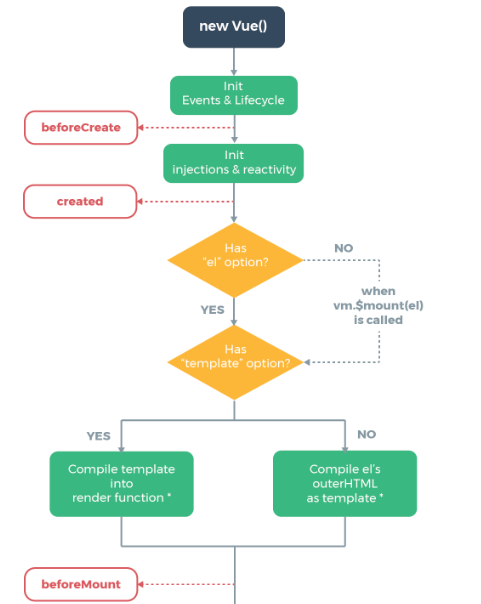
# Vue 生命周期

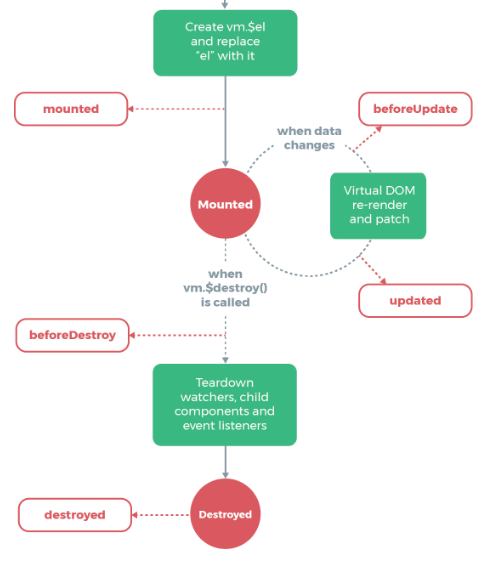
## 什么是生命周期

所谓“生命周期”，是指对象从构造函数开始执行（被创建）到被gc（Garbage Collection：垃圾回收）回收销毁的整个存在的时期。

“钩子”就是在某个阶段给你一个做某些处理的机会，其实这里也可以理解为回调函数，具体说明下，比如在组件刚刚挂载完成的时候我们想去完成一系列的操作，我们就可以在对应的生命周期的函数上去调用回调函数去实现操作，说直白一些就是在对象从创建到被回收的整个过程中会在不同的时期有不同的钩子函数，我们可以利用这个不同时期的钩子函数去完成不同的操作。

那么在vue框架体系中生命周期都有哪些呢，接下来我们看下vue的生命周期的图解：





从上面的图中可以很清楚的看到Vue的整个生命周期的过程，下面我们就来分析一下各个生命周期的具体含义：

|  |  |
| --- | --- |
| beforeCreate（创建前） | 组件实例刚被创建，组件属性计算之前，比如data属性等等 |
| created（创建后） | 组件实例刚创建完成，属性已经绑定，当时DOM还未生成，$le属性还不存在 |
| beforeMount(载入前) | 模板编译、挂载之前 |
| mounted（载入后） | 模板编译、挂载之后 |
| beforeUpdate（更新前） | 组件更新之前 |
| updated（更新后） | 组件更新之后 |
| beforeDestroy（销毁前） | 组件销毁前调用 |
| destroyed（销毁后） | 组件销毁后调用 |

每一个生命周期的具体含义都给大家做了一个介绍，下面我们对某几个特别的生命周期做一些简单的介绍，然后我们主要是结合我们项目当中使用比较频繁的生命周期做介绍：

下面我们针对生命周期的注意点做介绍：

beforeCreated：在使用这个生命周期时，el 和 data 并未初始化 ，在这个生命周期函数的时候，我们举例说明下，比如我们要在项目中加一些loading事件，这个时候可以在这个对应的生命周期上加上；

