自我介绍

您好，我叫张耀昱，24岁，吉林四平人，电子信息科学与技术专业，我的上一家公司是做网站外包的，3年的工作经验里做过很多门户网站静态页h5页面，还有商城后台管理系统等等。今天来是想面试贵公司的web前端岗位，技术方面我一直从事于web方面的工作，掌握的前端技术有vue全家桶，js，div+css，html，ajax同时我也了解jquery，也能熟练使用PS做出网页PSD图，我喜欢全心全意的投入工作中，对于工作也很认真负责。

谢谢，以上是我的自我介绍。

项目介绍：在19年6月到20年1月，做的是中诺后台管理项目，

我主要负责用户注册、用户管理、权限管理模块。

我着重介绍一下用户注册模块

1.创建登录组件并配置路由 - 注册点击登录的事件

- 表单验证

- 发请求提交

- 根据请求结果做下一步处理

2.布局结构用flex布局 - 获取表单数据（根据接口要求使用 v-model 绑定）

3.数据绑定（v-model）4.表单验证 使用专门的验证插件 vee-validate 使用内置验证规则和使用自定义验证规则（正则）5。验证通过后获取token，存储token

**1.computed与watch  用在什么业务场景 有什么区别**

Computed具有缓存功能，只有它依赖的data中的数据发生变化的时候才会重新执行，如果data中的数据没有变化，计算属性不会重新执行。

Methods调用一次执行一次

Watch属性是用来监测数据变化的，当数据变化之后执行的方法。当需要在数据变化时执行异步或开销较大的操作时，侦听器最有效

都是用来监听数据的变化，他们执行过程是不一样的。方法methods调一次执行一次，而计算属性执行完毕之后，会把执行的结果缓存起来，下次再调用的时候，如果data中的数据没有变化，计算属性不会重新执行，从缓存中加载数据。计算属性依赖于data中数据的变化，计算属性没有参数。

**2.过滤器的用法 业务场景**

用在两个地方：双花括号插值和v-bind表达式中，用于常见的文本格式化，比如对日期的格式化。有全局过滤器和局部过滤器两种,全局过滤器不用加s，filter。局部过滤器加s，filters。至少要有一个参数。{{ time | fmtDate }} vue.filter（filfate，function（）{}）全局过滤器

**3.vue项目的搭建**

1. 安装vue-cli Npm i -g vue-cli

2. 创建项目

Npm init webpack simple 你项目的名称

Cd 你项目的名称

Npm install

Npm run dev

Npm run build

首先页面放在pags或者views中

静态文件放在static中

资源文件放在assets中

样式文件放在styles中

辅助库放在utils中

配置文件可以放在config或者constants中

vuex的文件放在stores中，至于getter，操作，突变，模块可以参考vuex的文档

路由文件放在routes中

所有组件放在components中

共享代码也可以使用shared作为目录

布局组件可以放在layouts目录中

**4.hash模式和history模式的区别**

hash模式：符号#以及#后面的字符被称为hash 用windows.location.hash获取，hash模式的工作原理是hashchange事件，可以在window监听hash的变化

history模式：这种模式充分利用 history.pushState API 来完成 URL 跳转而无须重新加载页面，这种模式需要后台配置支持

hash模式url里面永远带着#号，我们在开发当中默认使用这个模式。如果用户考虑url的规范那么就需要使用history模式，因为history模式没有#号，是个正常的url适合推广宣传。当然其功能也有区别，比如我们在开发app的时候有分享页面，那么这个分享出去的页面就是用vue或是react做的，咱们把这个页面分享到第三方的app里，有的app里面url是不允许带有#号的，所以要将#号去除那么就要使用history模式，但是使用history模式还有一个问题就是，在访问二级页面的时候，做刷新操作，会出现404错误，那么就需要和后端人配合让他配置重定向到你的页面路由上就ok啦。

1. **常用的生命周期，有啥区别   created与mounted**

Vue实例从创建到销毁的过程，就是生命周期。

从开始创建、初始化数据、变编译模板、挂载DOM->渲染、更新->渲染、销毁等一系列过程，称之为Vue的生命周期

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

1、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

2、在created阶段，vue实例的数据对象data有了，$el还没有载入前后

3、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。

4、在mounted阶段，vue实例挂载完成，data.message成功渲染更新前后

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

6、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

created与mounted 前者获取data里面数据，没有DOM 后者 用上面编译好的html内容替换掉el所指向的DOM对象，此过程中进行ajax操作

**6.业务中使用闭包的场景**

闭包就是能够读取其他函数内部变量的函数。需要获取函数内部变量时 。闭包可以用在许多地方。它的最大用处有两个，一个是可以读取函数内部的变量，另一个就是让这些变量的值始终保持在内存中，

