

15. 组件的注册

学习要点：

1. 规则说明
2. 全局局部

本节课我们来开始学习 **Vue3.x** 的组件注册的基本规则以及全局和局部的注册。

一. 规则说明

1. 上一节课，我们简单的进行了组件的注册，了解了如何创建注册和复用方法；
2. 那么，我们探讨一下组件名称的一些规则，具体如下：
 - (1) . 组件名称是唯一的；
 - (2) . 命名规则可以采用 **kebab-case(button-counter)**，减号隔开；
 - (3) . 命名规则也可以用 **PascalCase(ButtonCounter)**，单词首字母大写；
 - (4) . 这里推荐使用 **kebab-case**，全小写并用减号隔开，遵循 W3C，防止冲突；

PS：这里防止冲突是，防止与未来的 HTML 扩展元素标签的冲突；

二. 全局和局部

1. 上一节课，我们定义了一个全局组件，用连缀方法再写一遍，如下：

```
// 定义一个全局组件
Vue.createApp({}).component('button-counter', {}).mount('#app')
```

2. 有全局注册，就有局部注册的方法，局部注册的方法如下：

```
// 可以理解为根组件，最顶层的组件
const app = Vue.createApp({
  data() {
    return {}
  },

  // 定义局部组件(也就是这个根组件的子组件)
  components : {
    'button-counter' : {
      // 数据
      data() {
        return {
          counter : 100
        }
      },

      // 模板
      template : `
        计算器：{{counter}}`
    }
  }
})
```

```
        <button v-on:click="counter++">counter++</button>
        <br>
      },
      'button-counter2' : {},
      'button-counter3' : {},
    }
  })
```

3. 当然，我们可以把组件的参数二分离出来，让代码更清晰且更好管理；

// 定义一个局部组件的对象内容部分

```
const buttonCounter = {}
```

```
components : {
  'button-counter' : buttonCounter,
}
```

PS：在调用上，并没有差异，而我们推荐使用局部组件，因为在以后的项目构建上更具优势；