

1、active-class是哪个组件的属性？嵌套路由怎么定义？

答：vue-router模块的router-link组件。

2、怎么定义vue-router的动态路由？怎么获取传过来的动态参数？

答：在router目录下的index.js文件中，对path属性加上/:id。 使用router对象的params.id

3、vue-router有哪几种导航钩子？

答：三种，
一种是全局导航钩子：router.beforeEach(to, from, next)，作用：跳转前进行判断拦截。
第二种：组件内的钩子；
第三种：单独路由独享组件

4、scss是什么？安装使用的步骤是？有哪几大特性？

答：css的预编译。
使用步骤：
第一步：用npm 下三个loader（sass-loader、css-loader、node-sass）
第二步：在build目录找到webpack.base.config.js，在那个extends属性中加一个拓展.scss
第三步：还是在同一个文件，配置一个module属性
第四步：然后在组件的style标签加上lang属性，例如：lang="scss"
有哪几大特性：
1、可以用变量，例如（\$变量名称=值）；
2、可以用混合器，例如（）
3、可以嵌套

5、mint-ui是什么？怎么使用？说出至少三个组件使用方法？

答：基于vue的前端组件库。npm安装，然后import样式和js，
vue.use（mintUi）全局引入。在单个组件局部引入：
import {Toast} from 'mint-ui'。组件一：
Toast('登录成功')；组件二：mint-header；组件三：mint-swiper

6、v-model是什么？怎么使用？vue中标签怎么绑定事件？

答：可以实现双向绑定，指令（v-class、v-for、v-if、v-show、v-on）。
vue的model层的数据属性。绑定事件：<input @click=doLog() />

7、axios是什么？怎么使用？描述使用它实现登录功能的流程？

答：请求后台资源的模块。npm install axios -S装好，然后发送的是跨域，需在配置文件中config/index.js进行设置。后台如果是Tp5则定义一个资源路由。js中使用import进来，然后.get或.post。返回在.then函数中如果成功，失败则是在.catch函数中

8、axios+tp5进阶中，调用axios.post('api/user')是进行的什么操作？axios.put('api/user/8')呢？

答：跨域，添加用户操作，更新操作。

9、什么是RESTful API？怎么使用？

答：是一个api的标准，无状态请求。请求的路由地址是固定的，如果是tp5则先路由配置中把资源路由配置好。标准有：.post .put .delete

10、vuex是什么？怎么使用？哪种功能场景使用它？

答：vue框架中状态管理。在main.js引入store，注入。
新建了一个目录store，... export 。
场景有：单页应用中，组件之间的状态。音乐播放、登录状态、加入购物车

11、mvvm框架是什么？它和其它框架（jquery）的区别是什么？哪些场景适合？

答：一个model+view+viewModel框架，数据模型model，viewModel连接两个
区别：vue数据驱动，通过数据来显示视图层而不是节点操作。
场景：数据操作比较多的场景，更加便捷

12、自定义指令（v-check、v-focus）的方法有哪些？它有哪些钩子函数？还有哪些钩子函数参数？

答：全局定义指令：在vue对象的directive方法里面有两个参数，一个是指令名称，另外一个函数。组件内定义指令：directives
钩子函数：bind（绑定事件触发）、inserted（节点插入的时候触发）、update（组件内相关更新）
钩子函数参数：el、binding

13、说出至少4种vue当中的指令和它的用法？

答：v-if：判断是否隐藏；v-for：数据循环出来；v-bind:class：绑定一个属性；v-model：实现双向绑定

14、vue-router是什么？它有哪些组件？

答：vue用来写路由一个插件。router-link、router-view

15、导航钩子有哪些？它们有哪些参数？

答：导航钩子有：a/全局钩子和组件内独享的钩子。b/beforeRouteEnter、afterEnter、beforeRouterUpdate、beforeRouteLeave
参数：有to（去的那个路由）、from（离开的路由）、next（一定要用这个函数才能去下一个路由，如果不用就拦截）最常用就这几种

16、Vue的双向数据绑定原理是什么？

答：vue.js 是采用数据劫持结合发布者-订阅者模式的方式，通过Object.defineProperty()来劫持各个属性的setter, getter, 在数据变动时发布消息给订阅者，触发相应的监听回调。

