vue

// 我是子分支 style

/\*\* class \*\*/

// (一) 添加一个class || 一个变量控制,是否添加

v-bind:class = "{ class1 : selectStatus}" // 动态的加一个class，条件放在后面，就是通过变量 true false 控制class 的存在

data:{

selectStatus:true

}

< class="class1" >

// (二) 添加多个class || 同时添加多个class

v-bind:class = "[class1Name,class2Name]"

data:{

class1Name:'class1',

class2Name:'class2'

}

<class="class1 class2">

//(三) 切换两个class || 根据状态添加 ，不同的class

v-bind:class = "[ status == 1 ? 'class1' :'class2']" // 动态的切换两个class，三元表达式 ，提交成立 给不同的值，不是对象，是因为没有，赋值操作

data:{

status:1

}

<class="class1">

/\*\* style \*\*/

// (一) 赋值,可以是不同的值

v-bind:style="{ 'width' : widthNum }"

data:{

widthNum:70px;

}

<style="width:70px;">

//(一) 赋值，可以是对象

v-bind:style ="styleObj"

data:{

styleObj:{

width:170px;

height:90px;

}

}

<style="width:170px; height:170px;">

// (一)给个数组 赋多个变量的值

v-bind:style="[basestyle,overStyle]"

// (一) 赋多个值

v-bind:style="{display:['-webkit-box','-ms-flexbox','flex']}"

// (二) 赋值 || 根据条件，赋不同的值

v-bind:style='{ "width" : status == 1 ? "120px" : "40px" }' //动态给 属性赋不同的值，赋值放前面， 后面判断，所有是对象的格式 ===赋值

data:{

status:1

}

<style="width:120px">

/\*\* 控制显示和隐藏\*\*/

v-if="isShow"

data:{

isShow:true

}

a b c d e f g h i j k l m n o p q r s t u v w s y z

a b c d e f g h i j k l m n o p q r s t u v w s y z

<template v-if="loginType === 'username' ">

<lable>Username</lable>

<input placeholder = "Enter your username"/>

</template>

template template template

template template template

<template v-else>

<lable>Email</lable>

<input placeholder="Enter email address"></input>

</template>

<template v-if="loginType === 'username' ">

<lable>Username</lable>

<input placeholder = " Enter your username " key="username-input"/>

</template>

<template>

<lable> Email </lable>

<input placeholder=" Enter your email address" key =" email-input"></input>

</template>

v-show 的元素始终会被渲染在DOM中

v-show 只是简单的切换 CSS属性 display 不显示到界面上，但是在Dom中,displace :none; (visibility = hidden) 隐藏 但占空间

v-show 不支持 template

/\*\* 事件 \*\*/

v-on:click = "clickFun"

v-on:blur = "clickFun" // 失去焦点 blur blur

v-on:focus = "clickFun" // 获得焦点 focus focus

visibility visibility visibility visibility

Network 网络

console 控制台

Element 元素

application 应用

performance 性能

Sources 资源

Memory 存储

Audits 审计

security 安全

打开抽屉式 控制台 esc

清除控制台 ctrl +l

console.log();

console.error();

console.error();

// 断言,仅仅只当它的第一个参数为false时，才显示错误的字符串（第二个参数）

console.assert(list.childNodes.length <500 ,"Node count is >500")

分组打印

console.log("Authentication phase")

console.log('1');

console.log('2')

console.groupEnd();

//

<template></template>

export default{ // 对象

name:'index',

components:{}, components .. 组件 compontents compontants components components compotents

data(){ // 数据

return{

}

},

methods:{

},

created(){

},

}