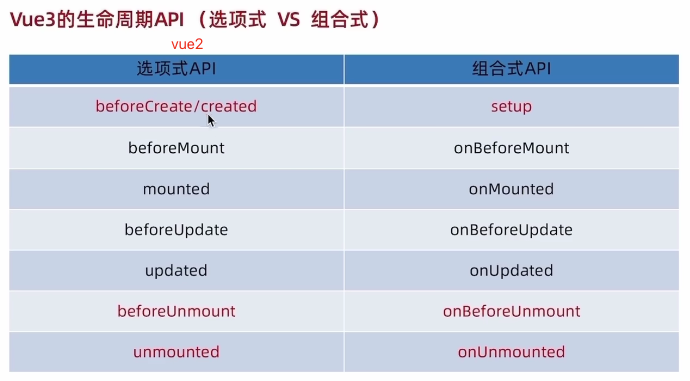
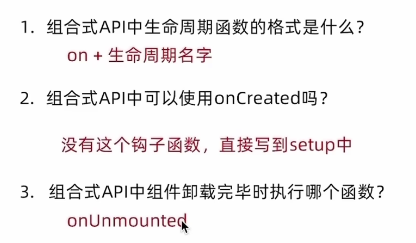
# 生命后期函数







# 父子通信

## 2.1父传子

1. <template>  
    <div>  
    父组件传入的数据---{{ ***message*** }}  
    </div>  
   </template>  
     
   <script setup>  
   import {***onMounted***} from 'vue'  
     
   const ***Twqprops*** = ***defineProps***({  
    message:***String***})  
     
   const btnClick = () =>{  
    //在script中使用父组件传过来的值，必须通过自定义变量Twqprops来调用，  
    // 而不能通过this.message来调用  
    ***console***.log(***Twqprops***.message);  
   }  
     
   ***onMounted***(()=>{  
    btnClick()  
   })  
   // export default {  
   // name: "text",  
   //  
   //  
   // }  
   </script>

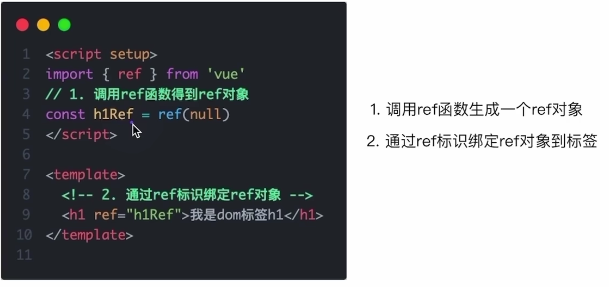
## 2.2子传父



const ***emit*** = ***defineEmits***(['get-message'])//这里就相当于是this.$emit（）

传入数组的原因是，以后可能会定义多个事件，这里只有一个事件，因此数组里也只有一项

# 3.模版引用



//调用ref函数 -> ref对象  
const ***h1Ref*** = ref(null)  
const ***comRef*** = ref(null)  
  
//组件挂载完毕之后才能获取  
***onMounted***(()=>{  
 ***console***.log(***h1Ref***.value);  
 ***console***.log(***comRef***.value);  
})

<h1 ref="h1Ref">我是父组件的h1标题</h1>  
<texTa ref="comRef"></texTa>

## defineExpose()

vue3的一个明显变化，通过ref拿到的元素并不包含元素内部的属性和方法，vue2中是包含的

默认情况下在<script setup>语法糖下组件内部的属性和方法是不开放给父组件访问的，可以通过defineExpose编译宏指定哪些属性和

方法允许访问



# 4.provide和inject