

HTML

1. HTML语义化
2. viewport
3. html 1px、响应式布局
 1. html 1px 涉及到下边两个问题的概念
4. devicePixelsRatio
5. 布局视口、视觉视口、理想视口(最理想的视口就是设备宽度，单位为设备逻辑像素)
6. 元信息类标签
7. localStorage与sessionStorage
8. canvas 1px
9. DTD, document.documentElement与document.body, 盒子模型

CSS

1. 盒子模型
2. 选择器及优先级
3. 继承
4. 新增伪类
5. 水平垂直居中
6. display有哪些值及作用
7. 颜色转换
8. 媒体查询
9. 画三角
10. margin, margin: auto的含义 . visibility与display、opacity区别
11. visibility属性有个collapse属性值? 在不同浏览器下以后什么区别?
12. position跟display、overflow、float这些特性相互叠加后会怎么样?
 1. display属性规定元素应该生成的框的类型; position属性规定元素的定位类型; float属性是一种布局方式, 定义元素在哪个方向浮动。类似于优先级机制: **position: absolute/fixed优先级最高**, 有他们在时, float不起作用, display值需要调整。float 或者absolute定位的元素, 只能是块元素或表格。
13. **BFC及怎么触发**
14. 清除浮动
15. margin塌陷, margin合并
16. 设置元素浮动后, 该元素的display值是多少? 自动变成display: block
17. less
18. css优化
19. margin与padding
20. 如果需要手动写动画, 你认为最小时间间隔是多久, 为什么?
 1. 多数显示器默认频率是60Hz, 即1秒刷新60次, 所以理论上最小间隔为 $1/60 * 1000ms = 16.7ms$ 。
 2. requestAnimationFrame
21. css位置
22. flex、默认值 stretch 1
23. GPU加速渲染
24. postcss

25. 安全区域
26. iPhoneX安全区域、苹果刘海屏
27. min-width与max-width
28. RGBA、HSLA(色相(hue)-饱和度(saturation)-亮度(lightness)), 看掘金
29. z-index生效条件: position 属性值不是 static
30. link 和 @import

JavaScript

1. 基本数据类型
2. map、weakmap、set、weakset
 1. set add
 2. map set、get
3. this
4. 深拷贝与浅拷贝, 扩展运算符与浅拷贝
5. 事件冒泡与捕获及应用、事件委托
6. 继承、prototype、**proto**、constructor的关系
7. **事件循环**
8. 常用的promise方法、promise的优缺点
9. 设计模式
10. 柯里化与函数式编程 . Array的方法, ES6的方法, 以及原有的方法, 哪些改变原数组哪些不改变
11. 更换短链原理
12. webview、什么是jsbridge及原理
13. 跨域问题
14. 双等号与三等号
15. typeof instanceof Object.prototype.toString类型判断
16. null与undefined
17. symbol
18. generator
19. 闭包
20. && ||
21. defer与async
22. 重排与重绘, documentFragment
23. 4中for循环
24. map、reduce
25. console.log(Boolean(-1)); === true
26. 跨站与跨域
27. 解构
28. document.documentElement与document.body
29. DOM基本操作、nodelist与collection
30. 前端工程化
31. cors原理和实现, cors会不会发请求到服务器
32. es6和commonJS模块管理的区别
33. es5继承与es6继承的区别
34. Object.create()

JavaScript 手写

1. new、call、apply、bind
2. promise、promise.all
3. async await
4. 防抖与节流
5. vue双向绑定
6. 事件委托
7. 深拷贝
8. 柯里化
9. 树的遍历
10. 继承

算法

1. 双指针
2. 滑动窗口
3. 回溯法
4. 动态规划
5. 贪心
6. 堆栈
7. 树
8. 排序，稳定性，最好和最差的时间复杂度

Webpack

1. loader与plugin
2. chunk与bundle
3. source-map
4. chainWebpack与configureWebpack
5. webpack-merge
6. tapable
7. 打包优化
8. 手写打包器学习原理
9. babel与ast、polyfill

Vue

1. 双向绑定源码
2. nexttick源码
3. vue与react的区别
4. mvc mvp mvvm
5. v-if与v-show
6. computed、watch与methods
7. 虚拟DOM与diff
8. vue3
9. proxy与defineProperty的区别
10. keep-alive与activated . 路由懒加载
11. vue + ts中混入的使用

12. .sync
13. history与hash
14. 为什么vue必须要有根节点
 - 因为在diff里边是单节点操作，而不是数组操作，所以必须有一个根节点
15. vue2 不能监听什么的变化、监听数组变化的原理
16. 父子组件钩子函数执行顺序
17. scoped
18. Vue.extend

Typescript

1. 看一遍

Git

性能优化

1. 首屏加载
2. 性能优化指标
3. 减少请求次数
4. 加快请求速度
5. 加速页面的解析与处理
6. 自己的理解
 - 合理的使用防抖和节流函数，来减少对服务的查询
 - 对于多图的页面，使用图片的懒加载
 - 减少页面发生重排
7. 浏览器缓存原理及策略，在浏览器相关.md里边 1

网络

1. http基本结构
2. 状态码,302,303,307
3. 浏览器从输入url到渲染发生了什么
4. cookie包括samesite属性、session、token
5. https
6. XSS、CSRF
7. DNS重绑定攻击，为什么host可以解决
 1. 因为这个重绑定攻击是诱导用户去访问恶意网站，然后更改恶意网站的ip来实现的，那么被攻击的服务器收到的host的值就是恶意网站的值，如果我们在服务器设置白名单，那么恶意网站一定不在白名单里边，所以可以防止重绑定攻击
8. 三次握手、四次挥手及各个阶段的字段
9. 强缓存与协商缓存
10. http 2.0 . tcp与udp
11. get与post区别
12. DNS查询过程
13. 七层与四层网络模型
14. localStorage与sessionStorage

浏览器

1. 浏览器的组成
2. 浏览器内核一般是指浏览器的渲染进程
3. 浏览器加载流程
4. 如何减少和避免重排

其他

1. 进程与线程的区别
2. 进程间通讯的方式
3. Git
4. [webview](#)相当于一个浏览器，在手机上用webview和在电脑上打开chrome类似
5. 开发思路
6. 介绍
7. d3
8. ast

亮点与难点

1. 组件的封装与复用
2. 考虑问题的全面性
3. 所学知识的应用
4. 沟通与联调
5. 上方购买按钮消失显示下方的bottombar