

2021年我的前端面试小结(70题)



大部分情况，面试官简单说一下自己然后让你自我介绍一下。这是第一步，基本都会有，我也有碰见不需要让我自我介绍的面试官(雷厉风行的那种 说你基本情况简历有，我们跳过自我介绍环节直接开始)

「自我介绍」：面试官早上/下午好，我叫王小二，某年某学校某专业毕业，今天来应聘贵公司的前端工程师岗位，我从事前端开发三年多，然后简单描述第一家入职情况，第二家公司入职情况。说一下自我性格等等，巴拉巴拉精简一点

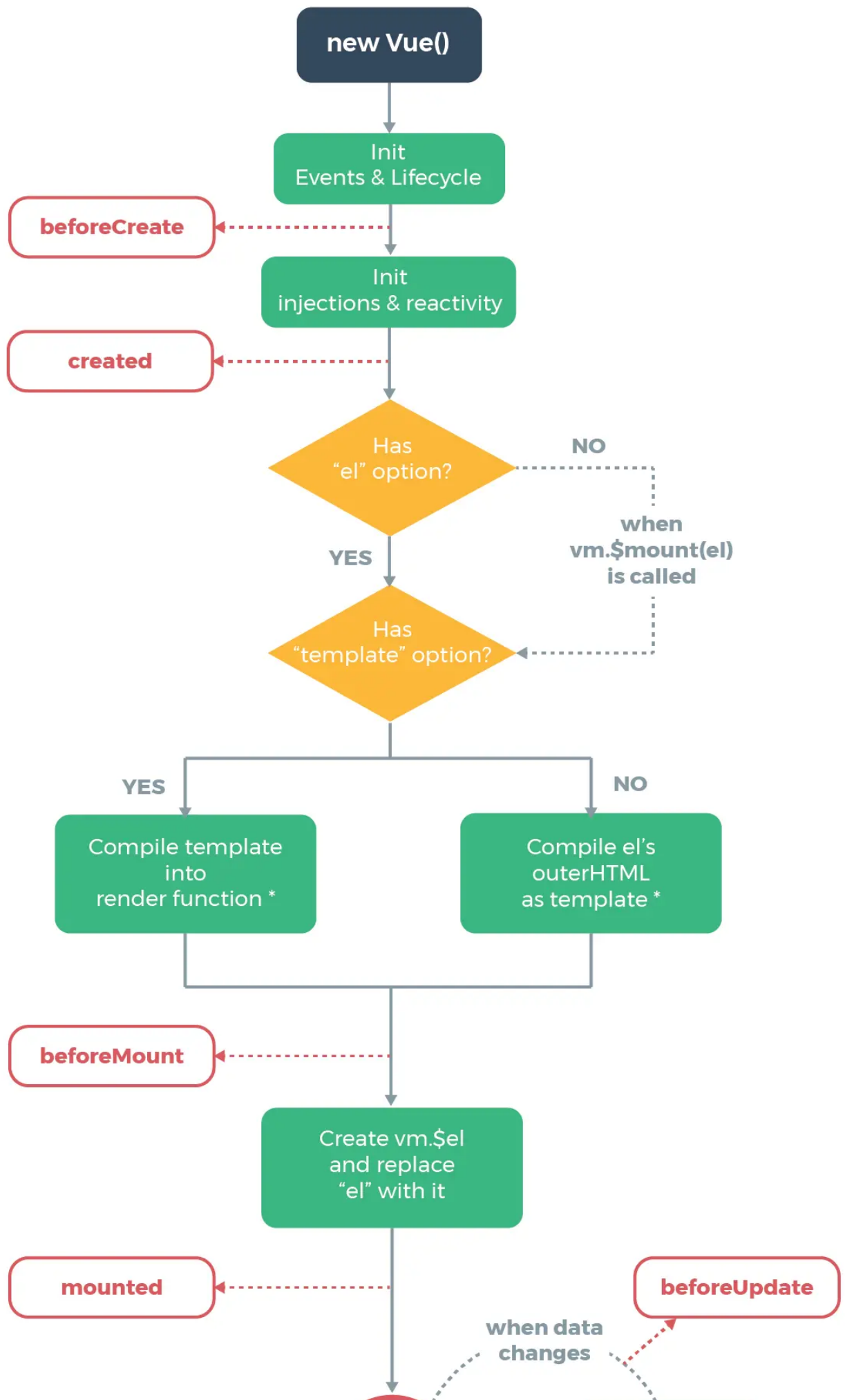


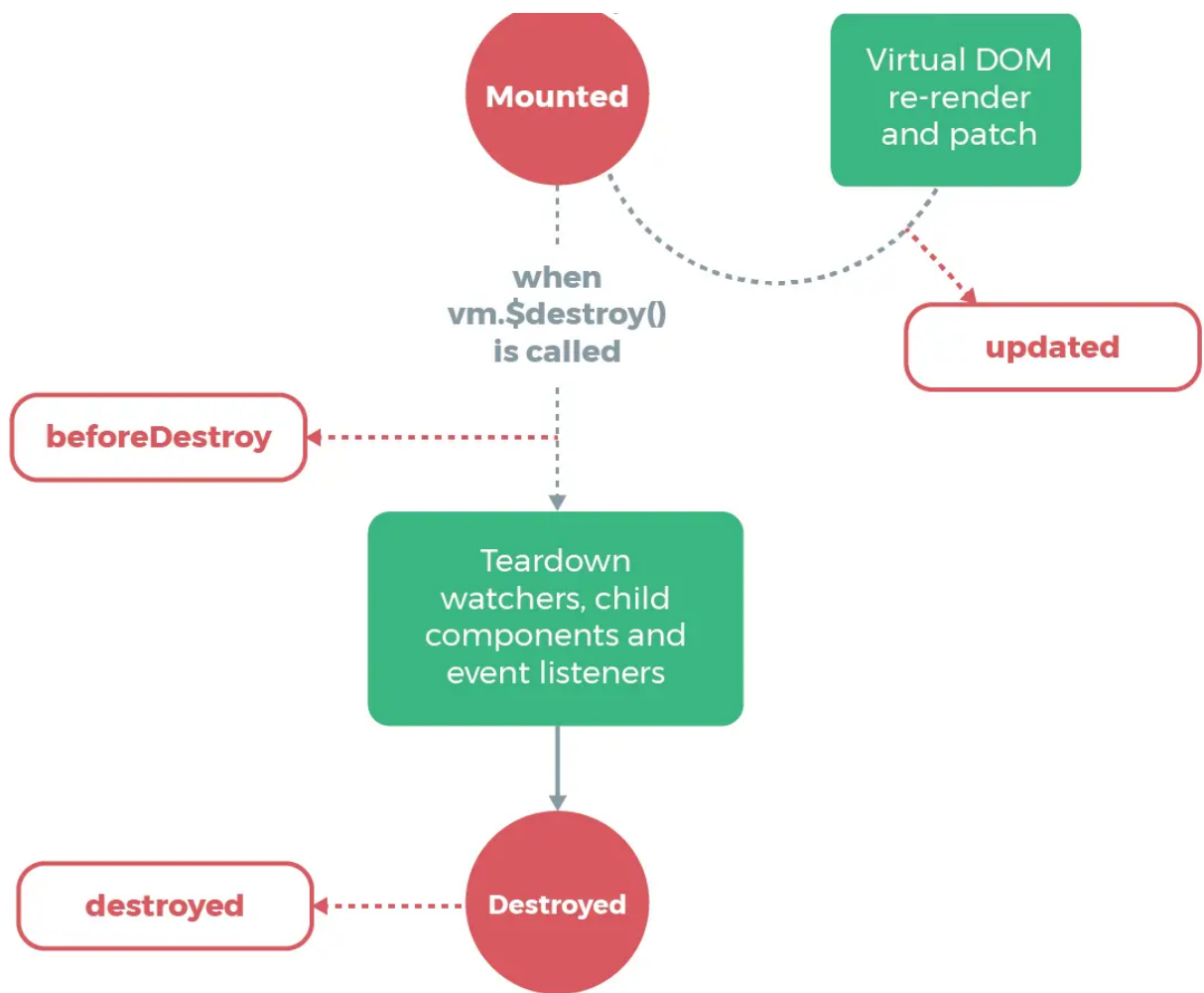
这里是我记录的一些面试题 没有难度顺序 只是一个简单复盘，个别题目有点小重复，没有进行二次过滤。下面面试题基本以vue2为准，少部分涉及vue3

