## JS 的定时器

分有两种，一种是setTimeout单次使用，setInterval周期调用，用配套的clean结束。A+B,A+2B,这样的主要是用递归+改对象里面的值，对线里面存有调用setTimeout的返回值用于结束

## 防抖与节流

防抖：n秒内函数只会执行一次，如果重复执行则会重新计时。

检测Timer的值是否为null

节流：n秒内只会执行一次

加一个标记位，每一次执行之后才能改。

## 闭包

外部可以访问函数内部的变量。1.读取函数作用域中的而变量2.避免变量被污染

闭包的回收，对象不再被引用。

应用：1.模块封装2循环创建闭包

## Lodash

Js库，便于处理数组等内容的小插件。

## 函数柯里化

curry的概念很简单：只传递给函数一部分参数来调用它，让它返回一个函数去处理剩下的参数。

<https://blog.csdn.net/weixin_44166439/article/details/117324174>

分有两种一种是参数的数量是固定的 第二种是不固定的。

1. 到达参数固定的时候执行 递归
2. 没有参数的时候执行 递归并收集参数

<https://blog.csdn.net/qq_31967569/article/details/82458540>

用链式操作的方式也可以达到类似的效果

## 链式操作

<https://blog.csdn.net/qq_31967569/article/details/82458540>

闭包+递归返回函数+valueof

还是就是返回this

## 类数组与数组

一个是array一个不是，就只是个普通的对象。不能够直接调用数组的方法。

Arrayform可以转换

## 面对对象的内容

对象是无序属性的集合

## Promise的内容+promiseall

https://es6.ruanyifeng.com/#docs/promise

异步编程，

<https://www.jianshu.com/p/ffce41377ac1>

## 事件循环

<https://segmentfault.com/a/1190000015317434>

宏任务与微任务

https://www.mianshiya.com/qd/bf4a0bf261c7e2500090d9482499675f

## 手写数组转树

<https://www.jianshu.com/p/a5be31c9eec5>

主要是递归的内容

## 手写用 ES6proxy 如何实现 arr[-1] 的访问

<https://es6.ruanyifeng.com/#docs/proxy>

## 大数计算如何实现

用字符串表示的话方式大概就是位数补齐，然后再单个字符的相加

用自带的函数的话就是BIGINT来解决。

## 深拷贝与浅拷贝

深拷贝是把数据内容全都重新拷贝一份，浅拷贝是把内存地址给复制了。

## 箭头函数

<https://es6.ruanyifeng.com/#docs/function>

https://www.mianshiya.com/qd/17e3426e61c802f300988a0d0fc94cb2

## dom操作

红宝书 270页左右开始的内容。

主要是选择符方面才是内容的重点。

## let var const

let 是类var 1.不存在变量提升 2.声明仅再块级作用域有效 3暂时性死区 4不允许重复声明 块级作用域

const 只读的常量 声明后的值就不能改变

全局变量和顶层对象的属性（window）

## 垃圾回收

红宝书 78页的内容。本质上是找出不再使用的变量，释放其内存。

1.标记清除2引用计数 引用技术会有循环引用导致的不会记数清零的问题解决方式是不使用的时候手动至null

<https://blog.csdn.net/weixin_46022982/article/details/117571968>

## import与require

<https://es6.ruanyifeng.com/#docs/module>

ES6里面有相关的 模块机制module

Commonjs和AMD两种前面一个是用于服务器上面的 后面一个用于浏览器上面

ES6的export是输出的接口是一种动态绑定的关系。CommonJ是出书值得缓冲。Export本质上是输出一个{}对象。Export default是输出匿名对象

<https://blog.csdn.net/MFWSCQ/article/details/108365307>

## 前端性能优化

## Async

用于异步操作，

<https://es6.ruanyifeng.com/#docs/async>

## 22.改this指向

Call apply bind

## 23.说一下module.exports和exports的区别，export和export default的区别

ES6 和commonjs模块加载的区别

## 24.Number的精度缺失

## 25.promise的链式调用

<https://blog.csdn.net/vivo_tech/article/details/105576692>

<https://www.jianshu.com/p/cb065fd8142b>

## 26. 如何把真实 dom 转变为虚拟 dom，代码实现一下

## 27. 介绍事件冒泡、事件代理、事件捕获，以及它们的关系

冒泡和捕获是 事件流动的方向 事件代理是将子节点的事件处理放到父节点上，通过事件的特征来判断子节点的。

## 28. for..of 和 for...in 是否可以直接遍历对象，为什么

Of不可以，没有interact接口

## 29. 在 map 中和 for 中调用异步函数的区别

Map是异步的 for是同步的

## 30. es6 类继承中 super 的作用

## 31. Object.defineProperty 与 Proxy

https://blog.csdn.net/qq\_42833001/article/details/83302149

## 32.ajax的步骤

红宝书P571

XMLHttpRequest

## 33.前端的本地存储cookie localstorage sessionstorage

<https://www.jianshu.com/p/47717c55381d>

对于cookie的操作：document.cookie来实现 删除和添加通过更改时间来实现。expires

## 34.跨域的问题

红宝书P582页

协议 域名 端口号

默认情况下只能访问与包含XHR的页面同一个域的资源。

CORS：自定义的header头让浏览器与服务器进行沟通。

1. 图像Ping 图像jpg后跟查询字符串
2. JSONP 回调函数与数据 用需要跨域的文件提供回调函数，在被调用的文件上执行回调函数返回相关的数据

<https://www.cnblogs.com/dowinning/archive/2012/04/19/json-jsonp-jquery.html>

1. Comet

## 35. 输入URL到页面加载显示完成发生了什么

<https://www.mianshiya.com/qd/54ad1eea61c8972000bf542b4dc906c6>

1. 缓存
2. DNS解析 域名->ip
3. TCP连接 三次握手
4. http请求
5. 服务器返回数据给浏览器
6. 浏览器解析数据

## 36.数据类型

1.Number String Boolean null undefined

2.Object Array function

## 37. 重排和重绘

减少DOM的访问和尽量缓存局部变量

DOM树和渲染树 DOM：页面结构 渲染树：DOM结点的如何显示

DOM的变换影响了元素的几何属性（宽和高）浏览器要重新计算元素的几何属性，其他的元素也会受到影响，故需要重新构造渲染树。这个叫重排。

重排之后，重新绘制受影响的部分到屏幕，这个叫重绘。

重排必然导致重绘

1添加或者删除可见的DOM元素

2元素位置改变

3元素尺寸改变

4元素内容改变（例如：一个文本被另一个不同尺寸的图片替代）

5页面渲染初始化（这个无法避免）

6浏览器窗口尺寸改变

## 38.this

关于对象的this，用that来操作的情况，因为函数的this会被绑定到全局对象上，在对象内用that代替对象的this来避免这个情况发生。

## 39.网站攻击

<https://www.mianshiya.com/qd/54ad1eea61c7e7d800a2a9056d0204b7>

xss，通过代码注入的方式来获取信息 cookie和token

csrf，模拟用户操作，不会获取信息，但因为cookie是自动的 而token是需要手动弄到header上的。

## 40.MVVM

<https://www.jianshu.com/p/6aeeecd64dcf>

## 41. cookie和session的区别

<https://www.cnblogs.com/l199616j/p/11195667.html>

## 42.token

<https://blog.csdn.net/weixin_30934229/article/details/112828514>

## 43. 浏览器和 Node 事件循环的区别

## 44.cookie防范xss攻击

<https://blog.csdn.net/qq_43263320/article/details/114413239>

httponly

secure https