**Vue 插槽slot / slot-scope的使用**

[圆脸黑猫警长](https://www.jianshu.com/u/51cd12367e21)关注

0.3872020.07.14 12:23:45字数 592阅读 1,511

**一、slot 插槽的使用**

**1.功能：**

通过slot可以对组件进行扩展。

**2.使用场景：**

对组件可能的不同显示部分，由外部进行 内容 的指定，起到拓展的作用。

**3.基本使用方法：**

1. 在组件中使用slot标签  
   <slot></slot>写在需要被替代的部分

2.在组件的使用时，在内部添加需要替换的东西即可<A>XXX</A>

// 定义

<myDiv>

A

<slot></slot>

B

</myDiv>

//使用

<template>

<myDiv>AA</myDiv>

</template>

效果即为在A、B之间插入AA的显示

3.显示默认内容的slot  
在定义的时候加上默认显示的内容

// 定义

<myDiv>

<slot> 定义默认内容 </slot>

</myDiv>

//使用

<template>

<myDiv></myDiv>

<myDiv>AA</myDiv>

</template>

效果上面的是显示定义默认内容，下面的显示AA。slot是全部替换，并不会对原来slot中的内容进行merge。

**4.进阶使用：**

一个组件中可以添加多个slot， 如果直接使用，这外部传入的会把slot全部都替换。为了进行区分，则需要对slot设置name属性，使用时用slot属性进行区分

// 定义

<myDiv>

A

<slot name='top'></slot>

B

<slot name='middle'></slot>

C

<slot name='bottom'></slot>

</myDiv>

//使用

<template>

<myDiv>

<div slot='middle'>mmm</div>

</myDiv>

</template>

此时的显示只替换中间的slot

**二、slot-scope 作用域插槽的使用**

**1.功能：**

组件进行扩展。

**2.使用场景：**

对组件可能的不同显示部分，由外部进行样式 的指定，起到拓展的作用。但数据仍然来源于子组件内部。

**3.基本使用方法：**

1.先声明一个slot ，并自定义一个属性（名字任意）绑定这个子组件的数据

<slot :customname="mydata">

<div>我是子组件：{{mydata}}}</div>

</slot>

这里用 customname绑定子组件数据mydata，customname可以任意命名，后面会用到

2.父组件中通过slot-scope获取slot引用

<mycomponent>

<template slot-scope="usename">

<div>{{usename.customname}}</div>

</template>

</mycomponent>

这里 usename可以随便命名，表示对子元素slot的引用变量名称，customname即是上一步中定义的。在里面即可使用usename.customname获取子组件中的数据了，然后在父组件中去写样式等，对子组件的slot进行替换。  
注意点：  
template这里是Vue 2.5.X以下的版本必须的语法格式，以上的可以为任意的标签了如div等

完整代码如下：

<!doctype html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport"

content="width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0">

<meta http-equiv="X-UA-Compatible" content="ie=edge">

<script src="https://unpkg.com/vue"></script>

</head>

<template id="abc">

<div>

AA

<slot :customname="mydata">

<div>{{mydata}}}</div>

</slot>

BB

</div>

</template>

<body>

<div id="app">

<mycomponent>

<template slot-scope="usename">

<div>{{usename.customname}}</div>

</template>

</mycomponent>

</div>

</body>

<script>

const app = new Vue({

el:"#app",

data:{

msg:'m1',

},

components:{

mycomponent: {

template: '#abc',

data(){

return{

mydata:['a1','a2','a3']

}

}

},

},

})

</script>

</html>

显示效果如下：

网页显示.png

和普通的slot作用一样，都是替换子组件中的slot的全部内容