管理员的账号信息

13700000000

111111

学习过的组件库：

React（Vue）：antd[PC] antd-mobile[移动端]

Vue: ElementUI[PC] vant[移动端]

P100讲的是如何引入element-UI 包括按需引入。

插件qrCode可以生成二维码

分页器是个重难点，搞清楚分页器还有分页器对应的接口以及点击分页器是怎么调用数据的这个问题，很多当前做的项目的技术难点迎刃而解。

图片懒加载

<https://www.npmjs.com/package/vue-lazyload>

vee-validate

项目打包后，代码都是经过压缩加密的，如果运行时报错，输出的错误信息无法准确得知是哪里的代码报错。有了map就可以像未加密的代码一样，准确的输出是哪一行有错。

Vue.config.js 配置

productionSourceMap:false

组件的通信方式：6种

第一种：props

父子间通信

父给子传递函数，本质是子给父传递数据

父给子传递非函数，本质是父给子传递数据

第二种：自定义事件

子给父传递数据

使用$on 和$emit

第三种：全局事件总线$bus

适用范围：万能

Vue.prototype.$bus = this;

第四种：pubsub.js，在react中用的比较多一点。（发布和订阅）

适用范围：万能

第五种：Vuex（store）

适用场景：万能

第六种：插槽

适用场景：父子组件通信---结构

分类：默认插槽、具名插槽、作用插槽

**事件的注意事项：**

事件的分类：系统事件+自定义事件

系统事件，给原生的DOM绑定，比如button，绑定click事件

自定义事件，给组件标签绑定（默认是自定义事件，但是.native可以把自定义事件变成原生DOM事件，原理是利用事件的委派，给根节点加事件）

自定义事件的使用：

父组件，组件标签里面写自定义事件和自定义事件的回调，回调会传event参数，还可以接收子组件给父组件传递的数据。

子组件，$emit(‘事件名’,要传递的参数)，也就是子组件中触发自定义事件。

**v-model的高级使用**

v-model是双向数据绑定

:value是单向数据绑定

@input 底层是oninput事件，可以结合表单元素一起使用，当表单元素文本内容发生变化的时候会发出一次回调

**V-model可以实现父子组件数据同步**

父组件

如果给自定义组件绑

:value是props传参，@input是自定义事件

<CustomInput :value=”msg” @input=”msg= $event”/>

<CustomInput v-model=”msg”/>

子组件

如果给原生DOM绑定后

:value是单向数据绑定，@input是原生的oninput事件

<input type=”text” :value=”value” @input=” $emit(‘input’,$event.target.value) ”>

**属性修饰符sync(组件通信方式的一种)**

可以实现父子组件数据的同步。

:money.sync，代表父组件给字符串传递props [money]，给当前子组件绑定一个自定义事件（update:money）

**$attrs和$listener（组件通信方式的一种）**

可以用来对el-button进行二次封装

父组件

前面的type icon title 是传递的props参数。@click是父组件给子组件绑定的自定义事件。

<HindButton type=”success” icon=”el-icon-delete” title=”提示按钮” @click=”alert(‘666’)”/>

子组件

接收父组件给的数据

Props:[‘title’]

或者直接用$attrs

$attrs属于组件的一个属性，可以获取父组件的props数据

但是需要注意，如果子组件通过props接收了属性，在attrs属性当中是获取不到的。

数据的使用

//利用a标签的提示功能实现按钮的提示功能

<a :title=”title”>

//简写形式，给子组件标签绑定的所有属性都会传递到这里

<el-button v-bind=”$attrs” v-on=”$listeners”></el-button>

</a>

$listeners ，是组件实例自身的一个属性，可以获取父组件给子组件传递的自定义事件。

**$children与$parent（可以实现父子组件的通信）**

ref可以获取某一个子组件

$children组件实例的属性，可以获取到当前组件的全部子组件【数组】

$parent组件实例的属性，可以获取到当前子组件的父组件，进而可以操作父组件的数据和方法。

**混入mixin**

如果项目当中出现很多结构类似的地方，要想到组件复用。

如果项目当中的组件JS业务逻辑相似，要想到混入mixin。【可以把多个组件JS部分重复】

**插槽**

子组件的数据来源于父组件（props）

子组件使用slot留个插槽，把数据回传给父组件，由父组件决定子组件的样式。

<template slot-scope=”todo”>

<span :style=”{color:red;}”>{{todo.text}}</span>

</template>