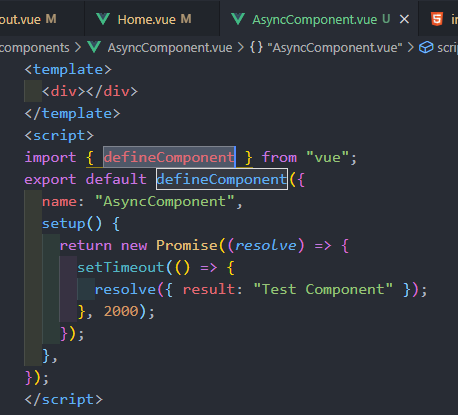
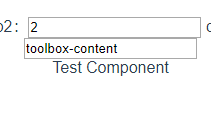
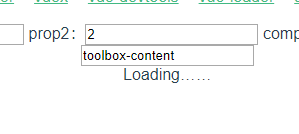


（toolbox 和 app 同级， 且内部的内容是响应式的）

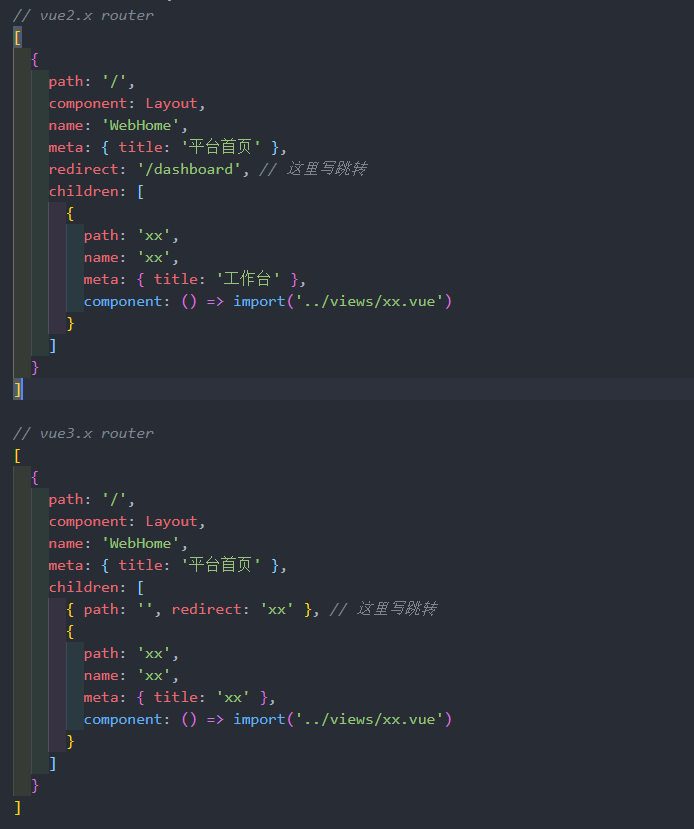
* 1. 新增<Suspense>
     1. 用于异步组件加载过程中占位功能。
     2. Default 是异步组件的内容，fallback 是静态占位内容
     3. 异步组件需要返回一个promise对象来确保 suspense 标签合理运行



两种返回方式，setup返回的内容是promise对象

1. [Vue-router@3.x](mailto:Vue-router@3.x) 与 [Vue-router@4.x](mailto:Vue-router@4.x)
   1. 重定向区别



* 1. 捕获路径



* 1. 获取当前路由  
     删除 router.getMatchedComponents, 从router.currentRoute.value.matched 获取
  2. push 或者 resolve 一个不存在的命名路由时，将会引发错误，而不是导航到根路由 "/" 并且不显示任何内容
  3. 获取路由信息  
     

1. 其他内容
   1. 执行顺序：vue3 会先执行setup中的方法，再执行兼容方法，data、computed、watch等，在setup中无法访问data
   2. Mount：2.x使用outerHTML 替换挂载元素，vue3.x 使用innerHTML 替换挂载元素的子元素
   3. 使用ref  
      
   4. for循环中使用 ref  
      
   5. Setup 返回普通对象  
      setup 返回普通对象，会跟reactive对象一样，具备响应式。
   6. Render 修改  
      