1.生命周期

`beforeCreate => created => beforeMount => Mounted => beforeUpdate => updated => beforeDestroy => destroyed`

keep-alive下: `activated deactivated`





* template compilation is performed ahead-of-time if using a build step, e.g. single-file components

@稀土掘金技术社区

扩展链接

[vue官网文档](#)

2.父子生命周期顺序

父beforeCreate => 父created => 父beforeMount => 子beforeCreate => 子created =>子beforeMount => 子Mounted=>父Mounted

子组件先挂载 然后到父组件,更新也类似 父beforeUpdate =>子beforeUpdate => 子updated => 父updated

扩展链接

[vue官网文档](#)

3.父子组件传值

vue2为例: `$emit`, `props`, `$ref`, `$parent`, `$children`, `eventBus`, `provide/inject`, `vuex`

扩展链接

[父子组件传值](#)

4.跨域8种解决方案

- `iframe + document.domain location.hash window.name`等三种
- `postMessage`
- `proxyTable`
- `nodejs` 中间件
- `cors`
- `websocket`
- `jsonp`
- `nginx`反向代理

扩展链接

5. 普通函数 箭头函数的区别* 构造函数

- 1.箭头函数没有原型 原型是undefined
- 2.箭头函数this指向全局对象 而函数指向引用对象
- 3.call, apply, bind方法改变不了箭头函数的指向

改正一下第二点：箭头函数的this指向创建时 父级的this。 或者用另一个兄弟说的：箭头函数的this应该是创建时所在作用域指向的对象

扩展链接

[普通函数与箭头函数的区别是什么？](#)

6.reduce(累加器的使用,自己实现)

- arr.reduce(a,b)
- a为一个函数，b为初始值
- arr.reduce((a,b,c,d)=>{a为初始值，计算后的total b为当前数组值 c为当前idnex d为整个数组 })

扩展链接

[8个JS的reduce使用实例，和reduce的骚操作](#)

7.jq和vue什么区别 *

Vue.js 一个核心思想是数据驱动。所谓数据驱动，是指视图是由数据驱动生成的，我们对视图的修改，不会直接操作 DOM，而是通过修改数据。它相比我们传统的前端开发，如使用 jQuery 等前端库直接修改 DOM，大大简化了代码量。特别是当交互复杂的时候，只关心数据的修改会让代码的逻辑变的非常清晰，因为 DOM 变成了数据的映射，我们所有的逻辑都是对数据的修改，而不用碰触 DOM，这样的代码非常利于维护。

- vue数据驱动
- jq控制dom元素
- 简单或者很少dom元素 如果对比两者看上去差距不大 dom元素和操作的多的话，就会发现vue更加方便简洁
- vue渲染优雅，代码易维护

8.基本数据类型、引用类型、基本类型的区别...

string、boolean、number、null、undefined、symbol、bigInt

object

[js数据类型](#)

9.promise 和 async await相关知识 es6

es6的知识点：

- 声明和表达式：let const 解构赋值 Symbol
- 内置对象：Map和Set proxy和reflect
- 字符串模板
- 函数：参数扩展 箭头函数 迭代器 for of
- class类
- export和import 模块
- promise async await和generator

