

2018 我所了解的 Vue 知识大全

2018-01-10 sunseekers 前端外刊评论

年初第一个 flag 就是掌握 vue ， 哈哈！

Vue ， React ， Angular 三大主流框架，最后我选择学习 Vue ， 接触过 React ， 自己感觉学习曲线有些陡峭，进而我选择了学习 Vue ， 他的学习曲线平稳很多；不管选择什么框架，除了对基础的知识有一定的了解掌握，还应该对 ES6 有一定的了解；

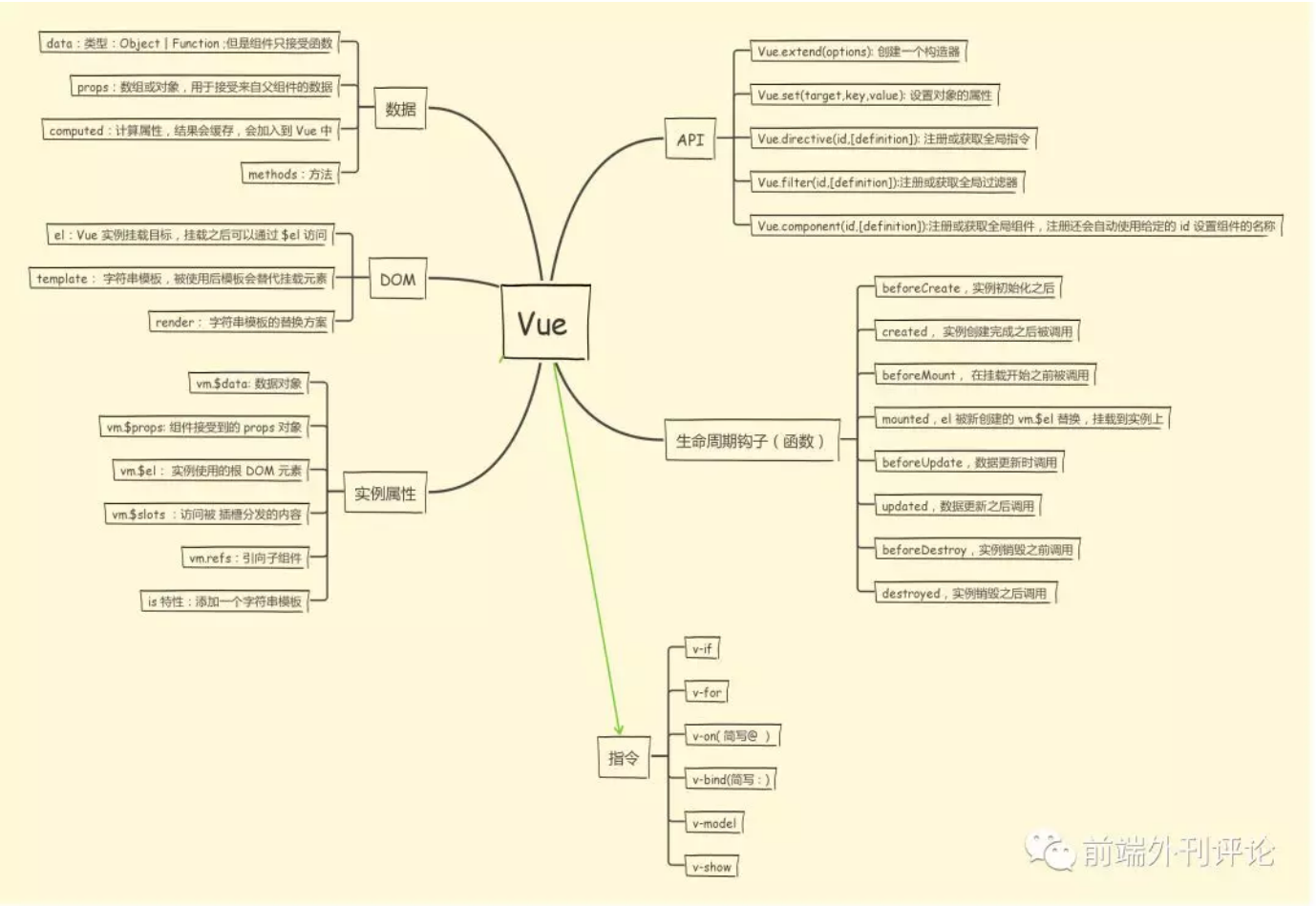


image.47cdd4e2ecaf.png

这就是我脑海中的 Vue 知识体系；

一句话概况了 Vue 通过尽可能简单的 API 实现响应的数据绑定和组合的视图组件。



真传X

免费公开课

BAT 大牛带你前端进阶

• 行业解读: 大前端时代 •

硬技能: Node实战 软技能: BAT面试

扫码获 3 场公开课免费听课地

前端外刊评论

广告

Vue 的创建

一个简单的 Vue 实例只需要四步即可



创建 Vue 简单的实例

```
<body>
  <div id="app">{{message}}</div>
</body>
