### 1：行内元素有哪些，块级元素有哪些？

行内元素：a,span,i,b,small,em,strong,img,br,button,input,label select,textarea任一几个就行

块级元素：div,h1,h2,h3,h4,h5,h6,table,ul,ol,dl,form

### 2：src与href的区别?

src：指向外部资源的位置，指向的内容将会嵌入到文档中当前标签所在位置；在请求src资源时会将其指向的资源下载并应用到文档内

href：指向网络资源所在位置，建立和当前元素（锚点）或当前文档（链接）之间的链接，用于超链接，当浏览器解析到该元素时，会暂停其他资源的下载和处理，直到将该资源加载、编译、执行完毕

3：px和em的区别？

两者都是代表单位

px：值是固定的，绝对单位

em:值不是固定的，相对单位，跟父级单位设定或者浏览器默认字体设置存在关联

### 4：cookies，sessionStorage和localStorage的区别？

共同点：用于浏览器端存储的缓存数据

不同点：

1：存储内容是否发送到服务器端：当设置了Cookie后，数据会发送到服务器端，造成一定的宽带浪费。 sessionStorage和localStorage会将数据保存到本地，不会造成宽带浪费

2：数据存储大小不同：Cookie数据不能超过4K,适用于会话标识sessionStorage和localStorage数据存储可以达到5M

3：数据存储的有效期限不同：cookie只在设置了Cookid过期时间之前一直有效，即使关闭窗口或者浏览器，sessionStorage,仅在关闭浏览器之前有效；localStorage,数据存储永久有效

4：作用域不同：cookie和localStorage是在同源同窗口中都是共享的，sessionStorage不在不同的浏览器窗口中共享，即使是同一个页面；

5：sessionStorage和localStorage有更多丰富易用的接口

5：编写一个方法 去掉一个数组的重复元素？

1：常用方法(主要利用object的key不可重复性）

(**function** () {  
 **var** arr = [0, 2, 3, 4, 4, 0, 2],  
 obj = {},  
 temp = [],  
 len = arr.length;  
  
 **for** (**var** i = 0; i < len; i++) {  
 **if** (!obj[arr[i]]) {  
 obj[arr[i]] = **true**;  
 temp.push(arr[i]);  
 }  
 }  
 console.log(temp);  
})();

2：可以利用数组的indexOf方法和filter方法实现

6：webpack特性？

1：对 CommonJS 、 AMD 、ES6的语法做了兼容

2：对js、css、图片等资源文件都支持打包

3：串联式模块加载器以及插件机制，让其具有更好的灵活性和扩展性

4：有独立的配置文件webpack.config.js

5：可以将代码切割成不同的chunk，实现按需加载，降低了初始化时间

6：支持 SourceUrls 和 SourceMaps，易于调试

7：具有强大的Plugin接口，大多是内部插件，使用起来比较灵活

8：webpack 使用异步 IO 并具有多级缓存。这使得 webpack 很快且在增量编译上更加快

7：CSS隐藏元素的几种方法？

opacity:元素本身依然占据它自己的位置并对网页的布局起作用。它也将响应用户交互

visibility:与 opacity 唯一不同的是它不会响应任何用户交互

display 设为 none 任何对该元素直接打用户交互操作都不可能生效。此外，读屏软件也不会读到元素的内容。这种方式产生的效果就像元素完全不存在

8:页面导入样式时，使用link和@import有什么区别？

1：link属于HTML标签，无兼容问题，而@import是CSS提供的，只在IE5以上才能识别;

2：页面被加载的时，link会同时被加载，而@import被引用的CSS会等到引用它的CSS文件被加载完再加载;

3： link方式的样式的权重 高于@import的权重.

9:javascript的typeof返回哪些数据类型？

Object、number、function、boolean、underfind

10：call（）和apply（）的区别？

1：表现方式一个在url中可以看到相关参数，一个是看不到的

2：发送数据大小方面get数据有大小限制

3：安全方面一个是一般请求，一个涉及到保密性，post方式不会直接暴露，相对更加安全一点

11：call（）和apply（）的区别？

在于参数的不同，call可以接受多个参数，apply只接受2个参数，apply的第二个参数只能为数组，call没限制

12：事件委托是什么？

利用事件冒泡的原理，让自己的所触发的事件，冒泡到他的父元素代替执行

13：position默认值为？

static

14:split() join() 的功能？

split是分割字符串，join是将数组链接成字符串

15：”==”和“===”的区别

==：值相等就行，不严格判断

===：值相等且对象是同一个对象，严格相等

### 16：angularjs中ng-show/ng-hide 与 ng-if的区别

1：ng-show/ng-hide实际上是通过display来进行隐藏和显示的，隐藏状态时也实际存在于dom中

2：ng-if实际上控制dom节点的增删除来实现的，条件为假时不存在于dom中

### 17：vue中is特性为了解决什么问题？

为了解决在某些特定的标签中只能存在或嵌套指定标签，比如 ul嵌套li，select 嵌套option,table嵌套tr等等

18：哪些操作可能会造成内存泄漏？

1:递归调用没有设置退出条件

2：大量使用闭包

3：大消耗操作

4：频繁大量操作dom文档

5：没有及时回收相关变量

#### 19：vue内容分发实现？

通过<slot></slot> 标签指定内容展示区域

20：

1：原型模式，原型继承机制

2：工厂模式，Object（）构造不同对象

3：混入模式，object设置不同属性

4：立即执行模式，隔离变量且初始化执行

21：javascript使用严格模式的理由

1：消除javascript语法的一些不合理、不严谨之处，减少一些怪异行为

2：消除代码运行的一些不安全之处，保证代码运行的安全

3：提高编译器效率，增加运行速度

4：为未来新版本的Javascript做好铺垫

22:箭头函数的优点

1：**箭头函数表达式**的语法比[函数表达式](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/function)更短

2：不绑定自己的[this](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/this)，[arguments](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Functions/arguments)，[super](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/super)或 [new.target](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/new.target)

23：说出几个数组常用方法？

1：[concat](mk:@MSITStore:C:\\Users\\Administrator\\Desktop\\W3School.chm::/www.w3school.com.cn/jsref/jsref_concat_array.asp)：连接两个或更多的数组，并返回结果

2：[pop：](mk:@MSITStore:C:\\Users\\Administrator\\Desktop\\W3School.chm::/www.w3school.com.cn/jsref/jsref_pop.asp)删除并返回数组的最后一个元素

3：push:向数组的末尾添加一个或更多元素，并返回新的长度

4:[reverse](mk:@MSITStore:C:\\Users\\Administrator\\Desktop\\W3School.chm::/www.w3school.com.cn/jsref/jsref_reverse.asp):颠倒(反向)数组中元素的顺序

5：[shift：](mk:@MSITStore:C:\\Users\\Administrator\\Desktop\\W3School.chm::/www.w3school.com.cn/jsref/jsref_shift.asp)删除并返回数组的第一个元素

6：unshift:向数组的开头添加一个或更多元素，并返回新的长度

7:slice:从某个已有的数组返回选定的元素

8:sort:对数组的元素进行排序

9:splice:删除元素（替换数组），并向数组添加新元素