**语法基础【笔试】**

1. 后疫情时代，一面都远程视频面试
2. 笔试时可以只写关键点，不必完整作答
3. 笔试主要考察语法基础，包括：**作用域、THIS 指向、异步执行顺序、原型链**
4. 看这篇文章就够了：[**送你 43 道 JavaScript 面试题**](https://juejin.im/post/5d0644976fb9a07ed064b0ca)

**浏览器里的事件循环**

1. 描述一下[**事件循环模型**](https://juejin.im/post/5b24b116e51d4558a65fdb70)**（宏任务、微任务）**

**VUE**

1. Vue 2.0 响应式原理：[**数据劫持Object.defineProperty() + 发布-订阅模式**](https://www.cnblogs.com/canfoo/p/6891868.html)

**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。**

1. Vue 3.0 响应式原理：**Proxy**
2. [2.0 和 3.0 的区别](https://juejin.cn/post/6940454764421316644)：
3. Composition API
4. 响应式原理不同
5. 生命周期钩子
6. 支持Fragment（多个根节点）
7. Teleport （任意门）
8. 对TS支持更友好
9. **V-for时为啥要绑定不同的 key？（vue的**[**diff 算法**](https://juejin.im/post/5affd01551882542c83301da)**）（不一定要掌握，但必问！）**

**在 Diff 算法中 会借助元素的 Key 值来判断该元素是新近创建的还是被移动而来的元素，从而减少不必要的元素重渲染。** [**https://zhuanlan.zhihu.com/p/281031340**](https://zhuanlan.zhihu.com/p/281031340)

1. [VUE 组件之间传值有哪些方法？](https://juejin.im/post/5d267dcdf265da1b957081a3)
2. VUE 在哪些情况下监控不到数据的变化？[为什么？](https://www.cnblogs.com/youhong/p/12173354.html)  
   1）Vue 不能检测对象属性的添加或删除  
   2）当你利用索引直接设置一个数组项时，例如：vm.items[indexOfItem] = newValue  
   3）当你修改数组的长度时，例如：vm.items.length = newLength
3. 啥时候使用 nextTick？（当我们需要在更新后的DOM中执行操作时，需要在nextTick的回调函数里执行一些操作）
4. [VUE 面试题大全](https://juejin.im/post/5d59f2a451882549be53b170)
5. Vue-router 的两种模式：**hash模式 和 history模式**

**ES6**

1. [箭头函数与普通函数的区别](https://juejin.im/post/5b14d0b4f265da6e60393680) 【必考】
2. 你使用过哪些 ES6 新特性？
3. Promise 中的 then 方法是同步的还是异步的？异步的
4. Promise 的异常处理，then 的第二个回调函数可以处理异常，为啥还需要 catch 方法？  
   **catch 可以捕获前面 then 方法执行中的错误,因此，建议总是使用 catch 方法，而不使用 then 方法的第二个参数。**
5. Object.assign()是深拷贝还是浅拷贝？浅拷贝
6. 深拷贝最简单的方法是什么？JSON.parse(JSON.stringify(obj))
7. 数组去重：[...new Set(arr)]
8. 数组拍平：arr.flat(Infinity)
9. CommonJS 和 es6 Module区别？ 前者传递值的拷贝，在执行时执行。后者传递值的引用，在解析时解析
10. script标签的属性 defer 和 async 区别？二者都是异步下载js脚本，前者在页面渲染完执行脚本，后者拿到脚本后立即执行

**HTTP 通信**

1. 200 from cache 和 304 有什么区别？
2. 401 和 403 有什么区别？401 需要身份认证、403 禁止访问
3. 跨域有哪几种解决方法？JSONP、CORS、nginx 反向代理、本地启动代理、PostMessage
4. 强缓存和协商缓存
5. CORS 解决跨域问题后，如何再解决携带 cookie 的问题？[参考这篇](https://www.cnblogs.com/sosohui/p/10273225.html)
6. JSONP 原理
7. https加解密原理：https://juejin.cn/post/6844904127420432391#heading-3

**服务器端**

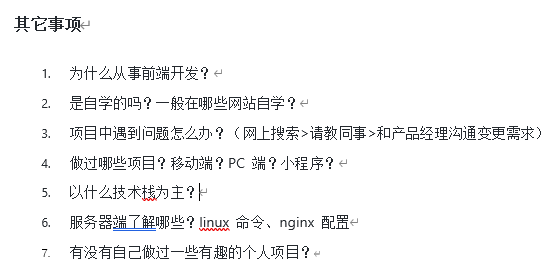
1. Nginx 服务器如何配置反向代理？
2. 是否熟练掌握 Linux 命令？（不熟练掌握也没事）

**样式布局**

1. Flex 布局
2. 浏览器的两种盒模型
3. px em rem % vw vh 这些单位的区别
4. 伪类和伪元素

伪类：:hover :focus :active :first-child :nth-of-type 修饰一些特殊状态

伪元素：:before :after :first-letter :first-line 凭空捏造一些元素

**性能优化**

1. 前端打包，拆包
2. 静态资源使用使用CDN
3. 开启gZip压缩
4. 设置缓存策略
5. 组件库按需引入
6. 节流和防抖
7. 图片压缩与懒加载
8. 打包拆包
9. 开启 Tree Shaking
10. 升级到http2，可以多路复用