

什么是vue生命周期和生命周期钩子函数？

1、vue生命周期和生命周期钩子函数：

vue 的生命周期是： vue 实例从创建到销毁，也就是从开始创建、初始化数据、编译模板、挂载Dom→渲染、更新→渲染、卸载等一系列过程。

在这个过程中也会运行一些叫做生命周期钩子的函数，这给了用户在不同阶段添加自己的代码的机会。

vue生命周期钩子函数有哪些？

beforeCreate	Function	在实例创建之前被调用，此时实例还未完成初始化。在这之前，实例无法访问数据属性，无法使用 computed 和 watch 选项，且无法使用生命周期钩子。
created	Function	在实例创建完成之后被调用，在这一步，实例已完成属性初始化、编译模板、挂载 DOM、初始化 watcher 和子组件。但此时，还没有开始渲染，且无法访问组件的 data 属性。
beforeMount	Function	在挂载开始之前被调用，相关的 render 函数已经准备好，但 DOM 还没有被挂载。
mounted	Function	在实例被挂载之后被调用，完成 DOM 的挂载，在 mounted 钩子中，可以访问组件的 data 属性。
beforeUpdate	Function	数据更新之前被调用，此时数据已经更新，但还没有开始渲染。可以在这里进行一些数据操作。
updated	Function	由于数据更新，虚拟 DOM 重新渲染并挂载 DOM 之后被调用。你可以在 updated 中，对更改后的 DOM 进行操作。
activated	Function	keep-alive 组件激活时调用。
deactivated	Function	keep-alive 组件不活动时调用。
beforeDestroy	Function	实例销毁之前调用。在这一步，实例仍然完全可用。
destroyed	Function	Vue 实例销毁后调用。调用后，Vue 实例的所有数据属性都将被清除，记忆也丢失，不再存在了。
errorCaptured (2.5.0+ 新增)	(err: Error, vm: Component, info: string) => ? boolean	当捕获一个来自子组件的错误时调用。默认情况下，该钩子返回 false，表示该错误应该被处理。如果返回 true，则阻止错误向父组件传递。

注意：

1、mounted、updated不会承诺所有的子组件也都一起被挂载。如果你希望等到整个视图都渲染完毕，可以用vm.\$nextTick 替换掉mounted、updated：

```
updated: function () {  
  this.$nextTick(function () {  
    // Code that will run only after the  
    // entire view has been re-rendered  
  })  
}
```

2、http请求建议在 created 生命周期内发出