<script type="text/javascript" src='js/vue.min.js'></script>
<script type="text/javascript">
  new Vue({
    el: '#app',
    data: {
      message: 'sunseekers'
    }
  })
</script>
```

前端外刊评论

image.ce4aae2af5fa.png

我们的学习目的肯定不止于创建简单的 Vue 实例；而是用它去实现更加多变的功能需求；那我们需要进一步去学习 Vue-router 和 Vuex 以及 Vue-cli；

先说说基础的 Vue 吧，在创建了一个简单 Vue 实例之后，我们是否还能在实例中添加更加完善的数据选项，去完成我们多变的功能需求；答案是肯定能的；

指令

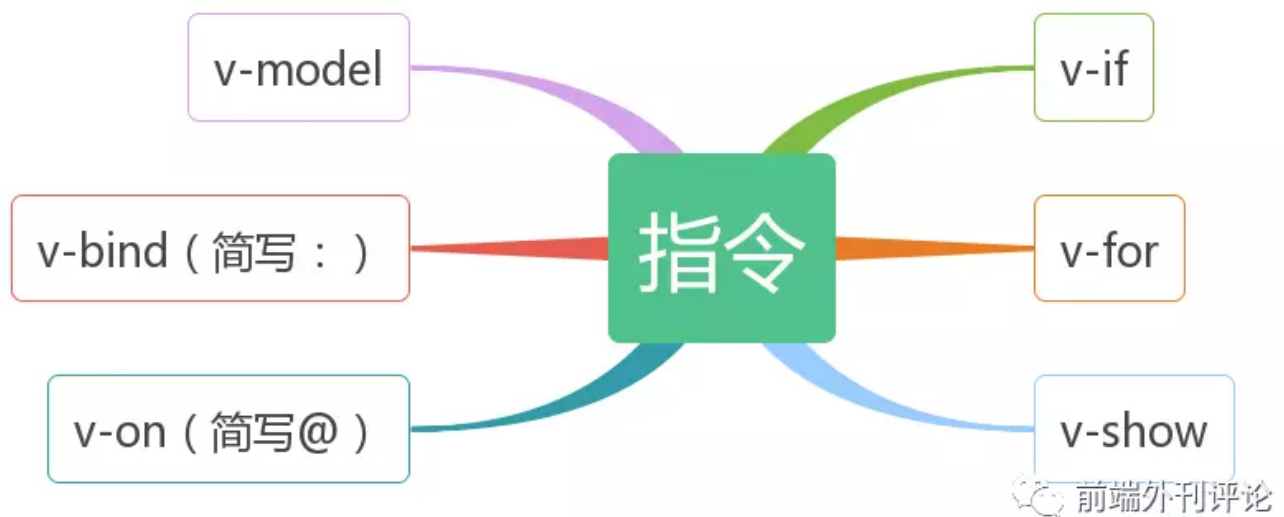


image.6c4c245ee76f.png

下面是我用上述指令写的一个小案例，刚好所有的指令都上了用场：

用我们学过的指令 `v-model` `@` `v-if` 和 `methods` 创建一个 Vue 实例；
在输入框内输入数据，按下回车键，自动添加到页面中，点击删除，删除该指令；动手试试吧



