

注意:组件通信在你们将来面试的时候出现频率超级高。、

前面基础的时候：组件通信讲了6种？

组件通信方式1: props

使用场景:[父子通信]

传递数据类型:

1:可能是函数 -----实质子组件想给父亲传递数据

2:可能不是函数-----实质就是父亲给予子组件传递数据

特殊情况：路由传递props

1:布尔值类型，把路由中params参数映射为组件props数据

2:对象，静态数据，很少用

3:函数，可以把路由中params|query参数映射为组件props数据

组件通信方式2:自定义事件 \$emit \$on[简写@]

事件:原生DOM事件---- 【click|mouseenter.....】

事件：自定义事件-----[子给父传递数据]

组件通信方式3: \$bus 全局事件总线---- 【万能】

组件实例的原型的原型指向的Vue.prototype

组件通信方式4: pubsub-js 【发布订阅消息】*在vue中根本不用【React】 ----万能

组件通信方式5:Vuex[仓库] -----数据非持久化----万能的

核心概念：5

state

mutations

actions

getters

modules

组件通信方式6: 插槽-----父子通信 【结构】

slot

默认插槽

具名插槽

作用域插槽:子组件的数据来源于父组件，但是子组件的自己的结构有父亲决定。

watch|computed|method区别?

```
{  
  name:'王二麻子',  
  eat:function(){  
  
  }  
  
}
```

1:事件相关的深入学习

事件:事件已经学习过两种, 第一种原生DOM事件, 第二种自定义事件。

<Event1 @click="handler1">

组件绑定原生DOM事件, 并非原生DOM事件, 而是所谓的自定义事件。

如果你想把自定义事件变为原生DOM事件, 需要加上修饰符.native修饰

这个修饰符, 可以把自定义事件【名字: 原生DOM类型的】变为原生DOM事件,

7:v-model实现组件通信?

v-model: 指令, 可以收集表单数据【text、radio、checkbox、range】等等

切记: v-model收集checkbox需要用数组收集

v-model:实现原理 :value @input 还可以实现父子数据同步。

8:属性修饰符.sync, 可以实现父子数据同步。

以后在elementUI组件中出现, 实现父子数据同步。

9: \$attrs与\$listeners ----vue-helper 父子组件通信

\$attrs: 组件实例的属性, 可以获取到父亲传递的props数据 (前提子组件没有通过props接受)

\$listeners: 组件实例的属性, 可以获取到父亲传递自定义事件 (对象形式呈现)

10:\$children与\$parent 可以实现父子组件通信

ref:可以在父组件内部获取子组件---实现父子通信

\$children:可以在父组件内部获取全部的子组件【返回数组】

\$parent:可以在子组件内部获取唯一的父组件【返回组件实例】

第一种:props ----父子

1:父子通信, props只读的

2: 写法三种【todos】 {todos: Array} {todos: {type:Array,default:[]}}

3: 路由props

第二种:自定义事件

1: 子给父传递数据

2: 两种【简单写法|复杂写法】

<Todos @erha="handler">

```
mounted(){
  this.$refs.erha.$on('xx',callBack)
}
```

3:全局事件总线 ---万能

4: pubsub-js-----万能

5: vuex-----万能

6: v-model

7: .sync

8:\$attrs|\$listeners

9:\$children|\$parent ----ref

10插槽 -----作用于插槽

插槽：父子组件通信（HTML结构）

插槽slot：

默认插槽|具名插槽|作用于插槽

作用域插槽:子组件的数据来源于父组件，但是子组件决定不了自身结构与外观。