[ ].concat() 将多个数组或一个数组对象拼接为一个数组

**关于三胞胎(slice,splice,split)**

Slice： 创建一个新数组，copy原数组其中或全部部分，不会影响

原数组,参数(copy开始下标，copy截止下标)

Splice：删除数组中的某一部分，被删除的部分可以被变量接收

该方法也可进行添加操作，参数(删除开始下标，删除数量，添加对象，……)

Split：切割字符串并返回一个数组，不哆嗦

Call和apply可以修改一个方法中的this，语法分别为call(对象，参数1，参数2……无限)；apply(对象,数组) 对象是要将this修改为的对象，后面的参数或数组是要给方法中形参赋值的实参，个人认为call方法比较灵活，而apply则适合批量转移数据时使用(如真伪数组转换)

’

Visibility：hidden 隐藏能见度，只是视觉上的隐藏

GetAttribute()获取一个元素的属性的当前值

Document.documentElement.clientWidth 页面可视区宽度 height高

Oncontextmenu 鼠标右键事件

默认行为：就是浏览器自带的交互事件，比如右键可以出现选项，文本框内可以输入文本等

Event.ctrlKey是键盘上的ctrl按键，ctrl+？就必须让keyCode同时符合ctrlKey和另一个按键 同理还有shiftKey、altKey

表单标签相关操作：(后面跟函数)

Onsubmit 提交时发生 onreset重置时发生 onkeyup 输入时发生

Onblur 失去焦点时发生

ScrollTOP + even.clientY 可以完善页面下拉，div紧跟鼠标

Event.cancelBubble=ture 中断事件冒泡

Let、var、const三者的区别是在局部修改可以在局部使用，对全局没影响；可以在局部修改使用，全局也改变；不能修改

AddEventListener()为一个事件添加一个函数，并且不会覆盖掉它原来的函数(比如可以给一个按钮设置两个点击事件，直接onclick的话，只会调用最后的一条)

不加var，，， 直接x = 0，这是一种隐形全局变量，如果声明在外部他就是一个全局变量，如果声明在函数内，只在调用函数时，他就是一个全局变量，否则他就是一个局部变量

JS调用HTML data自定义属性时 要用dataset



外联javascript：<script type="text/javascript src="js位置"><script> 文件后缀名：.js

弹窗脚本：alert("弹窗内容"); 获取现在的时间 new Date()

声明变量关键字：var A a 1 $ 不能声明关键字

var 变量一号;

变量一号="变量的值";

声明赋值 var 变量一号 ="赋的值" （变量类型和内容可随时改变）

确认框：confirm

if(yes\_or\_no){} else否则{}

prompt("弹窗上显示的内容","输出框的默认值")

比较运算符：">" "<" "==" ">=" "<=" "!="或 "<>"

逻辑运算符：且"&&" 或"||" 非"!"

算数运算符：+ - \* / % 编程特别符： ++ -- += -=

字符串换算数字：字符串=字符串\*1；

三目运算符（？：）：x >0 ?a :b; x大于0吗？ 如果大于，x=a，否则x=b；

switch(变量一号){

case "某种要求":

alert("触发相应的内容");

break; (中断继续向下执行)

}

变量++ 就等于： 变量=变量+1 -同理 （如果++ --运算符在一个表达式中，那么前置的时候，先计算结果再使用，后果后置，则相反。

变量一号=变量一号+变量二号 就等于 变量一号+=变量二号

如果“+”号在字符串中间使用，会起到一个连接符的作用（如果有一边是字符串，就会起到连接符的作用）

for（循环变量的初始值；循环条件；递增值）{} 只有for循环里的值计算完毕，才会向下进行

数组：splice()插入 sort（）排序

position定位：absolute绝对定位 relative相对定位

给父级标签添加相对定位，可以改变子标签的决定点。

Document(可修改为制定的区域div).getElementById获取id

document文档 . write输出文字 getElementById获取id 获取一个输入框内的值用 .value

浏览器从上向下执行代码，获取得id会无法执行，解决方法:将script标签转移到body结束标签下方

script操作css标注形式 :

var 变量一号 = document .getElementById("id")

id.style.属性=" "; (有横杠的属性，第二个单词首字母大写，如：fontSize=" ";）

改变html属性 如img.src=" ";

如果要改变图片文件，必须给html中图片img添加id，在js中使用：

文件图片id.src=" ";

函数固定格式 function 函数名(参数 可无){} arguments不定参数

调用函数的时候直接调用函数名加括号里的东西 （注意！ 函数内声明的变量，在函数外不可以调用！）

任何操作都可以叫“” onclick="" 点击触发事件 event事件

onmousemove 鼠标移动事件 event.clientX鼠标X轴坐标

onmouseover 鼠标移动到某元素上 onmouseout 移开

锤子移动：输出鼠标所在的值，将其赋给锤子

function xx(){

var x = event.clientX ;

console.log("x坐标"+x)

chuizi.style.left = x +"px";

<audio音频文件 src="地址" id="music" ></audio>

var music = document.getElementById("music");获取音频

music.play()触发音频 audio.cloneNode().play();在js中触发

音频属性： autoplay="autoplay" 自动播放 loop="loop" 循环播放

函数内部可以运用函数外声明的函数 init 初始化

随机数Math.random(); 去掉小数：parseInt(需要去小数的值)

例：随机产生一个1-3之间的数

var a = ((parseInt(Math.random()\*1000))%3)+1;

alert(a);

例：随机生成一个0到16之间的数：

var a = Math.floor(Math.random() \* 17) + 0; