vue

定义：前端渐进式框架 （mvvm）

设计模式：mvvm 、 mvc 、 mvp

mvvm:

m->model 数据模型 v->view 视图 vm->交互模式

mvc: (angular的思想完全基于mvc发展过来的)

m->model 数据模型 v->view 视图 c->controler 控制器

mvp: (swift(干ios用的))

m->model 数据模型 v->view 视图 p->provider 注册器

使用插值表达式{{}}

特点:

模块化

组件化

简洁

轻巧

易上手

vue也是一个包,需要下载

npm install vue -g

课程目录:

指令 directive

v-model 双向数据绑定

v-for 循环

v-on 事件绑定

v-bind 动态绑定 简写-》：

v-bind:class 绑定类名 简写-》：class

字符串 v-bind:class="变量"

对象 v-bind:class={'red'}

数组 v-bind:class=['red','blue']

三目运算 a?'red':'blue'

v-bind:style 动态绑定样式

v-bind:href 绑定链接

v-bind:src 绑定路径

v-show 显示或隐藏 默认值为false，即隐藏 切换display属性

v-if 显示或隐藏 默认值为true 根据是否删除dom节点

v-else-if

v-else

v-cloak 防止页面闪烁

v-once 只加载一次 尽量不要使用

事件 event 简写=》@

v-on：

普通事件：

click

dblclick

mouseenter

mouseover

mousemove

mouseup

mousedown

keydown

keyup

keypress

按键修饰符:

enter space left right up down ctrl shift

事件修饰符:

stop 阻止事件冒泡 stoppropagation

prevent 解决默认事件 preventDefault

native 原生方法

自定义指令 directive

全局自定义指令

钩子函数

Vue.directive('aa',{

bind(){} 当元素绑定时执行

inserted(){}当元素插入父级时被执行

updata(){}当内容更新时被调用

componetUpdate(){}当组件更新时被调用

unbind(){}元素解绑时被执行

})

局部自定义指令

new Vue({

directive:{

'aa':{}

}

})

自定义过滤器 filter 管道通配符

全局过滤器

局部过滤器

生命周期 lifecycle 4个阶段8个过程 **面试题**

定义：到对应阶段执行相应的函数

创建

beforeCreate 创建前

created 已经创建

挂载

beforeMount 挂载之前

mounted 已经挂载

更新

beforeUpdate 更新之前

updated 已经更新

销毁 销毁 $destroy()

beforeDestroy 销毁之前

destroyed 已经销毁

**组件 component ！！！重点**

全局组件

优点：复用性强，耦合度高

Vue.component('组件名称',{})

**单向数据流**

父子 **props**

原理：将父组件要传递的值通过v-bind绑定给子组件，子组件中用props接收这个绑定的属性名，在想要引入父组件值的地方用{{}}插入这个属性名，即可将父组件的值传递给子组件

2.子父 自定义事件**$emit**

原理：子传父要用$emit自定义一个事件，先把你要传的内容定义在data里面，然后写一个按钮，绑定一个点击事件，自定义事件的第一个参数写自定义事件的名称，第二个参数写data里面定义的内容，在父组件里给元素绑定刚才的自定义事件@recieve,自己定义事件的名称getchild，在父组件里用一个变量（写在父组件的data里面）getval接收子组件传递过来的val。当点击事件触发时即可将子组件的值传递给父组件



