## let const var的区别

var声明变量可以重复声明，而let不可以重复声明

var是不受限于块级的，而let是受限于块级

var会与window相映射（会挂一个属性），而let不与window相映射

var可以在声明的上面访问变量，而let有暂存死区，在声明的上面访问变量会报错

const声明之后必须赋值，否则会报错

const定义不可变的量，改变了就会报错

const和let一样不会与window相映射、支持块级作用域、在声明的上面访问变量会报错

let一般应用于基本数据类型，定义的变量不会预解析，必须先声明再使用，不能定义已经定义的变量，块级作用域

const一般是引用数据类型，定义的变量不会预解析，必须先声明再使用，否则会报错，定义的变量不允许被修改，在数组中可以，因为const存储的是地址，值的内容可以变化

## es6新增判断数组的方法：Array.isArray()

## es6中set和map方法：

ES6 提供了新的数据结构 Set。它类似于数组，但是成员的值都是唯一的，没有重复的值。Set 本身是一个构造函数，用来生成 Set 数据结构。

ES6 提供了 Map 数据结构。它类似于对象，也是键值对的集合，但是“键”的范围不限于字符串，各种类型的值（包括对象）都可以当作键

## ES6新增数组方法

forEach filter map every some reduce includes

## map和forEach区别

forEach()方法不会返回执行结果，而是undefined。也就是说，forEach()会修改原来的数组。 map()方法会得到一个新的数组并返回。

## 箭头函数的this指向

由于箭头函数不绑定this， 它会捕获其所在（即定义的位置）上下文的this值， 作为自己的this值

1.

1. 所以 call() / apply() / bind() 方法对于箭头函数来说只是传入参数，对它的 this 毫无影响。
2. 2.
3. 考虑到 this 是词法层面上的，严格模式中与 this 相关的规则都将被忽略 作为方法的箭头函数this指向全局window对象，而普通函数则指向调用它的对象

## es6中的promise

Promise是异步编程的一种解决方案，比传统的解决方案（回调函数和事件）更合理更强大。

Promise对象有以下2个特点：

1.对象的状态不受外界影响。Promise对象代表一个异步操作，有三种状态：Pending(进行中)、Resolved(已完成)和Rejected(已失败)。只有异步操作的结果，可以决定当前是哪一种状态，任何其他操作都无法改变这个状态。这也是Promise这个名字的由来，它的英语意思就是“承诺”，表示其他手段无法改变。

2.一旦状态改变，就不会再变，任何时候都可以得到这个结果。Promise对象的状态改变，只有两种可能：从Pending变为Resolved；从Pending变为Rejected。只要这两种情况发生，状态就凝固了，不会再变了，会一直保持这个结果。就算改变已经发生了，你再对Promise对象田静回调函数，也会立即得到这个结果。这与事件(Event)完全不同，事件的特点是，如果你错过了它，再去监听，是得不到结果的。

## 说下promise

简单来说可以把promise当作一个装着异步操作结果的容器。从语法上说，Promise 是一个对象，从它可以获取异步操作的消息。Promise 提供统一的API，各种异步操作都可以用同样的方法进行处理。它将异步函数以同步的方式书写，也解决了回调地狱问题

特点：（1）对象状态不受外界影响

（2）一旦状态改变，就不会再变，任何时候都可以得到这个结果

缺点：（1），无法取消promise，一旦新建它就会立即执行，无法中途取消

（2），如果不设置回调函数，promise内部抛出的错误，不会反应到外部

（3）无法得知目前进展到哪一个阶段（刚刚开始还是即将结束）

三个状态：进行中、已成功、以失败

## async和await

async、await 优缺点

async 和 await 相比直接使用 Promise 来说，优势在于处理 then 的调用链，能够更清晰准确的写出代码。

缺点在于滥用 await 可能会导致性能问题，因为 await 会阻塞代码，也许之后的异步代码并不依赖于前者，但仍然需要等待前者完成，导致代码失去了并发性。

## async和await与promise的区别

1、Promise本身是同步的立即执行函数， 当在executor中执行resolve或者reject的时候, 此时是异步操作， 会先执行then/catch等，当主栈完成后，才会去调用resolve/reject中存放的方法执行，打印p的时候，是打印的返回结果。

2、async 函数返回一个 Promise 对象，当函数执行的时候，一旦遇到 await 就会先返回，等到触发的异步操作完成，再执行函数体内后面的语句。可以理解为，是让出了线程，跳出了 async 函数体。

3、await的含义为等待，也就是 async 函数需要等待await后的函数执行完成并且有了返回结果（Promise对象）之后，才能继续执行下面的代码。await通过返回一个Promise对象来实现同步的效果。

async和await实际上是建立在promise之上的

## symbol应用场景

1.使用symbol作为对象属性名

2.可以代替常量