image.c3283963399a.png

详细请看 demo 源码

自定义指令

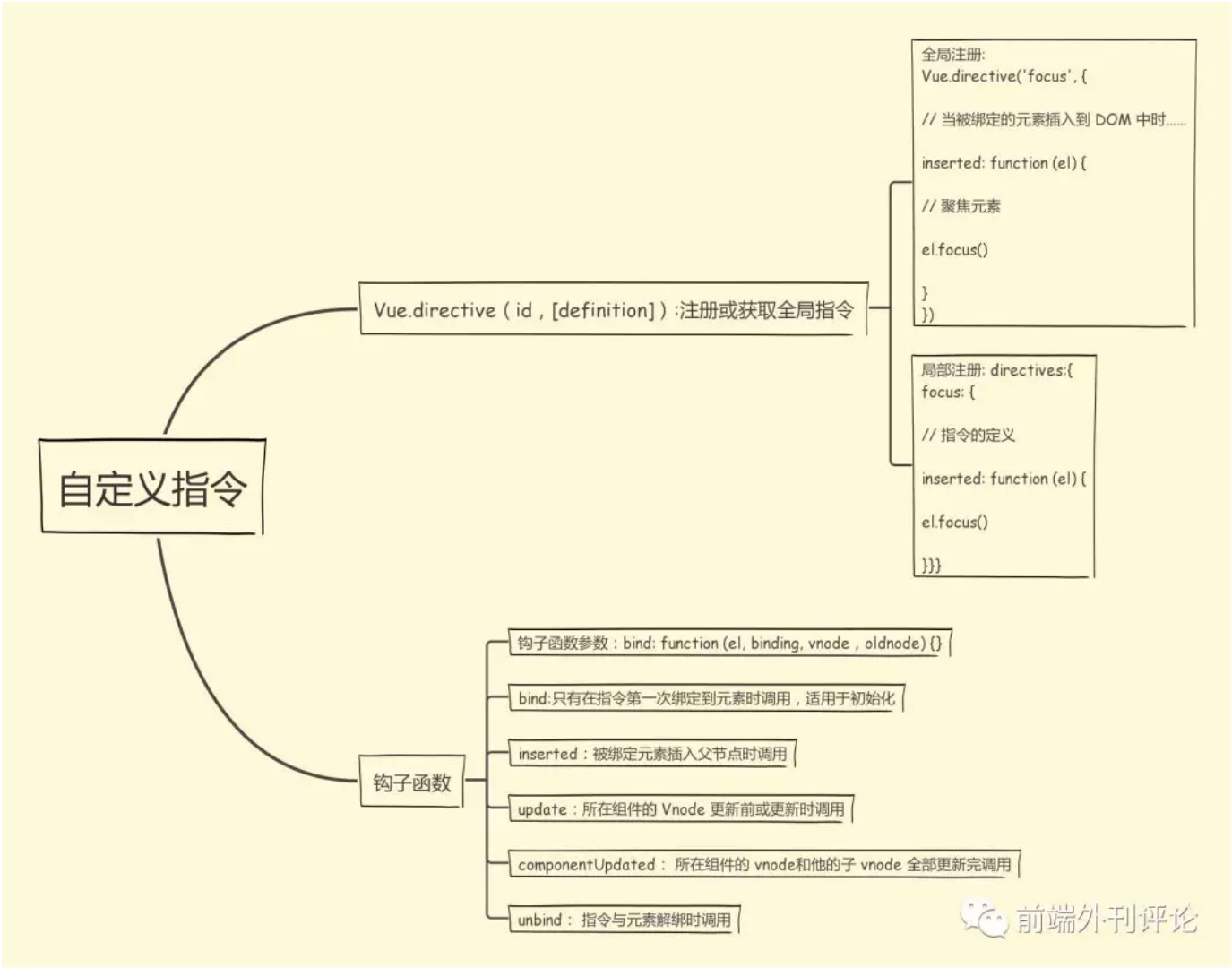


image.ba2d42f3c4b5.png

在代码的实例中：

```
<body>
  <p v-tack class='p'>
    I will now be tacked noto the page
    <input v-focus type='text'>
    <input v-border type='text'>
  </p>
</body>
<script type="text/javascript" src='js/vue.min.js'></script>
<script type="text/javascript">
  // 自定义全局指令
  Vue.directive('border',{
    bind(el){
      el.style.border='1px red solid'
    }
  })

  new Vue({
    el:'.p',
    //自定义局部指令
    directives:{
      tack:{
        inserted(el){
          el.style.color='red'
        }
      },
      focus: {
        // 指令的定义
        inserted: function (el) {
          el.focus()
        }
      }
    }
  })
})
```