[阮一峰的es6学习](#)

10.宏任务和微任务(event loop) promise属于哪个

- 事件循环
- 任务队列
- 宏任务
- 微任务 [js中的宏任务与微任务](#)

11.vue2和vue3的区别 (vue3快在哪)

- diff算法增加patchFlag静态标识，只对比有静态标识的dom元素
- 事件增加缓存
- 很多文本节点提升 只定义一次，渲染时不需要再次定义，vue2每次都需要重新定义
- ssr渲染 以字符串方式渲染
- proxy替换了之前的defineProperty
- vite
- ts ...

12.vue2的一些原理 (深入了解一下)

nextTick、diff、虚拟dom、watch、等等

[vue2原理学习链接](#)

13.cesium和three.js、babylon.js 地图引擎和3d地图等等(个别需要)

地图或者3D相关知识

14.vuex的一些知识 (扩展 router axios)

state modules getter actions mutation

[vuex官网](#)

15.axios fetch ajax区别

axios的拦截器interceptors request response

[ajax和axios、fetch的区别](#)

[axios、fetch 和 ajax 等的区别详解](#)

16.vue2的filter

[Vue2.x之过滤器filter函数](#)

[vue filter的四种用法](#)

17.es6 比如promise ...

重复了 说明es6知识点的重要性

[阮一峰的es6学习](#)

18.es6 解构为什么出现 解决了什么问题

- 数据取值赋值
- 获取对象属性值
- 代码更简洁等 [阮一峰的es6学习](#)

19.v-if和v-show的区别

- 有无渲染
- 建议使用的场景

v-for+v-if 同时使用出现警告的原因等等

20.v-for的key的作用

diff算法的对比标识 加快diff

[快速、简洁讲明Vue中v-for循环key的作用](#)

21.diff算法和虚拟dom的一些知识

- 创建dom树
- 生成样式表
- 把dom树和样式表关联起来，生成Render树
- 布局layout
- paint 绘制

虚拟dom是一个js对象

[1.让虚拟DOM和DOM-diff不再成为你的绊脚石](#)

[2.虚拟dom](#)

22.路由的两种方式 路由守卫

hash history 使用 常用钩子 实现原理

[vue-router的两种模式的区别](#)

23.动态加载路由的实现

[动态加载路由的实现1](#)

[动态加载路由的实现2](#)

24.系统权限的实现和控制 (可以考虑一下项目登陆相关问题)

- 结合后端
- 纯前端
- 如何封装
- 不同角色 [vue中如何实现后台管理系统的权限控制](#)

25.watch和computed的区别

- 使用场景
- computed有缓存
- 都是vue watcher类

[搞懂computed和watch原理，减少使用场景思考时间](#)

26.说说你自己项目碰到的难点

结合自己项目 说几个 可以往自己擅长方向引 比如性能优化 数据埋点 或者工程化实践等等

27.[]==false 和 ![]==false

- 第一个 []==false 转为数字 0==0
- 第二个 ![]==false 转为布尔 false==false

可以思考一下显性转换和隐性转换

28.性能优化

巴拉巴拉 从代码 网络 资源加载 打包部署 等等层面去简单阐述 性能优化可说的地方太多

社区里就有很多优秀的性能优化文章(大佬)，建议收藏阅读

[1.前端性能优化 24 条建议 \(2020\)](#)

[2.写给中高级前端关于性能优化的9大策略和6大指标 | 网易四年实践](#)

[3.聊一聊前端性能优化](#)

[4.Vue 项目性能优化 — 实践指南 \(网上最全 / 详细\)](#)

思考：用户体验优化 比如白屏加载问题(骨架屏)

[首页白屏优化实践](#)

