Vue基础及ES6基础语法

## MVVM模式介绍

Vue是一款轻量级的渐进式前端框架，主要功能:

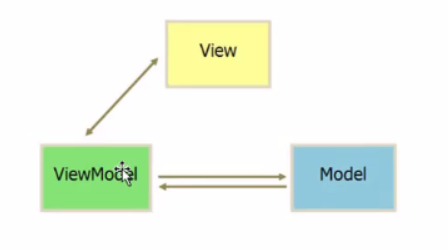
1. **模板渲染**/**数据同步**
2. **组件化**、**模块化**
3. 扩展功能: **路由** 等等

**MVVM**:

M:Model数据模型 负责数据存储

V:View 视图 负责页面的显示

VM : View Model负责业务处理。对数据进行加工，之后交给视图。



流程概述:

当我们的Model(模型)发生变化时会通知ViewModel(视图模型)然后更改我们的页面，当我们的页面数据模型更改时会通知ViewModel然后更改我们的数据(即双向绑定).

## Vue常见指令

1. 插值表达式

{{}}

当模型中的数据发生改变时，那么视图中的数据也对应发生改变。

1. v-text

将一个变量渲染到指定元素中

<h1 v-text=”name”></h1>

1. v-html

可以输出html元素

1. v-model

实现双向数据绑定

1. v-bind

绑定页面中元素的属性

1. v-if和v-show

v-if=”判断表达式”

作用:判断是否加载固定的内容，如果是真就加载，否则不加载

v-show=“判断表达式”

作用:判断是否显示内容

v-if有更高的切换消耗，安全性能高。v-show初始化消耗大一点，**所以如果需要频繁切换对安全性要求时，可以使用v-show**.**如果在运行时条件不能改变的话。使用v-if更好一点**。

1. v-for

作用: 控制html元素中的循环

语法: v-for=”item in 集合”



1. v-on

作用:对页面中事件进行绑定

语法:v-on:事件类型=”监听器”

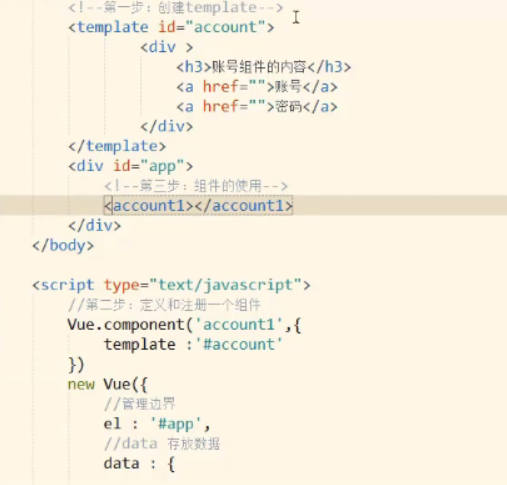
缩写: @事件类型=”监听器”



## 组件

组件是Vue.js中一个非常强大的功能，可以扩展HTML元素，封装可重用的代码。

1. 将组件内容定义到template模板中
2. 组件实现指定以及事件绑定



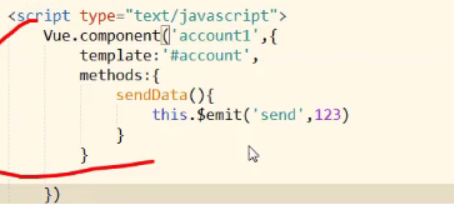
## 组件传参

父传子



子传父







## Vue-router路由基本使用

在一个系统中，由多个页面组成，通常会使用Vue中的组件来实现。那么当从一个页面跳到另一个页面中时，通过url路径来实现的那个url对应那个页面在Vue中是通过vue-router来实现的。





## 路由传递参数





# ECMAScript简介



## Let使用

Let是声明变量的关键字

1. 在相同的作用域内，let不能重复声明一个变量
2. Let声明的变量不会被预解析
3. 暂时性死区(变量在let声明前都不能访问，为了防止先调用后声明这个现象)



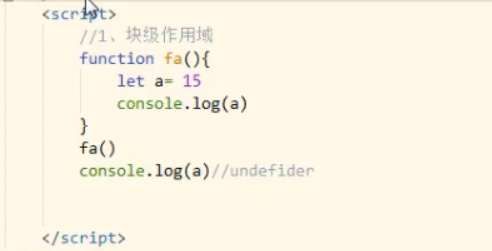


## Let与For的使用

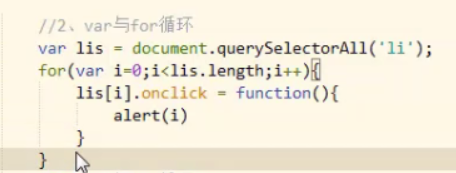
1. **块级作用域**

Let声明变量具有块级作用域，块级作用域时一对大括号

块级作用域可以直接写一对大括号，以后就不用写自执行函数了



1. Var与for循环



假如有3个li不管点击那个i都是4

1. Let与for循环



点击结果0 1 2 3

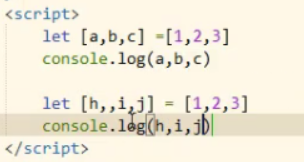
**Const用法**

Const声明一个常量，一旦声明后就不能修改了

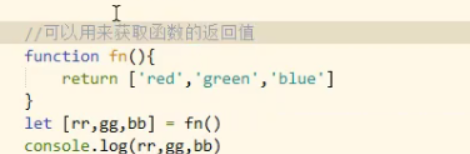
1. 如果声明后再修改的话就会报错
2. 只声明不赋值也会报错
3. 只能先声明后使用，不会被提前解析
4. 不能重复声明一个常量

