

# 第七部分：综合问题版

这里不做解答，请自行查阅相关答案。

## 一、基础

### 1.1 HTML

- `html5` 新标签有哪些
- `canvas` 、 `svg` 、 `webGL`
- 你是如何理解 `HTML` 语义化的，有什么好处
- 前端需要注意哪些 `SEO` ？
- `manifest` 、 `worker` 、 `socket`
- `input` 和 `textarea` 的区别
- 用一个 `div` 模拟 `textarea` 的实现

### 1.2 CSS

- `css3` 有什么新特性，浏览器支持怎么样
- 伪类是什么？有哪些？会有哪些兼容性问题？如何处理？
- `css` 预处理器知道吗？用过哪些？有什么优劣？后处理器知道吗？
- 盒模型有哪几种？怪异模式和标准模式？
- `less` 、 `sass` 、 `postcss` 、 `prefix`
- 层叠优先级
- 圣杯、双飞燕布局
- `float` 清除浮动
- `flex` 布局、 `grid` 布局、 `table` 布局
- `css` 以及中轴旋转、动画变换
- 绘制三角形、矩形、菱形、梯形（奇巧淫技，可以不问）
- `sprite` 图（雪碧图）知道吗？ `svg` 雪碧图了解吗？
- `px` 、 `em` 、 `rem` 、 `vw` 、 `vh` ？ `rem` 的根节点样式在什么时候设置？
- `position` 有哪些？他们的定位原点是什么？
- 媒体查询用 `css` 好还是用 `js` 好？

- `link` 和 `@import` 的区别？
- 响应式布局的原理
- `css` 低版本浏览器兼容问题，额外需要什么后缀来声明浏览器兼容
- `!important` 意义，是否应当规避使用？
- `BFC` 块级上下文、`IFC`，实现双栏高度对齐
- `BEM` 命名法，有什么优势，有什么劣势
- `1px` 边框问题
- （水平）居中有哪几种实现方式、（垂直）居中有哪几种实现方式

## 1.3 JavaScript

- `typeof` 以及弱类型转换规则？`NaN`、`undefined`、`null`
- `dom` 的节点操作？能够背 `api` 还是知道 `api`？
- `ajax` 是什么？知道底层实现原理吗？知道 `fetch` 吗？自己封装过吗？
- `GET`、`POST` 意义？`restful` 架构下还有别的什么请求？`OPTION` 请求是什么？
- 事件冒泡和事件捕获是怎样的？对应的默认方法有什么？一般在什么情况使用？
- `call`、`apply`、`bind`
- 如何判断数据类型？
- `hoisting` 是什么？具体表现是怎样的？
- 匿名函数是什么？函数表达式和函数声明的区别？
- `let`、`const` 暂时性锁区知道吗？表现是怎样的？
- 严格模式是什么？有什么好处？`use strict`
- `arguments` 是什么类型？`callee` 和 `caller` 有什么区别？
- `Date.format` 实现过吗？思路是怎样的？
- 动画：`setTimeout` 何时执行，`requestAnimationFrame` 的优点
- 你知道 `new` 一个对象有几步吗？

## 二、高级

---

### 2.1 Browser

- 打开一个网页经历了那些过程？
- 浏览器加载白屏是什么原因？
- 千万访问量的项目，前端需要注意些什么？

### 2.2 JavaScript

- 表达式和语句有什么区别？如何把语句转换为表达式？

- 什么叫执行上下文栈(·Execution Context Stack)·? 一个函数调用会产生多少个上下文环境? 如何激活一个上下文, 什么叫caller(调用者), 什么叫callee(被调用者)? 给你一段代码能否画出执行过程中的上下文堆栈变化?
- 执行上下文包括哪些结构(状态/属性), 如何追踪关联代码的执行进度?
- `eval` 在调用的时候有哪些特别的地方? `eval` 函数自身会产生上下文吗? 会影响当前的调用上下文吗?
- 什么叫变量对象? 什么叫活动对象?
- 词法作用域是什么? 闭包是如何形成的?
- `var foo = function bar () {}` 命名函数表达式中(上述的foo函数)bar变量是定义在哪一层作用域的?
- `(0, 1, 2)` 的结果是什么?
- `var foo = { value: 2, bar: function () { return this.value; } }` 中`(foo.bar, foo.bar)()` 的 `this` 值是什么? `(foo.bar = foo.bar)()`、`(false || foo.bar)()` 呢?