具体步骤：

第一步：需要observe的数据对象进行递归遍历，包括子属性对象的属性，都加上 setter和getter

这样的话，给这个对象的某个值赋值，就会触发setter，那么就能监听到了数据变化

第二步：compile解析模板指令，将模板中的变量替换成数据，然后初始化渲染页面视图，并将每个指令对应的节点绑定更新函数，添加监听数据的订阅者，一旦数据有变动，收到通知，更新视图

第三步：Watcher订阅者是Observer和Compile之间通信的桥梁，主要做的事情是：

1、在自身实例化时往属性订阅器(dep)里面添加自己

2、自身必须有一个update()方法

3、待属性变动dep.notice()通知时，能调用自身的update()方法，并触发Compile中绑定的回调，则功成身退。

第四步：MVVM作为数据绑定的入口，整合Observer、Compile和Watcher三者，通过Observer来监听自己的model数据变化，通过Compile来解析编译模板指令，最终利用Watcher搭起Observer和Compile之间的通信桥梁，达到数据变化 -> 视图更新；视图交互变化(input) -> 数据model变更的双向绑定效果。

ps：16题答案同样适合“vue data是怎么实现的？”此面试题。

17、请详细说下你对vue生命周期的理解？

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

18、请说下封装 vue 组件的过程？

答：首先，组件可以提升整个项目的开发效率。能够把页面抽象成多个相对独立的模块，解决了我们传统项目开发：效率低、难维护、复用性等问题。

然后，使用Vue.extend方法创建一个组件，然后使用Vue.component方法注册组件。子组件需要数据，可以在props中接受定义。而子组件修改好数据后，想把数据传递给父组件。可以采用emit方法。

19、你是怎么认识vuex的？

答：vuex可以理解为一种开发模式或框架。比如PHP有thinkphp，java有spring等。

通过状态（数据源）集中管理驱动组件的变化（好比spring的IOC容器对bean进行集中管理）。

应用级的状态集中放在store中； 改变状态的方式是提交mutations，这是个同步的事物； 异步逻辑应该封装在action中。

20、vue-loader是什么？使用它的用途有哪些？

答：解析.vue文件的一个加载器，跟template/js/style转换成js模块。
用途：js可以写es6、style样式可以scss或less、template可以加jade等

21、请说出vue.cli项目中src目录每个文件夹和文件的用法？

答：assets文件夹是放静态资源；components是放组件；router是定义路由相关的配置；view视图；app.vue是一个应用主组件；main.js是入口文件

22、vue.cli中怎样使用自定义的组件？有遇到过哪些问题吗？

答：第一步：在components目录新建你的组件文件（smithButton.vue），script一定要export default {
第二步：在需要用的页面（组件）中导入：import smithButton from '../components/smithButton.vue'
第三步：注入到vue的子组件的components属性上面，components:{smithButton}
第四步：在template视图view中使用，<smith-button> </smith-button>
问题有：smithButton命名，使用的时候则smith-button。

23、聊聊你对Vue.js的template编译的理解？

答：简而言之，就是先转化成AST树，再得到的render函数返回VNode（Vue的虚拟DOM节点）
详情步骤：
首先，通过compile编译器把template编译成AST语法树（abstract syntax tree 即 源代码的抽象语法结构的树状表现形式），compile是createCompiler的返回值，createCompiler是用以创建编译器的。另外compile还负责合并option。
然后，AST会经过generate（将AST语法树转化成render function字符串的过程）得到render函数，render的返回值是VNode，VNode是Vue的虚拟DOM节点，里面有（标签名、子节点、文本等等）

24、vue的组件是怎么定义的？父组件怎么给子组件传值？

答：首先注册vue.components，第一个参数是组件名称，第二个参数是选项。
直接绑定一个属性，然后在子组件props里面接收

25、使用过element.ui吗？说下它其中两个组件的使用方法？

答：使用过用过一个布局的，它是由24份，它的写法是：span后面带的数字它占24份里面的宽度。：offset是它的间距，后面也是跟数字，也是从24份里面取的。
input按钮，标签是el-input，后面type跟上一个属性就是显示不同按钮的类型，有默认的default（默认的）、success（成功的）、warning（警告）、danger（危险）、info（）、primary（）