截图(Alt + A)

 前端外刊评论

image.52778c3f29c4.png

[查看代码源](#)

Vue API

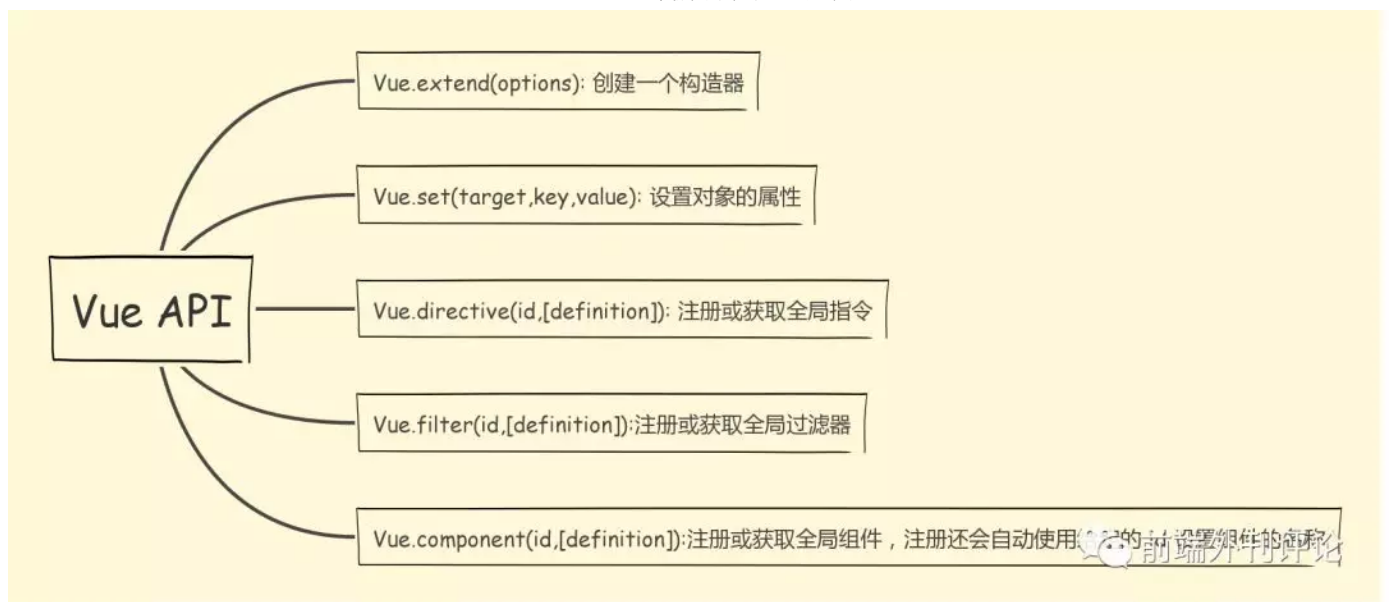


image.a2181e5c6527.png

Vue 常用选项

```
// Vue 实例常用选项
new Vue({
  // 1. 实例挂载目标
  el: '#app',
  // 2. 设置数据对象
  data: { ... }
  // 3. 计算属性
  computed: { ... }
  // 4. methods 方法
  methods: { ... }
  // 5. watch 观察
  watch: { ... }
  // 6. 生命钩子函数
  mounted: { ... }
  // 7. 自定义局部指令
  directives: { ... }
  // 8. 局部注册组件
  components: { ... }
  // 9. 过滤器
  filters: { ... }
})
```

image.3457b0191ecd.png

这就是我在学习 Vue 的时候，在 Vue 实例中添加的较多的；directives 和 components 在 Vue 实例中为创建的局部自定义指令和注册组件，Vue.directive() 和 Vue.component() 则是注册全局

