111、Jquery的好处

1、少写代码多做事

2、免费开源轻量级

3、兼容主流浏览器

4、强大的选择器

5、文档手册很详细

6、插件成熟供选择

112、本地存储和离线存储

区别：

本地存储存的是数据，离线存储存的是网页

本地存储主要是将网络会话过程中的数据保存在客户端，突破cookie存储大小的限制，相当于针对前端页面的数据库。

离线存储存的是没有网络依然可以缓存访问的网页，减少服务器负载

**localStorage用法：**

localStorage.setItem(“key”,”value”) //存储

localStorage.getItem(key) //按key取值

localStorage.removeItems(key) //按key单个删除

localStorage..clear() //删除全部数据

localStorage.length //获得数据的数量

localStorage.valueOf() //获取全部值

**离线缓存的用法：**

1. 在html标签添加manifest属性，属性值为manifest文件的路径
2. manifest会告知浏览器需要缓存和不缓存的内容，可分为三部分

<1>、CACHE MANIFEST -此标题下的文件会在首次连接后缓存

<2>、NETWORK 此标题下需要与服务器连接并且不会被缓存

<3>、FALLBACK 当页面无法访问时的回退页面

113、json的优缺点

json是一种轻量级的数据交换格式，具有良好的可读性与便于快速编写的特性

json支持的数据类型：

1、number

2、boolean

3、string

4、null

5、array

6、object 即{……}

json的优缺点：

1、数据格式简单，易于读写，格式是压缩的，占用带宽小

2、支持多种语言，编码简单

3、解码难度比XML简单

4、json和js交互更方便

5、json速度远快于XML

115、jquery有几种类型选择器 8类

**一、基本选择器**

1、ID选择器，返回单个元素

2、类选择器 class

3、标签选择器 element

4、\*通配符选择器

5、selectorN，并集选择器，$(p,span,p/myClass)

**二、层次选择器**

1、后代选择器 $("p span")

2、子选择器 $("parent>child")

3、兄弟选择器 $("prev+next") ：选取紧接在prev元素后的next元素，返回元素集合

4、兄弟选择器 $("prev~siblings") 选取prev元素后的所有siblings元素，返回元素集合

**三、过滤选择器**

I、基本过滤选择器，记得别忘了冒号：

1、:first 选择第一个元素返回单个元素

2、:last 返单

3、:not(selector) 不返单反集合

4、:even 偶

5、:odd 奇

6、:eq(index) 索引值等于index的元素，索引值从0开始

7、:gt(index) 索引值大于

8、:lt(index) 索引值小于

9、:focus 当前获取焦点的元素

II、内容过滤选择器

1、:contains(text) 选取含有文本内容为text的元素集合

2、:empty 选取不含子元素和文本元素的空元素集合

3、:has(selector) 选取含有选择器所匹配的元素的元素集合

示例：$("p:has(p)") 选取含有<p>元素的<p>元素（<p><p/></p>）

4、:parent 选取含有子元素或者文本的元素集合

$("p:parent") 选取含有子元素或者文本元素的<p>元素（<p><p/></p>或者<p>文本</p>）

III、可见性过滤选择器

1、:hidden 选取所有不可见的元素集合

2、：visible 选取所有可见的元素集合

IV、属性过滤选择器（返集）

1、[attribute] ：$("p[id]") 选取拥有id属性的p元素

2、[attribute=value] $("input[name=text]")

3、[attribute!=value]

4、[attribute^=value] $("input[name^=text]") 选取拥有name属性以text开始的input元素

5、[attribute$=value] 选取拥有name属性以text结束的input元素

6、[attribute~=value] $("input[class~=text]") 选取拥有class属性以空格分割的值中含有text的input元素

7、attribute1[attributeN] 合并多个属性过滤选择器

V、表单对象属性过滤选择器（返集）

1、:enabled 选取所有可用元素

2、:disabled 选取所有不可用元素

3、:checked 选取所有被选中的元素（单选框，复选框）

4、:selected 选取所有被选中的选项元素（下拉列表）

四、表单选择器（返集）

1、:text

描述：选择所有的单行文本框

示例：$(":text")选取所有的单行文本框

2、:password

描述：选择所有的密码框

3、:button

描述：选择所有的按钮

4、:checkbox

描述：选择所有的多选框

116、两数组合并，并删除第二个元素

数组合并：

<1>Array的concat()方法

<2>遍历数组并把遍历的数组元素依次添加到另外一个数组

<3>ES6语法，展开运算符 var array = [...array1,...array2]

<4>apply方法 array1.push.apply(array1,array2);

删除第二个元素（记住，索引号从0开始）

1、//array.splice(1,1,5,6)表示在删除的位置添加元素5，6

2、delete array[1] //只会删除了第二个元素，但是在其位置会显示empty，索引不会改变

117、数组相关方法的值

['a','b'].filter(x=>true) //['a','b']