26、说下你对mvvm的理解？双向绑定的理解？

答：mvvm就是vm框架视图、m模型就是用来定义驱动的数据、v经过数据改变后的html、vm就是用来实现双向绑定
双向绑定：一个变了另外一个跟着变了，例如：视图一个绑定了模型的节点有变化，模型对应的值会跟着变

27、说出你所使用过的vue指令

答：v-on (监听事件、@change、@click)、v-if (判断的)、v-show (显示/隐藏)、v-bind (绑定属性、:disabled、:src)

28、你觉得怎样的自定义组件是完善的？至少说出4点

答：第一点、可以通用
第二点、代码尽量简洁
第三点、容易修改
第四点、功能要多一点

一、请说下具体使用vue的理解？

答：1、使用vue不必担心布局更改和类名重复导致的js重写，因为它是靠数据驱动双向绑定，底层是通过Object.defineProperty() 定义的数据 set、get 函数原理实现。
2、组件化开发，让项目的可拓展性、移植性更好，代码重用性更高，就好像农民工建房子，拿起自己的工具包就可以开工。项目经理坐等收楼就好。
3、单页应用的体验零距离接触安卓原生应用，局部组件更新界面，让用户体验更快速省时。
4、js的代码无形的规范，团队合作开发代码可阅读性更高。

二、你觉得哪些项目适合vue框架？

答：1、数据信息量比较多的，反之类似企业网站就无需此框架了。
2、手机web和app应用多端共用一套界面的项目，因为使用vue.cli+webpack后的前端目录，非常有利于项目的跨平台部署。

三、怎么理解MVVM模式的这些框架？

答：1、M就是Model模型层，存的一个数据对象。
2、V就是View视图层，所有的html节点在这一层。
3、VM就是ViewModel，它通过data属性连接Model模型层，通过el属性连接View视图层。

四、PC端项目你会在哪些场景使用Vue框架？

答：上万级数据需要瀑布流更新和搜索的时候，因为数据庞大的时候，用原生的dom操作js和html都会有列表的html布局，迭代很困难。再一个dom节点的大面积添加会影响性能。
那么vue为什么解决这些问题呢？
第一：只需用v-for在view层一个地方遍历数据即可，无需复制一段html代码在js和html两个地方。
第二：vue通过Virtual Dom就是在js中模拟DOM对象树来优化DOM操作。

vuex 1、vuex有哪几种属性？

答：有五种，分别是 State、Getter、Mutation、Action、Module

2、vuex的State特性是？

答：

- 一、Vuex就是一个仓库，仓库里面放了很多对象。其中state就是数据源存放地，对应于与一般Vue对象里面的data
- 二、state里面存放的数据是响应式的，Vue组件从store中读取数据，若是store中的数据发生改变，依赖这个数据的组件也会发生更新
- 三、它通过mapState把全局的 state 和 getters 映射到当前组件的 computed 计算属性中

3、vuex的Getter特性是？

答：

- 一、getters 可以对State进行计算操作，它就是Store的计算属性
- 二、虽然在组件内也可以做计算属性，但是getters 可以在多组件之间复用
- 三、如果一个状态只在一个组件内使用，是可以不用getters

4、vuex的Mutation特性是？

答：

- 一、Action 类似于 mutation，不同在于：
- 二、Action 提交的是 mutation，而不是直接变更状态。
- 三、Action 可以包含任意异步操作

5、Vue.js中ajax请求代码应该写在组件的methods中还是vuex的actions中？

答：

- 一、如果请求来的数据是不是要被其他组件公用，仅仅在请求的组件内使用，就不需要放入vuex 的state里。
- 二、如果被其他地方复用，这个很大几率上是需要的，如果需要，请将请求放入action里，方便复用，并包装成promise返回，在调用处用async await处理返回的数据。如果不要复用这个请求，那么直接写在vue文件里很方便。