Vue 组件

```
//注册全局组件常用选项
Vue.component('myarticl',{
  props:[],
  template:'',
  render:function(){},
  data:function(){},
  computed:{},
  //组件里面插入子组件
  components:{},
  methods:{},
  watch:{},
  mounted:{},
  filters:{}
})
```


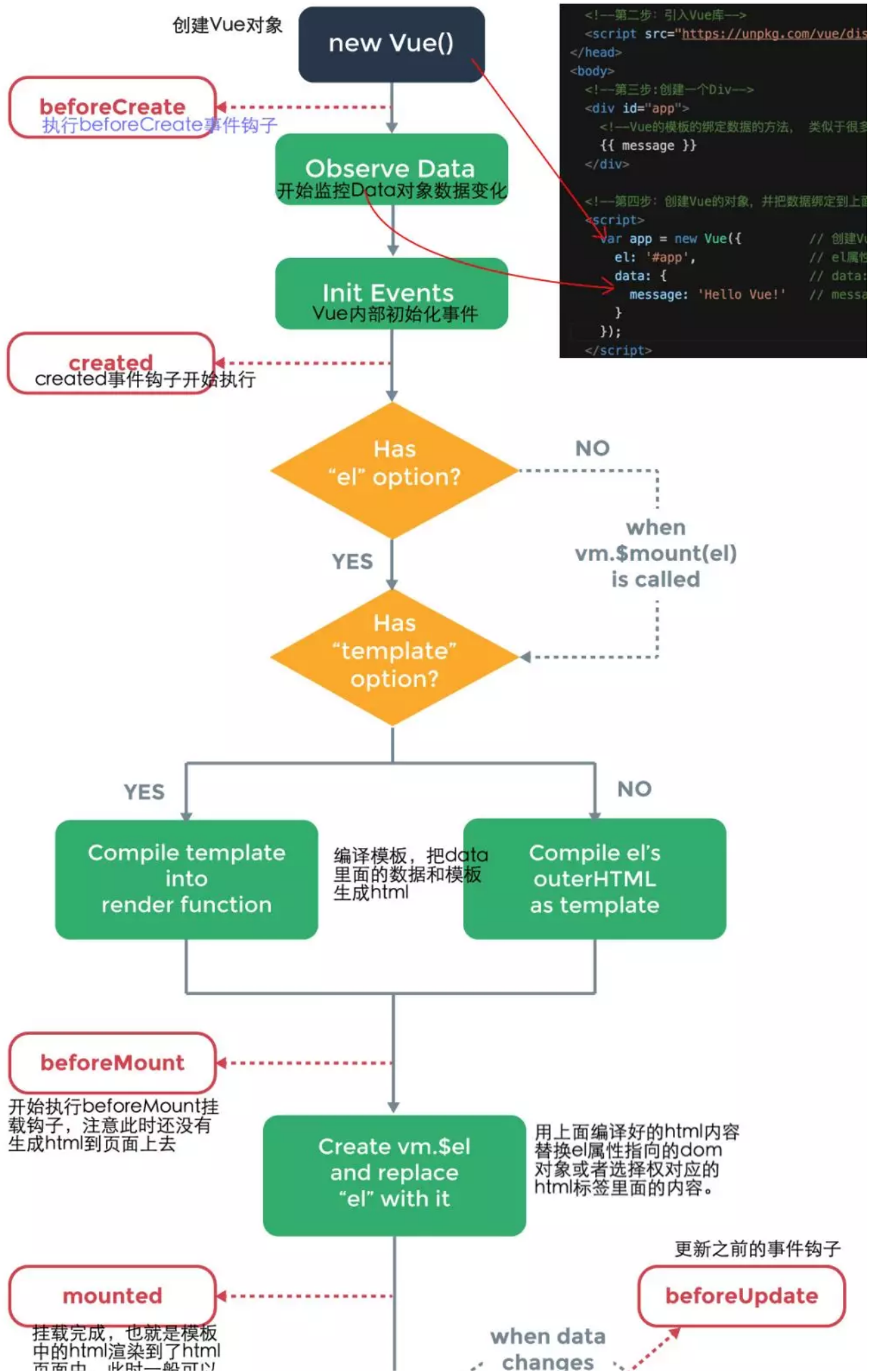


image.c5a965b84674.png

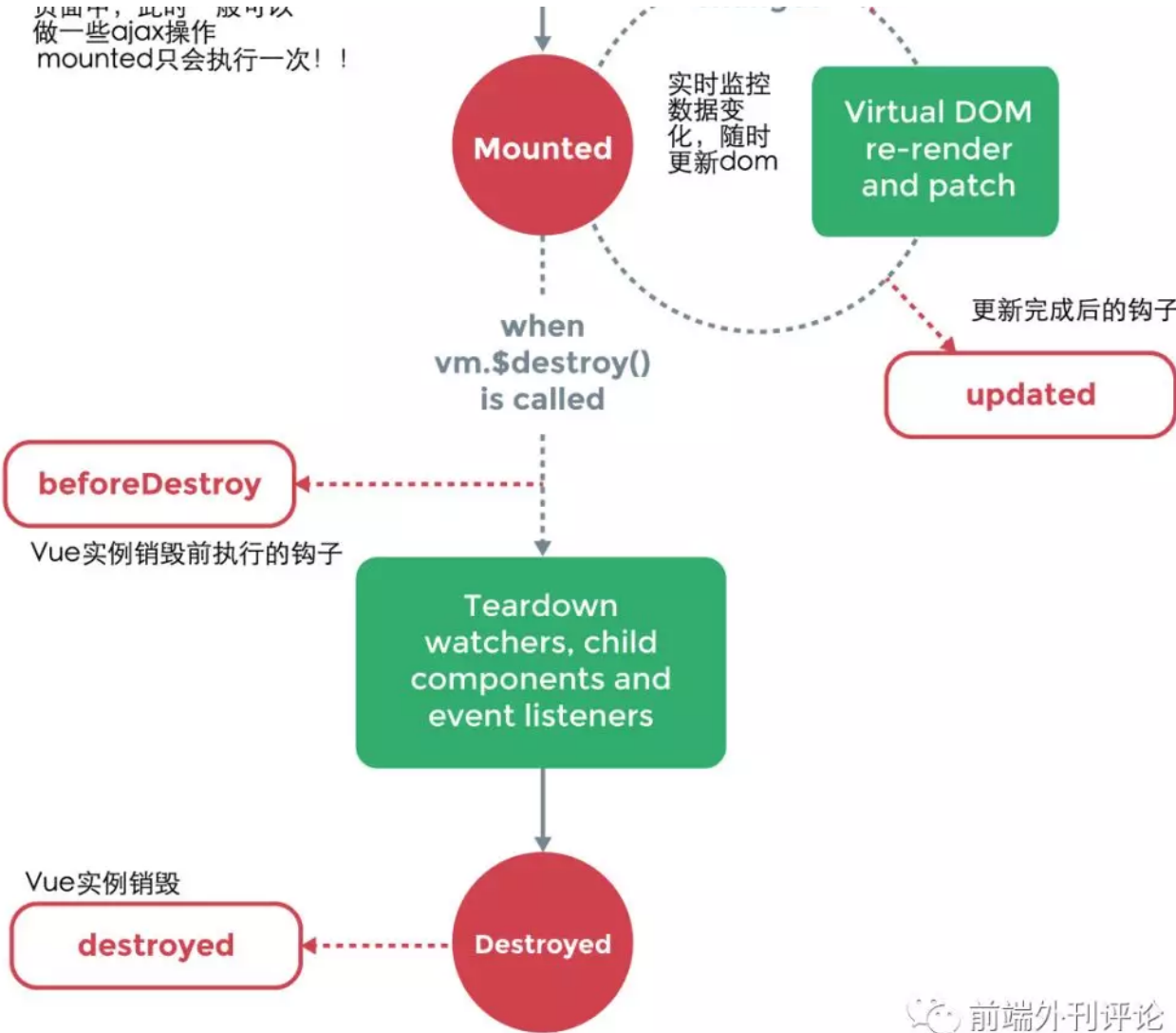
生命周期钩子

每个 Vue 实例在被创建之前都要经过一系列的初始化过程。例如需要设置数据监听、编译模板、挂载实例到 DOM、在数据变化时更新 DOM 等。同时在这个过程中也会运行一些叫做生命周期钩子的函数，给予用户机会在一些特定的场景下添加他们自己的代码。