['a','b'].every(x=>x==='a') //false

['a','b'].some(x=>x==='a') //true

[1,2].reduce((x,y)=>x+y) //3

[1,2].map(x=>x+1) //[2,3]

['a',...['a','b']] //["a", "a", "b"]

new Array(3).fill("a") //["a", "a", "a"]

['a','b','c'].findIndex(x=>x==='c') // 2

118、GET和POST的区别，何时用POST

1、GET：一般用于信息获取，使用URL传参，字符限制在2000以内

2、POST一般用于修改服务器资源，对所发的信息无限制

3、GET通过Request.QueryString获取变量的值，POST通过Request.Form获取

4、以下情况用POST

I 无法使用缓存文件（更新服务器上的文件或数据库）

II 向服务器发送大量数据（POST无数据量限制）

III 发送包含未知字符的用户输入时，POST更稳定可靠

119、async与await的串行与并行

async两个作用：使用await，返回一个Promise

使并行执行的异步任务串行执行

120、数组求最大值

reduce方法，最终返回的是累计计算得出的结果

//reduce:让数组中的前项和后项做某种计算，并累计最终值.

121、数组以num值大小重新排列，并返回新数组

sort()方法，还有冒泡排序，快速排序，选择排序，插入排序

122、jquery事件处理return false的作用

1、阻止默认行为

2、取消事件冒泡

3、停止回调立即返回

123、jquery获取元素不包括边框和padding的宽高

$a.width() 获取内容content的宽度

$a.height() 高

$a.innerWidth() 获取边框以内的宽度

$a.outerWidth() 获取边框以及边框以内的宽度

$a.outerWidth(true) 获取margin+border+padding+content总宽度

124、jquery中on和one函数的作用

on可以在元素上绑定一个或多个事件处理函数，使用off()方法可以删除on()绑定的事件。

参数：

events 一个或多个用空格分隔的事件类型和可选的命名空间

selector 可选，选择器字符，用于过滤选定的元素，

data 可选，作为额外数据对象传递给事件对象

fn 事件触发执行的函数

125、现有数组约定为Number，a为0，添新项值为1，a不为0，先进先出移一项

a.length === 0 ? a.push(1) : a.shift();

126、 10+“20” =？

1020 //string

127、用XML和JSON两种格式描述下面表格数据

XML 用标签包裹对象以及对象中的属性

JSON 用数组和对象包裹住属性

128、使用typeof bar ===”object” 的陷阱

null也会被视为对象，先检查是否为null

129、封装JavaScript源文件的全部内容到一个函数块的意义

创建一个私有的命名空间，防止命名污染

允许一个易于引用的别名用于全局变量，这通常可以用于jquery将$引用到jQuery命名空间。

130、不借助临时变量实现变量值交换

1、var a = 445; var b = 33;谁大谁小减不重要

b = a - b; b=412 -412

a = a - b; a=33 445

b = a + b; b=445 33

先一个值减去另一个值然后把值赋给减数，再重复执行一次减法把值赋给被减数，最后两数相加赋给减数。即先A-B赋给b，，再a-b赋给a，最后a+b赋给b

2、用ES6解构赋值交换 let [b,a] = [a,b]

131、ajax同步请求与异步请求的区别

1、同步是指：发送方发出数据等待接收方响应才能发下一个数据包

用户填写表单所有信息提交给服务器等待服务器回应是一次性的， 错误信息要重新填写

2、异步指：发送方发出数据后，不等接收方回应，接着发下个数据包的通讯方式，填写完一条发一条，不等回应接着填，填写会向服务器请求多次，提高了用户体验

3、open第三个参数默认为true，即异步请求

Xhr. open(method,url,async)。method请求方式，url请求路径，

132、setTimeout和setInterval

setTimeout只执行一次，执行完就结束了，想要再执行还需要再调用

setInterval循环执行，每达到指定时间间隔就执行定时器，最好在结束调用时清除

133、避免JS全局变量污染问题

1、定义全局变量命名空间

只创建一个全局变量并定义为当前应用容器，其他全局变量追加在该命名空间下

2、使用匿名函数，也就是自调用函数

3、使用模块化方案

requires AMD

sea.Js CMD

CommonJs

ES6模块化

134、从函数外部访问函数内部的局部变量

1、闭包，在函数内部再定义一个函数，并将该内部函数作为返回值返回，

2、将局部变量挂载在window上

135、Math对象的常用方法，以及Array，String方法和属性

**Math常用方法**

1. Math.PI 圆周率
2. Math.random();//介于 0 和 1 之间的伪随机数。
3. Math.ceil(6.6);//获取大于或等于提供数值的最小整数--向上取整
4. Math.floor(8.8);//获取小于或等于提供数值的最大整数--向下取整
5. Math.round(9.9);//四舍五入
6. Math.max(10,20,15);//取多个值中的最大值
7. Math.min(10,20,15);//取多个值中的最小值
8. Math.pow(10,2);//返回x的y次幂
9. Math.sqrt(100);//求平方根