## 2.3 HTTP

- `http1.0` 和 `http1.1` 的区别
- `http` 请求码有哪些? `206` 、 `302` 、 `303` 、 `304` 、 `307`
- 能说下 `304` 具体怎样实现吗?
- `http` 缓存逻辑是怎样的? 协商缓存与强缓存? `Last-Modified` / `Etag` / `Expires` / `Cache-Control` ?
- `osi` 七层协议和 `tcp/ip` 四层协议
- 三次握手和四次握手
- 跨域是什么? `http`协议中如何判断跨域? 如何解决跨域问题?
- `http2` 具体内容? `SDPY` 了解么?
- `HTTPS` 如何实现? `ssl/tls` 是什么? 对称加密、非对称加密在什么时候、对什么数据加密?
- `DNS` 劫持是什么?
- 浏览器在一次 `HTTP` 请求中, 需要传输一个 `4097` 字节的文本数据给服务端, 可以采用那些方式?

## 2.4 优化

- 首屏优化
- 预加载资源、预解析 `DNS`
- 懒加载图片怎么做
- `iframe` 一般怎么使用?

- `window.onload` 、 `document.ready` 顺序？ `iframe` 会阻塞吗？
- `cdn` 是什么？如何查找最近的 `cdn` ？
- `gzip` 、 `chucked`
- 前端安全： `xss` 攻击和防范、 `CSRF` 、 `CORS`
- 硬件加速怎样开启
- 优化中会提到缓存的问题，问：静态资源或者接口等如何做缓存优化
- 页面 `DOM` 节点太多，会出现什么问题？如何优化？

## 2.5 Products

这里主要是一些生产中真实存在的问题

- 一个 `feed` 流，有哪些展现方式？如果当前页面可视区有5个格子，而你有 `20` 条信息需要展示，请问有哪些自动展示的算法？
- 移动端的缓存机制有哪几种？
- 小程序如何与 `native` 通信？ `react-native` 实现原理？ `jsbridge` 如何实现？
- 过万条数据如何加载和渲染？
- 如何在前端解析二进制，流媒体、图片二进制数据怎样渲染到页面上？
- 如何解析二进制音频、视频？
- 文件如何显示上传百分比？
- 如何制作一个富文本，需要考虑哪些结构？
- `html5` 的播放器怎么做？视频直播如何在浏览器里面实现？ `HLS` 、 `RTMP`
- 如何在浏览器里面裁剪图片？如何裁剪视频，对视频做逐帧分析？
- 如何实现一个具有引导功能的组件库？
- 如何封装一个 `Form` 和 `FormItem` ，使其能够跟 `input` 、 `select` 、 `checkbox` 、 `radio` 等组件进行 数据存储、数据校验（自定义逻辑）、校验反馈？
- `echart` 这类图像库的实现原理？
- `i18n` 的国际化方案应该是怎样的？
- 数据埋点的意义是什么？应当针对哪些数据进行埋点？如何构建一个埋点系统？
- 小程序的实现原理是什么？这【五篇】文章将带你深入了解「微信小程序」🔗

## 三、Framework

### 3.1 React

- `React` 的生命周期 `mount` 和 `update` 描述下
- `React` 的生命周期中的 `isBatchingUpdates` 了解吗？ `Transaction` 知道吗

- **React** 的 **vdome** 如何实现？**jsx** 是怎样解析的？
- **React** 的 **Fiber** 是什么？具有什么样的特性？
- **React** 的 **diff/patch** 算法原理
- **React** 的组件逻辑（受控、非受控）？如何设计一个组件库
- **React** 的数据流，**Redux**、**Mobx**、**Rxjs**，发布订阅模式、观察者模式，**flux** 和 **no-flux**
- **React** 的事件注册和事件分发知道吗？
- **Redux** 解决了什么痛点（有什么优点），又有什么缺点
- **Redux** 的中间件有哪些？**redux-actions**、**redux-promise**、**redux-thunk**、**redux-saga**、**redux-immutable** 了解过哪些？说说中间件的意义
- **Redux** 有什么优化方案？
- **SSR** 了解过吗？怎样做？了解 **Koa** 么？
- **React-Native** 了解过吗？**JavascriptCore** 是什么？

## 3.2 Vue

- **Vue** 是如何设计响应式系统的？（依赖收集）
- 什么时候使用 **computed**，**methods**，**watch**？之间有什么样区别？
- **template** 和 **render(jsx)** 有什么的联系？
- **this.\$nextTick** 是如何设计的？
- **Vue** 组件 **data** 为什么必须是函数？

## 3.3 AngularJS

尽管angularJS已经基本退出了历史舞台，但是相信有的同学还是做过相关的项目，并很有可能被面试官问起

- **angularJS** 的数据绑定采用什么机制？详述原理
- 如果通过 **angularJS** 的 **directive/component** 规划一套全组件化体系，可能遇到哪些挑战？
- 一个 **angularJS** 应用应当如何良好地分层？
- **ng-click** 中写的表达式，能使用JS原生对象上的方法，比如 **Math.max** 之类的吗？为什么

## 四、文章

- 总结了17年初到18年初百场前端面试的面试经验(含答案) [🔗](#)

- [前端基础面试整理\(一\)](#)