created: 在使用这个生命周期时，完成了 data 数据的初始化，但是el没有，

在这需要结束loading，可以在这里做一些初始化的操作，实现函数自执行等等

beforeMount：在使用这个生命周期时，完成了 el 和 data 初始化

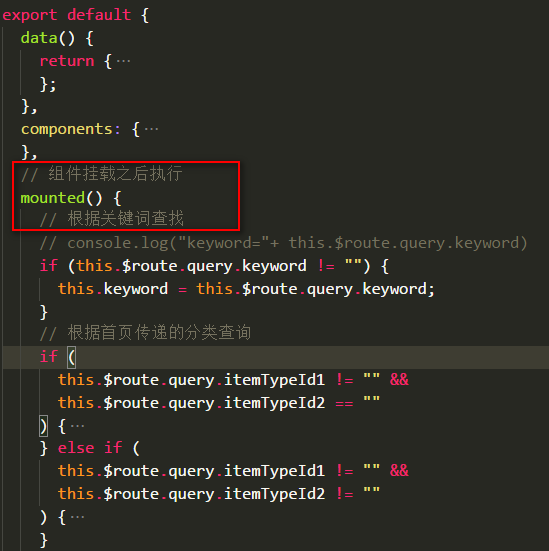
mounted ：完成挂载，这个生命周期用的是比较多的，之后我们会单独做介绍

## Mounted

这个生命周期函数用的是比较频繁的，基本上在我们项目中的每一个页面都会用到，具体的用法可以看下面的截图，生命周期是一个回调函数，然后函数内部书写需要在当前的Mounted的生命周期中完成的事情，这里我们来分析下，在mounted生命周期中，我们可以做的一系列操作：

1. 在这可以发起后端请求，拿回数据，然后通过一些列的操作去遍历循环数据
2. 在这块我们还可以来接收页面之间的传递的参数，比如我们由首页搜索内容，然后点击搜索之后会带着我们搜索的内容跳转到列表页面，那么问题来了，我们要在哪里接收传递过来的内容呢，我们就可以在mounted生命周期函数的里面来接收内容，并且去赋值给相应的变量去做进一步的操作；
3. 还有一种情况是，组件之间传值的时候，例如我们子组件需要传递给父组件一个参数，但是在父组件的哪里进行接收就是一个问题了，特定情况下，我们也是可以在这个位置进行接收子组件转递过来的数据的；

以上三种情况是我们mounted生命周期函数应用比较多的情况。



## Updated

这个Updated生命周期函数在项目中有的时候也会用到，这个生命周期函数代表当我们的组件更新的时候，我们想要做的操作，给大家举一个例子，比如我们项目中的地图组件，我们每一个剧院都有具体的位置，当我们点击具体剧院的位置的时候都会去加载不同的位置出来，当我们点击位置的时候，地图组件会更新其他的位置剧院的地图显示情况，现在这样的情况下就会用到updated生命周期函数，在我们点击不同剧院的时候，这个周期函数需要去接收不同剧院的经度和纬度，然后去加载显示剧院的位置，这就以具体的案例说明了我们这个updated生命周期的使用场景；



## 组件嵌套的生命周期

在使用vue进行开发时候，组件之间的嵌套是非常常见的一种情况，那么组件之间的生命周期顺序是怎么样的呢？是父组件先开始，然后子组件再开始呢还是其他的情况呢，这里需要给大家做一个明确的说明：

**父子组件的执行顺序为：**

- 父组件beforeCreated ->父组件created ->父组件beforeMounted ->子组件beforeCreated ->子组件created ->子组件beforeMounted ->**子组件mounted -> 父组件mounted。**

从上面的执行顺序上，可以看到有几个关键的点需要注意，当父组件生命周期执行到beforeMounted 的时候，下一步开始执行子组件的生命周期，等到子组件执行到mounted生命周期的时候，下一步重新回到父组件的生命周期mounted上，这里大家一定要注意一点，一定是子组件的mounted之后才会去执行父组件的mounted函数，我们理清楚这个关系对我们具体的开发项目有什么作用呢，举例给大家说明一下，比如我们父组件传递给子组件值的时候，如果我们在子组件的的mounted函数的时候去接收是接收不到 ，因为子组件的mounted在父组件mounted之前就已经完成了，所以是接收不到的，那么我们该在哪里或者说如何去接收父组件传递的值呢，之后的文档《vue计算属性和观察者》会做详细的介绍，这里大家先不用关心，只要知道造成接收不到的原因就可以了；

## 总结

生命周期的函数还是比较重要的，在项目开发中使用是比较多的，希望大家一定要去重点看下其中比较常用的几个生命周期函数，比如mounted、updated等，至于更为详细的介绍希望大家可以武看一下vue的官方文档。

生命周期文档地址：

<https://cn.vuejs.org/v2/guide/instance.html#%E5%AE%9E%E4%BE%8B%E7%94%9F%E5%91%BD%E5%91%A8%E6%9C%9F%E9%92%A9%E5%AD%90>