在模块是开发是封装request请求时，配置baseurl就是用了闭包

**7 scoped原理**

实现组件的私有化，不对全局造成样式污染，表示当前style属性只属于当前模

块虽然方便但是我们需要慎用，因为当我们使用公共组件时会造成很多困难，

增加难度，想知道造成那些难度，需要先从scoped实现原理了解

**8.描述一下cookie，sessionStorage、localStorage的区别？**

Cookie：本身是用来客户端服务器之间通信用的

存储小只有4kb，每次http请求都带着，影响获取资源效率

sessionStorage、localStorage：存储大有5M，存储在本地，

方法：getitem获取内容 setitem修改和新增

sessionStorage：生命周期为当前窗口和标签，一旦当前窗口和标签关闭，就会清理

localStorage：生命周期为永久，只要不清除localStorage信息，信息就永远存在

**9 css-loader，style-loader**

css-loader 的作用是处理css中的 @import 和 url 这样的外部资源

style-loader 的作用是把样式插入到 DOM中，方法是在head中插入一个style标签，并把样式写入到这个标签的 innerHTML 里

要把CSS文件的内容插入DOM，所以我们要获取CSS文件的样式。如果按照默认的从右往左的顺序，我们使用 css-loader ，它返回的结果是一段JS字符串，这样我们就取不到CSS样式了。为了获取CSS样式，我们会在 style-loader 中直接通过require来获取，这样返回的JS就不是字符串而是一段代码了。也就是我们是先执行style-loader，在它里面再执行 css-loader。 同样的字符串，但是在默认模式下是当字符串传入的，在pitching模式下是当代码运行的，就是这个区别。

也就是，我们处理CSS的时候，其实是 styled-loader先执行了，它里面会调用 css-loader 来拿到CSS的内容，拿到的内容当然是经过css-loader 编译过的。

需要提的一点是，其实 css-loader 返回的也不是css的内容，而是一个对象，不过他的 toString() 方法会直接返回css的样式内容，那么为什么是这样的呢，因为这个是 css-loader 返回的结果

**10.vue组件渲染和更新的过程？**

\1. 把模板编译为render函数

\2. 实例进行挂载, 根据根节点render函数的调用，递归的生成虚拟dom

\3. 对比虚拟dom，渲染到真实dom

\4. 组件内部data发生变化，组件和子组件引用data作为props重新调用render函数，生成虚拟dom, 返回到步骤3

5 **jsonp具体实施细节**

jsonp(JSON with padding)你一定不会陌生，前端向后端拿数据的方式之一，也是处理跨域请求的得利助手。

ajax请求受同源策略影响，不允许进行跨域请求，而script标签src属性中的链接却可以访问跨域的js脚本，利用这个特性，服务端不再返回JSON格式的数据，而是返回一段调用某个函数的js代码，在src中进行了调用，这样实现了跨域。

**Jsonp优点:**

完美解决在测试或者开发中获取不同域下的数据,用户传递一个callback参数给服务端，然后服务端返回数据时会将这个callback 参数作为函数名来包裹住JSON数据，这样客户端就可以随意定制自己的函数来自动处理返回数据了。

**Jsonp缺点:**

Jsonp只支持get请求而不支持post请求,也即是说如果想传给后台一个json格式的数据,此时问题就来了,浏览器会报一个http状态码415错误,告诉你请求格式不正确.

**11单双页面优缺点**

单页面应用(SPA),通俗一点说就是指只有一个主页面的应用，浏览器一开始要加载所有必须的html,js,css.

　　优点：用户体验好

　　　　　前后端分离

　　　　　页面效果会比较炫酷（比如切换页面内容时的专场动画）

　　缺点：不利于seo

　　　　　导航不可用，如果一定要导航需要自行实现前进，后退。

　　　　　初次加载时耗时多

　　　　　页面负责度提高很多

.什么是多页面？

　　多页面(MPA),就是只一个应用中有多个页面，页面跳转时是整页刷新

　　优点：有利于seo

　　　　　开发成本较低

　　缺点：网站的后期维护难度较大

　　　　　页面之间的跳转用时较长，用户体验较差。

　　　　　代码重复度大

**12. vue中怎么处理跨域**

axios方法，到min.js 我们需要将axios文件进行引入到min.js当中 全局使用axios

min.js中设置请求头

axios.defaults.headers.common["tonken"] = "asd9iass939kasd993w3k9"

axios.defaults.headers.post["Content-type"] = "application/json"

生产环境跨域要后台设置或者用中间层做代理，和vue没太大关系。开发环境如果你用的是vue-cli，那么找到config/index.js文件