1. **vue中的MVVM模式**

MVVM 是Model-View-ViewModel 的缩写，它是一种基于前端开发的架构模式，其核心是提供对View 和 ViewModel 的双向数据绑定，这使得ViewModel 的状态改变可以自动传递给 View，即所谓的数据双向绑定。

Vue.js 是一个提供了 MVVM 风格的双向数据绑定的 Javascript 库，专注于View 层。它的核心是 MVVM 中的 VM，也就是 ViewModel。 ViewModel负责连接 View 和 Model，保证视图和数据的一致性，这种轻量级的架构让前端开发更加高效、便捷。

1. **v-show和v-if的共同点和不同点**

V-if：是操作DOM元素的，设置条件的DOM元素必须连续写在一起，表示DOM元素是销毁还是重建;如果不满足条件的话则不会出现在dom中。

V-show:是操作CSS的，适用于频繁的切换显示和隐藏；不管漫步满足条件均在dom中，如果不满足条件，会设置成display:none;

1. **Vue如何让css只在当前组建中起作用**

第一种方法：

<style scoped>  
 …

</style>

当添加scoped参数后书写的CSS尽量避免出现tag名，用class。因为类似div[data-v-xxxxxxx]这种选择器的效率极低。

第二种：做法是外面包个大的class，手动添加作用域

第三种：在组件的最外层写一个特定的classname，然后其他的css写在这个class下面

1. **Vue中<keep-alive></keep-alive>的作用是什么？**

是Vue的内置组件，能在组件切换过程中将状态保留在内存中，防止重复渲染DOM。

keep-alive 是 Vue 内置的一个组件，可以使被包含的组件保留状态，或避免重新渲染。

<keep-alive> 包裹动态组件时，会缓存不活动的组件实例，而不是销毁它们。和 <transition> 相似，<keep-alive> 是一个抽象组件：它自身不会渲染一个 DOM 元素，也不会出现在父组件链中。

在看demo的时候看到在vue-router写着keep-alive，keep-alive的含义：  
如果把切换出去的组件保留在内存中，可以保留它的状态或避免重新渲染。为此可以添加一个keep-alive指令

1. **Vue引入组件的步骤**
2. 引入

let handsome = {

template:'<h1>{{msg}}</h1>'，

data(){

return {msg:'帅帅帅'}

}

}

1. 注册组件

components:{

myHandsome:handsome

}

1. 使用组件

<my-handsome></my-handsome>

1. **Vue指令v-el的作用**

Vue1.0中的v-el和v-ref在2.0中被废弃

v-el这个是写在html标签里的，如<div v-el:container>，这相当于告诉vue有一个div element，这个element叫container，然后在vuejs代码里可以用如this.$el.container代码，则相当于是获取到了这个div object

1. **Vue常用的插件，使用插件的步骤**

添加全局方法或者属性，如: [vue-element](https://github.com/vuejs/vue-element" \t "https://blog.csdn.net/qq_24122593/article/details/_blank)

添加全局资源：指令/过滤器/过渡等，如 [vue-touch](https://github.com/vuejs/vue-touch" \t "https://blog.csdn.net/qq_24122593/article/details/_blank)

通过全局 mixin方法添加一些组件选项，如: [vuex](https://github.com/vuejs/vuex" \t "https://blog.csdn.net/qq_24122593/article/details/_blank)

添加 Vue 实例方法，通过把它们添加到 Vue.prototype 上实现。

一个库，提供自己的 API，同时提供上面提到的一个或多个功能，如 [vue-router](https://github.com/vuejs/vue-router" \t "https://blog.csdn.net/qq_24122593/article/details/_blank)

封装的步骤：【声明插件】——【写插件】——【注册插件】——【使用插件】

1. 先引入import
2. vue.use（引入的模块）
3. New 引入的模块
4. **Vue的生命周期**

1>Vue实例有一个完整的生命周期，也就是从开始创建、初始化数据、编译模板、挂载Dom、渲染→更新→渲染、卸载等一系列过程，我们称这是Vue的生命周期。通俗说就是Vue实例从创建到销毁的过程，就是生命周期。