29.router守卫和系统权限 对应23、24

- 路由封装、懒加载等等
- 系统权限

30.vue和react的相同点和不同点 两个问题

相同：

- 都使用了虚拟dom、
- 都有ssr、
- 都支持jsx、
- 性能好、
- 响应式、

- 数据驱动、
- 将注意力集中保持在核心库，同时也关注路由和负责处理全局状态管理的辅助库。 区别：
- vue使用于小项目、
- vue学习成本低、
- vue渲染更快、 -vue有模板而react是jsx、
- vue双向数据绑定，
- react是单向、
- vue有指令语法、

小结：react是all in js 和vue是 all in .vue

[个人理解Vue和React区别](#)

[关于Vue和React的一些对比及个人思考（上）](#)

[关于Vue和React的一些对比及个人思考中）](#)

31.webpack打包图片的方式

base64 和 地址引入

[我学习webpack的一个csnd作者](#)

[前端图片最优化压缩方案](#)

32.content-type的类型有哪些

- application/json
- multipart/form-data
- raw
- binary
- plain/text
- ...

[http Content-Type 知多少](#)

33.http请求的方法有哪些

- get
- post
- head
- put
- delete
- connect
- trace
- options

GET 和 POST 有什么区别？

34.script标签的async和defer

- async 表示异步
- defer 表示延迟

图解 script 标签中的 async 和 defer 属性

35.高阶函数和偏函数、柯里化

参数是函数的函数为高阶函数

返回函数为偏函数

36.async Generator promise区别 都是异步解决方案

- async/await为Generator的语法糖。
- Generator yield分步调用，then()去持续调用
- async/await返回的是一个promise对象，所以可以在后面写then();

37.怎么判断浏览器是否支持es6

可以使用一个es6语法 看看是否抛出错误

38.form和json等等post传参的区别

content-type类型

GET 和 POST 有什么区别？

39.vue的data为什么是函数返回而不是直接一个对象

组件的data如果直接返回一个对象，那多次引入该组件，他们的data的映射地址是一样的，造成了互相污染，而如果是函数返回一个对象，就是每次注册组件都是生成一个新的对象，他们的映射地址是不同的

data属性为什么不能是一个对象

40.hock:生命周期

```
this.$once('hook:beforeDestroy',()=>{ })
```

41：怎么判断登录用户 是否有该页面权限 token过期 等等知识

又是权限相关 很容易被问

42.路由守卫的函数再过一遍 比如beforeEach

重复出现的问题

43.defineProperty的属性值有哪些

- configurable 是否可以删除属性和属性描述
- enumerable 才能出现在对象枚举中
- value 初始值
- writable 是否能被赋值运算符改变
- get
- set

[扩展链接](#)

手写题：一个obj={a:0}每次取属性a 值+1

简单实现：外面增加一个记录值

44.vuex中mutations与actions的使用及区别

- Action 提交的是 mutation，而不是直接变更状态。
- Action 可以包含任意异步操作。

[vuex官网](#)

45.observe observer dep watcher、、、vue2的原理

vue2原理学习

(鲨鱼哥的Vue2源码系列专栏)[juejin.cn/column/6961...]

46.webpack 的loader

use test loader exclude

区别于plugins

[手把手教你撸一个 Webpack Loader](#)

47.ts的readOnly ? public

ts知识、

[还不会TS? 带你 TypeScript 快速入门](#)

48.nextTick原理

dom更新之后执行的回调

可以用来获取更新后的dom元素

[Vue中\\$nextTick源码解析](#)

49.axios拦截器

interceptor

[聊聊 Vue 中 axios 的封装](#)

50.extends和vue.use mixins混入

插件要有install才可以use

区分 extends:... 和 vue.extend

- extend只能导入一个对象而mixins可以一个数组多对象
- `vue.extend()` 方法其实是 `vue` 的一个构造器，继承自 `vue`
- 可以通过 `extent` 拓展全局组件

51.vue的init() 原理

beforeCreate init了events和生命周期

[30 道 Vue 面试题，内含详细讲解](#)