**Array属性**

1、constructor 返回对创建此对象的数组函数的引用。

2、length 设置或返回数组中元素的数目。

3、prototyp 可以向对象添加属性和方法。

**Array方法**

1、concat()连接两个或更多的数组，并返回结果。

2、join()把数组的所有元素放入一个字符串。元素通过指定的分隔符进行分隔。

3、pop() 删除并返回数组的最后一个元素

4、push() 向数组的末尾添加一个或更多元素，并返回新的长度。

5、reverse() 颠倒数组中元素的顺序。

6、shift() 删除并返回数组的第一个元素

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，并向数组添加新元素。

10、toString() 把数组转换为字符串，并返回结果。

11、unshift() 向数组的开头添加一个或更多元素，并返回新的长度。

12、valueOf() 返回数组对象的原始值

13、some() 数组中任何一个元素是否满足指定要求就会返回true，所有元素不满足指定条件就返回false

14、every() 数组中所有元素都满足指定条件才会返回true

15、filter() 过滤数据，返回新的数据

16、map() 遍历数组，返回新的 数据

17、forEach() 遍历数组

18、findIndex() 根据条件查询下标

19、reduce() 累计运算

**String属性**

1、constructor 返回对创建此对象的数组函数的引用。

2、length 设置或返回数组中元素的数目。

3、prototyp 可以向对象添加属性和方法。

**String方法**

1、charAt()返回在指定位置的字符。

2、concat() 连接字符串。

3、indexOf() 检索字符串。

4、match() 找到一个或多个正则表达式的匹配。

5、replace() 替换与正则表达式匹配的子串。

6、slice() 提取字符串的片断，并在新的字符串中返回被提取的部分。

7、split() 把字符串分割为字符串数组

8、sub() 把字符串显示为下标。

9、sup() 把字符串显示为上标。

10、toLowerCase() 把字符串转换为小写。

11、toUpperCase() 把字符串转换为大写。

136、ES6箭头函数与ES5普通函数

箭头函数的不同点：

1. 没有this，super，arguments和new.target绑定，箭头函数这些值由外围最近一层非箭头函数决定
2. 不能通过new关键字调用，不能作构造函数
3. 没有原型
4. 不可以更改this绑定，函数内部this’值不可改，
5. 不支持arguments对象，只能通过命名参数和不定参数两种形式访问
6. 不支持重复命名的函数

137、splice代码输出结果

arr.splice(2,3,"William") //从数组第2个位置开始把后面3个元素替换成William

第一个参数是起始位置，第二个参数是改变数量，第三个参数是替换值

138、从输入一个URL地址到浏览器完成渲染的整个过程

1、浏览器地址栏输入地址并回车

2、浏览器查看当前URL是否存在缓存，并比较缓存是否过期

3、DNS解析URL对应的IP

4、根据IP建立TCP连接（三次握手），我要数据，我知道了回应，建立连接

5、发送http请求

6、服务器处理请求，浏览器接收http响应

7、浏览器解析并渲染画面

8、关闭TCP连接（四次握手）我要关闭，收到关闭，服务器关闭，客户端关闭

139、闭包应用场景

1、防抖节流

2、在js中模拟块级作用域

3、在对象中创建私有变量。a就是私有变量，在函数体内不return返回的就是私有变量

140、http3.0的改进相较2.0

http2.0特性：

1、二进制分帧传输

2、多路复用

3、头部压缩

4、服务器推送

http3建立在UDP基础上，http2建立在TCP基础上

http3特性：

1、连接迁移

2、无队头阻塞

3、自定义的阻塞控制

4、前向安全和前向纠错

141、深浅拷贝

浅拷贝：仅复制引用，彼此间的操作会影响

深拷贝：在堆中重新分配内存给内部对象，地址不同，值一样也不影响

**浅拷贝实现方案：**

1. 遍历赋值实现
2. ES6扩展运算符
3. ES6的Object.assign()方法
4. 数组方法（仅适用于类数组对象）

Array.from() 将一个类[数组](https://so.csdn.net/so/search?q=%E6%95%B0%E7%BB%84&spm=1001.2101.3001.7020)对象或者可遍历对象转换成一个真正的数组

Array.prototype.concat() 拼接

Array.prototype.slice() 截取

**深拷贝实现方案**

1. JSON.parse()和JSON.stringify() 结合转换，先把对象转JSON吗，再JSON转对象

但是缺点：

1. 对于时间类型不能深拷贝
2. 会忽略undefined
3. 会忽略Symbol
4. 无法对function进行处理，需要确认
5. 不能解决循环引用的对象
6. 递归对象中的对象实现深拷贝

进行条件判断，对数组和字符串进行不同的递归赋值