在Vue的整个生命周期中，它提供了一系列的事件，可以让我们在事件触发时注册js方法，可以让我们用自己注册的js方法控制整个大局，在这些事件响应方法中的this直接指向的是vue的实例。

特别值得注意的是created钩子函数和mounted钩子函数的区别

2>答：总共分为8个阶段创建前/后，载入前/后，更新前/后，销毁前/后。

创建前/后： 在beforeCreated阶段，vue实例的****挂载元素****$el和\*\*数据对象\*\*data都为undefined，还未初始化。在created阶段，vue实例的数据对象data有了，$el还没有。

载入前/后：在beforeMount阶段，vue实例的$el和data都初始化了，但还是****挂载之前****为****虚拟的dom节点****，****data.message****还未替换。在mounted阶段，vue实例挂载完成，****data.message成功渲染****。

更新前/后：当data变化时，会触发****beforeUpdate和updated****方法。

销毁前/后：在执行destroy方法后，对data的改变不会再触发周期函数，说明此时vue实例已经解除了事件监听以及和dom的绑定，但是dom结构依然存在

### 图解生命周期

1. **如果使用中间件**
2. **Vue如何和后台做交互（axios）**
3. **父子之间组件通讯**
4. **Vuex是干嘛的？Vuex中提供的方法都是干嘛的**

Dispatch commit

1. **Vue2.0对于vue1.0废弃了什么?**
2. 在1.0中，ready钩子函数已废弃不用，进而使用mounted替换

同时还新增了beforeMount、beforeUpdate、updated等，私以为越来越向react看齐了有木有。。

beforeUpdate的作用是在页面初始渲染视图之后，一旦监测到data发生变化，在变化的数据重新渲染视图之前会触发beforeUpdate钩子函数，这也是重新渲染之前最后修改数据的机会

updated的作用则是在data发生变化渲染更新视图之后触发。

1. 同时废弃的还有events、$dispatch、$broadcas

官方推荐使用vuex或者全局的事件驱动，然而废弃的这些方法在vux UI框架中很多地方都有使用，无疑在项目中用到它的地方在2.0版本都会不起作用，甚至会报错。

1. v-ref、v-el 弃用

统一使用ref属性为元素或组件添加标记，然后通过this.$refs获取

1. $els 是用来获取元素DOM对象弃用

$refs可以起到替代性作用。

1. .v-for循环中常用的$index、$key也已不支持使用

trackby被key属性替换。

6. 自定义组件中的partial，弃用，这个一直没用到

7. 新增 v-once指令

8. 新增 propsData

9. 新增 render

1. **Computed和其他方法的区别**
   1. 计算属性有缓存功能，其他方法没有

是依赖其他属性的（属性是vm实例的响应式数据）

* 1. Computed有get和set方法，其他方法没

Get必须有返回值，默认执行的

设置计算书形式调用set方法

1. vue常用指令

1、v-if="表达式"，根据表达式的结果来判断，是true则渲染元素，false则将元素注释掉

2、v-once:不允许修改数据

1. v-model:将组件与变量进行双向绑定，当组件数据修改时，变量会随之改变。它后面可加修饰符，例如.lazy，只有在标签中的值发生改变时才同步到变量，.trim过滤用户输入中的空格。

4、v-show与v-if类似，true就渲染，但是false并不是注释掉，而是通过display:none;来将元素隐藏，如果需要来回切换，使用v-show更好。

5、v-else、v-else-if与v-if连在一起使用构成逻辑判断，不可以单独或者分开使用

6v-for用于遍历元素，使用格式为v-for="(值的形参,索引) in 遍历内容"，其遍历的内容可以说数组/对象/字符串，例如在js的data中有一个person对象，person:{name:'tony',age:15,sex:'男'},将其内容在页面遍历输出：

7、v-text：向标签内注入数据，并且覆盖标签内的其他内容

8、v-html：向标签内覆盖注入HTML内容作为其子元素

9、v-bind：给标签绑定属性、类、样式等，可以缩写为冒号，

10、v-on:给标签绑定函数，可以缩写为@，例如绑定一个点击函数

同步更新类似于刷新，更新一个内容要把所有的内容都刷新一遍。而异步则只刷新更新或修改的部分，不重新将所有内容全部刷新一遍。异步可以减少服务器的压力和等待时间