注意:const声明的对象中的属性是可以修改的。

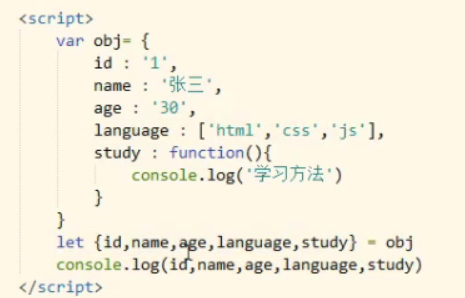
**数组对象赋值**



a=1 b=2 c=3，值要一一对应否则undefined



**对象解构赋值**

****

注意:如果变量之前已经被声明了，再次赋值的时候不能加let

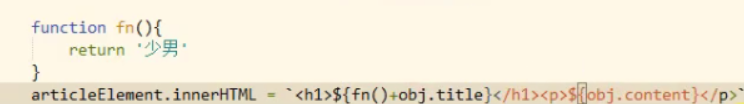
并且整个赋值语句要用小括号包裹，以防止程序识别成作用域。

****

**字符串模板拼接**

1. 字符串需要用一对反引号包括起来，它可以定义多行字符串，只用一对反引号
2. 要拼接的数据需要放在${}里面
3. 大括号里面可以进行运算
4. 大括号里面可以调用函数

****

****

**箭头函数**

**语法:**

1. funcation用var、let、const来表示
2. 参数要写在第一个等号后面
3. 如果没有参数，需要写一对空的小括号
4. 只有一个参数那就直接写，不用加括号
5. 参数有多个，需要加一个小括号，参数用逗号隔开
6. 函数的主题内容是放在箭头后面，如果语句只有一条，那就直接写，如果语句有多条需要放在一对大括号里



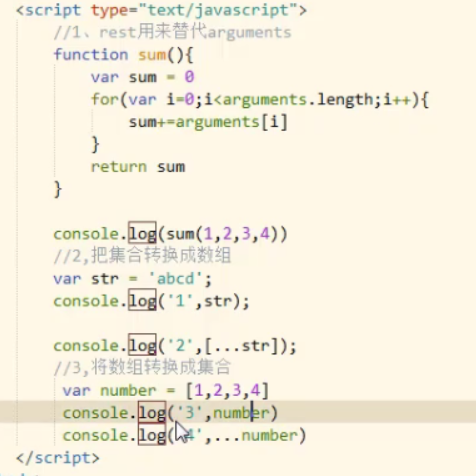
**Reset参数**

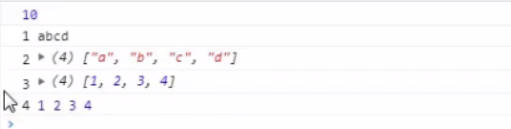
**Rest 参数 …变量名**

**Reset参数是一个数组，它的后面不能再有参数，不然报错。**

**扩展方法…**

1. **三个点后面是一个类数组，它的作用是吧这个类数组转为真正的数组，但是它需要放到一对中括号里**
2. **三个点后面是一个真正的数组，它的所用是把数组转为一个普通集合数据不需要加中括号。**

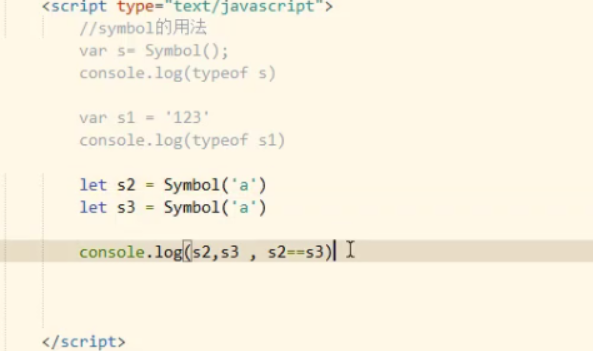
****

****

**Symbol函数**

**新增的第7中函数类型，表示独一无二。用来做属性名能保证不与其他属性名冲突。**

1. **它是通过Symbol函数生成的**
2. **它的前面不能用new,因为它生成的是一个原始类型的数据，不是对象**
3. **它可以接受一个参数，为了便于区别。即使张的一样他们也不同**
4. **不能与其他值运算，没有隐式转换**
5. **它的值可以转为boolean或字符串不能转为数字**

****

**Set集合**

**Set数据结构，类似数组。所有的数据都是唯一的，没有重复的值。它本身是一个构造函数。**

**size数据的长度**

**add()添加一个数据**

**delete()删除一个数据**

**has()是否存在某条数据**

**clear()清空所有数据**

****

**Map数据结构**

**Map数据结构，类似于对象。键值对集合所有数据都是唯一的不会重复。每条数据都要放在数组中。它本身就是一个构造函数。**

**size数据的长度**

**add()添加一个数据**

**delete()删除一个数据**

**get**

**has()是否存在某条数据**

**clear()清空所有数据**

****