6、不用Vuex会带来什么问题？

答：

- 一、可维护性会下降，你要想修改数据，你得维护三个地方
- 二、可读性会下降，因为一个组件里的数据，你根本就看不出来是从哪来的
- 三、增加耦合，大量的上传派发，会让耦合性大大的增加，本来Vue用Component就是为了减少耦合，现在这么用，和组件化的初衷相背。

生命周期 1、什么是vue生命周期？

答：Vue 实例从创建到销毁的过程，就是生命周期。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。

2、vue生命周期的作用是什么？

答：它的生命周期中有多个事件钩子，让我们在控制整个Vue实例的过程时更容易形成好的逻辑。

3、vue生命周期总共有几个阶段？

答：它可以总共分为8个阶段：创建前/后，载入前/后，更新前/后，销毁前/销毁后

4、第一次页面加载会触发哪几个钩子？

答：第一次页面加载时会触发 beforeCreate, created, beforeMount, mounted 这几个钩子

5、DOM 渲染在 哪个周期中就已经完成？ 答：DOM 渲染在 mounted 中就已经完成了。

6、简单描述每个周期具体适合哪些场景？

答：生命周期钩子的一些使用方法： beforecreate：可以在这加个loading事件，在加载实例时触发 created：初始化完成时的事件写在这里，如在这结束loading事件，异步请求也适宜在这里调用 mounted：挂载元素，获取到DOM节点 updated：如果对数据统一处理，在这里写上相应函数 beforeDestroy：可以做一个确认停止事件的确认框 nextTick：更新数据后立即操作dom

axios 1、axios的特点有哪些？

答：

- 一、Axios 是一个基于 promise 的 HTTP 库，支持promise所有的API
- 二、它可以拦截请求和响应
- 三、它可以转换请求数据和响应数据，并对响应回来的内容自动转换成 JSON类型的数据
- 四、安全性更高，客户端支持防御 XSRF

2、axios有哪些常用方法？

答：

- 一、axios.get(url[, config]) //get请求用于列表和信息查询
- 二、axios.delete(url[, config]) //删除
- 三、axios.post(url[, data[, config]]) //post请求用于信息的添加
- 四、axios.put(url[, data[, config]]) //更新操作

3、说下你了解的axios相关配置属性？

答：

``url`` 是用于请求的服务器URL

``method`` 是创建请求时使用的方法,默认是get

``baseUrl`` 将自动加在 ``url`` 前面,除非 ``url`` 是一个绝对URL。它可以通过设置一个 ``baseUrl`` 便于为axios实例的方法传递相对URL

``transformRequest`` 允许在向服务器发送前,修改请求数据,只能用在 'PUT', 'POST' 和 'PATCH' 这几个请求方法

``headers`` 是即将被发送的自定义请求头

headers: { 'X-Requested-With': 'XMLHttpRequest' },

``params`` 是即将与请求一起发送的URL参数,必须是一个无格式对象(plainobject)或URLSearchParams对象

```
params: {  
  ID: 12345  
},
```

``auth`` 表示应该使用HTTP基础验证,并提供凭据

这将设置一个`Authorization`头，覆写掉现有的任意使用`headers`设置的自定义`Authorization`头

```
auth:{  
  username: 'janedoe',  
  password: 's00pers3cret'  
},
```

`proxy`定义代理服务器的主机名称和端口

`auth`表示HTTP基础验证应当用于连接代理，并提供凭据

这将会设置一个`Proxy-Authorization`头，覆写掉已有的通过使用`header`设置的自定义`Proxy-Authorization`头。

```
proxy:{  
  host: '127.0.0.1',  
  port: 9000,  
  auth:{  
    username: 'mikeymike',  
    password: 'rapunz31'  
  }  
},
```

keep-alive

vue性能

polyfill import引入并执行 应用场景：页面开始一片空白 1、vue响应式原理？ 2、vue-router实现原理？ 3、为什么要选vue？与其它框架对比的优势和劣势？ 4、vue如何实现父子组件通信，以及非父子组件通信？ 5、vuejs与angularjs以及react的区别？ 6、vuex是用来做什么的？ 7、vue源码结构