在 QQ 群里看到一张生命周期图，我觉得写的特别好,感谢总结这张图的朋友



页面中，此时，就可以做一些ajax操作
mounted只会执行一次！！



前端外刊评论
Created by Paint X

image.b6b05c1a1b16.png

配合着实例代码效果更佳查看代码源

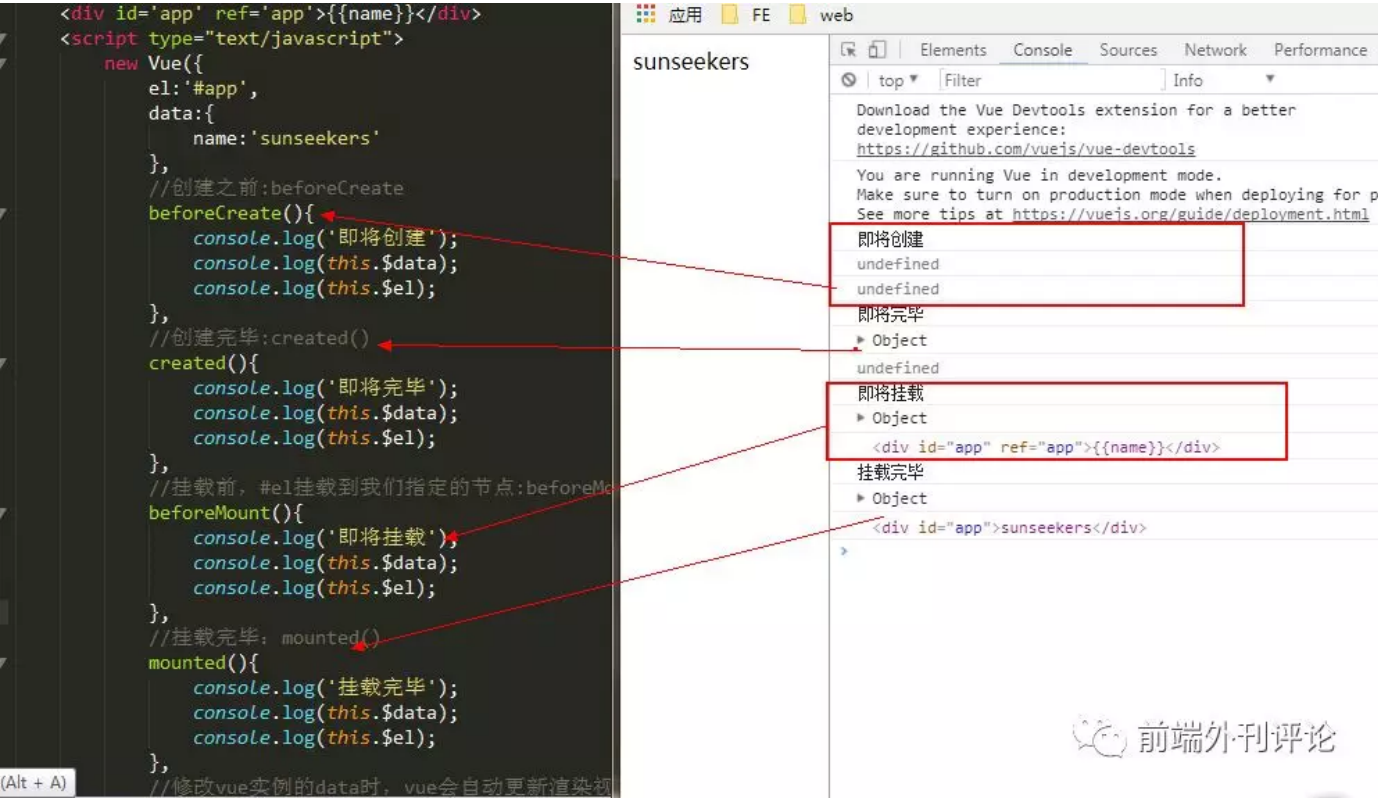