52.vue的component 全局注册和局部注册有什么区别

- 全局组件: `Vue.component()`
- 局部组件: `components:`

53.项目的api和router封装看看

重复题

54.eventbus手动实现

[如何自己实现一个 EventBus](#)

55.虚拟dom的一些思考

56.vite快在哪 vite优点

- webpack使用require注册依赖
- vite使用了ESM注册依赖
- vite打包体积更小 vite更快 vite配置更简单

57.defineProperty和proxy的区别

- defineProperty是劫持对象的数据
- 而proxy是整个对象

58.react和vue的区别

重复：

- 相同点：都有虚拟dom 都有ssr 响应式 都支持jsx 都有对应的状态管理和路由管理库 都专注于自身库
- 不同：vue双向数据绑定 vue渲染更快 vue学习成本更低 vue中文文档较多 vue有template而react是jsx

59.setup data耦合相关知识

相关链接

60.js闭包的概念

- 一个函数和该函数所拥有的变量之和的环境
- 外部获取使用该变量 但是不能被外部直接改变 要使用该函数拥有的方法去调用

61.三次握手 和四次挥手 http相关概念1.1增加了keep-alive 常用状态码 比如301 302

http http1.0 http1.1 https http2.0等等（待加强）

[学习链接](#)

62.js为什么是单线程

js设计之初是为了加快客户端和服务器的访问，多线程的话，多个地方改变dom元素，为导致混乱，更难设计

63.h5的worker

- 这是h5的一个api
- 也是一个类
- new worker新增一个线程 但是这个线程有限制不能操作dom元素而且受主线程管理

64.eventLoop的宏任务和微任务

- 事件循环
- 任务队列
- 宏任务
- 微任务 [js中的宏任务与微任务](#)

（先执行宏任务再执行微任务）

65.pnpm npm yarn的区别

[学习链接](#)

66.webpack和vite对比 和webpack常用api (loader和plugins)

- webpack使用polyfill加载模块
- 而vite使用ESM加载模块
- vite速度更快 打包体积更小 易于配置
- loader加载器

67.vue3和vue2对比, vue3快在哪

- diff算法增加了一个静态标记 只对比有标记的dom元素
- 文本dom元素提升 只声明一次 render的时候可以多次使用不像之前vue2需要每次都重新声明渲染
- 事件增加了缓存
- 现在使用proxy直接获取属性而不是之前defineProperty劫持数据属性 要知道对应属性名字
- 使用了ssr更新渲染 把它们当做字符串快速更新渲染

扩展链接

68.缓存分为强缓存和协商缓存 区别于存储(cookie sessionStorage localStorage)

- 强缓存不需要向服务器发送请求
- 而协商缓存需要发送一个get请求 304

69.双向绑定原理

实现 然后不绑定value的话用model钩子改变

model钩子(props 、 event)

70.MVVM

[相关链接](#)

算法题（被问到的）

- 反转链表
- 合并两个有序链表（扩展到合并k个有序链表）



- 面试碰到了阿宝哥是面试官，当时挺紧张的
- 最长一次当次面试超过了100min，口干舌燥(佩服自己那么能bilibili说一大堆)
- 入职后找到帮忙内推的小伙伴 期待可以面基了



一般来说面试被问：

- html(涉及较少，被问过，没记住差不多)：如语义话标签或者非语义话标签有哪些
- css（涉及少）：如权重 布局
- js：涉及多，数据类型、原型链、闭包... 其中几个
- 框架：涉及多 vue或者react重点设计一个 使用和原理 或者一些设计思想
- 性能优化：一题 (如果你有这方面优势可以主动往这引)
- 计算机基础 重要，问的不一定多
- 服务端 node.js nginx等

- 工程化 ci/cd
- 算法(大厂必备吧)
- ...

非大厂准备js和框架 然后再有1-2点其他熟悉或者优势就没问题