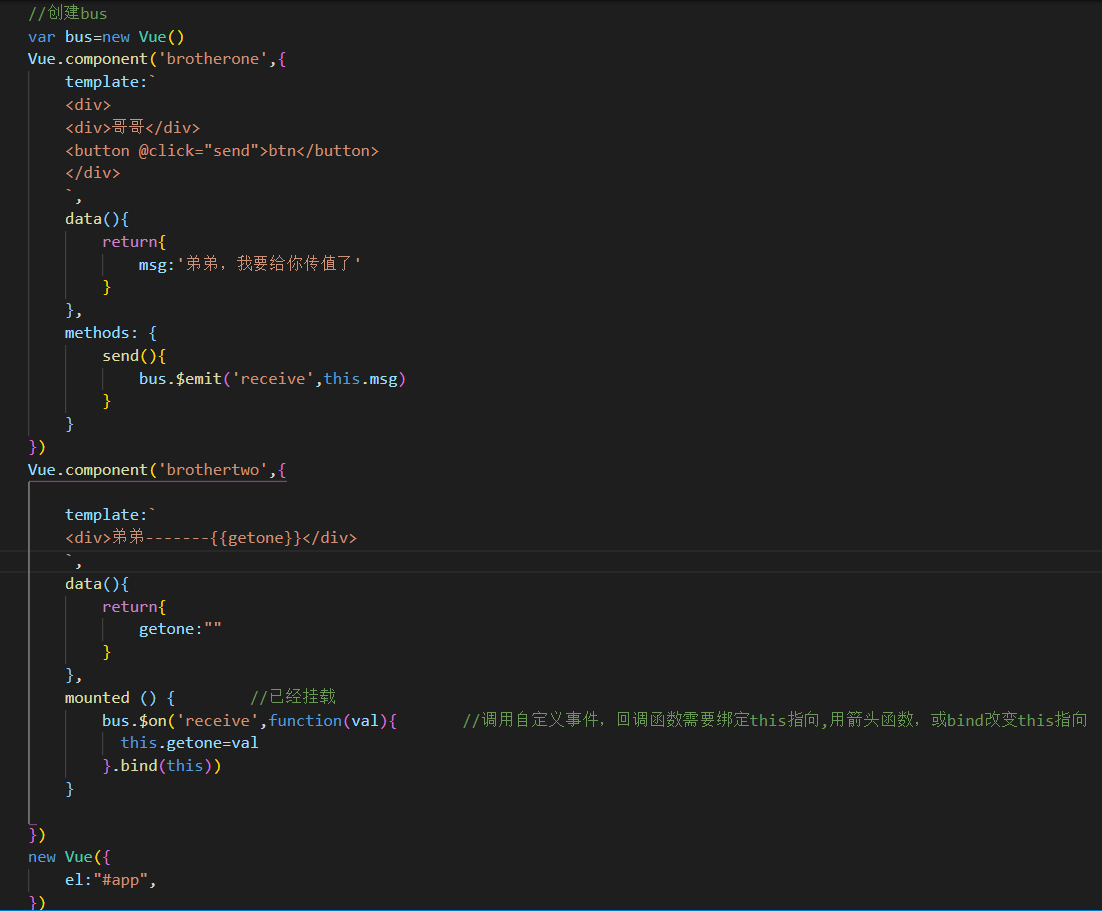
v-bind：绑定动态变化的值



3.兄弟 bus传值

bus.$emit自定义事件

bus.$on触发自定义事件



4.跨级 provide注册 inject接收

5.this$listeners

6.this.$attrs

局部组件



计算属性 computed

特点：1、简化vue在页面中的逻辑操作

2、实时更新页面的数据（购物车添加商品，总价变化）

作用：

分离数据

缓存

双向数据绑定（setter ， getter）

监听属性 watch

动画 animation

路由 router

交互(Ajax) axios/fetch(es6新增)

vuex

框架:

element-ui =>pc端

mint-ui =>mobile

antd-design-vue

vant

面试题:

1、v-model的底层原理

其实就是既绑定数据又添加一个input事件监听

通过v-bind把父组件的数据绑定到了子组件的props属性中，而在props上默认用value取值，然后通过$emit触发事件input，因为v-model绑定的事件是input，故在子组件上触发了父组件的input事件，通过触发事件来进行传值，实现了父子组件数据的双向绑定，相对于直接使用v-bind以及自定义事件代码量有所减少。

es6 的模板字符串 ``

${}

contextmenu鼠标右键事件

钩子函数：在一定的时间段内会触发相应的函数