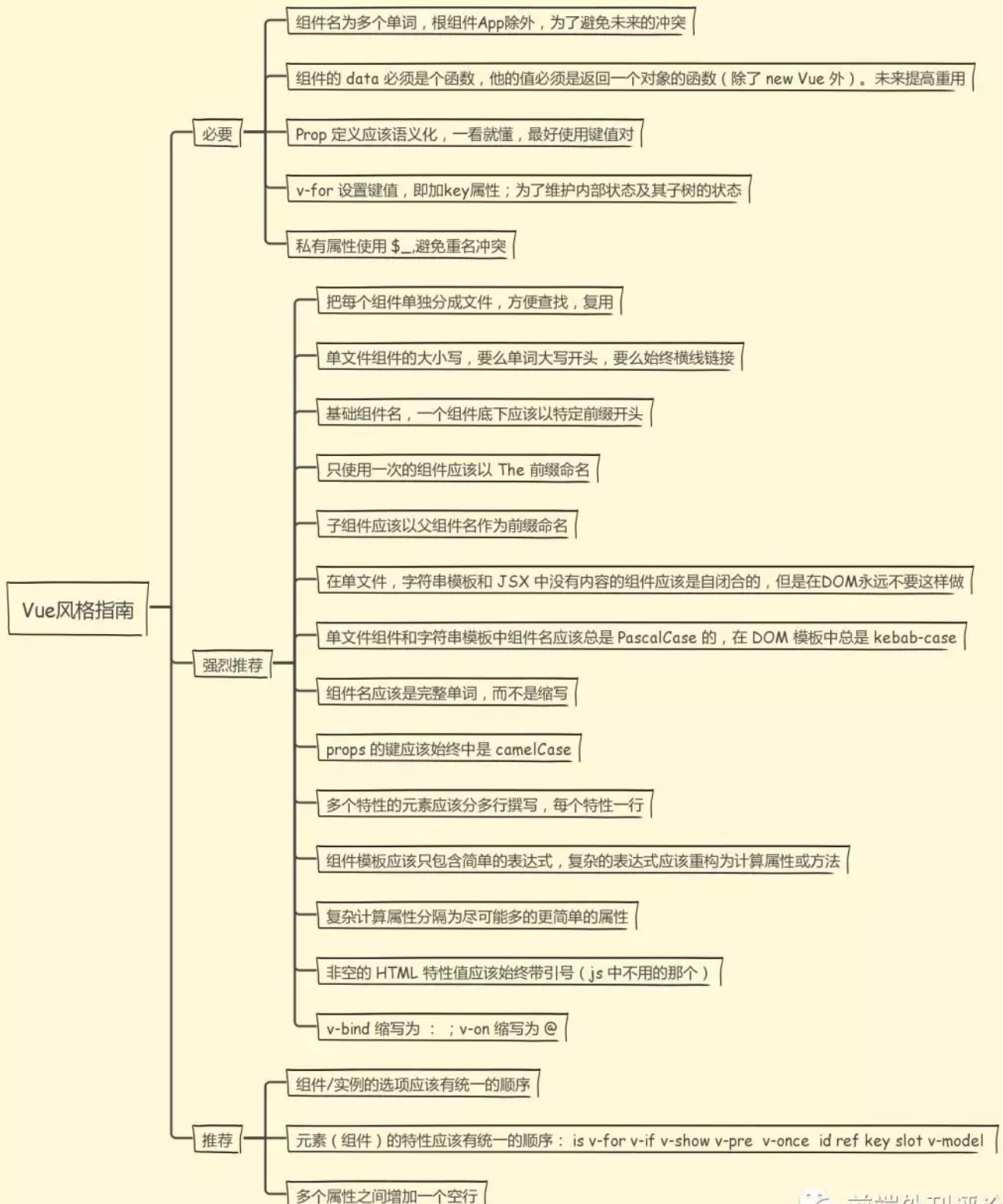
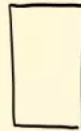
image.3ea6b2930269.png

生命周期钩子的函数简单说就是八个函数



image.3b51cb21948b.png

如何让 Vue 书写更佳优美?



详细 style-guide 请参看官网 进一步学习请参考官方文档



部分资源来自网络

[阅读原文](#)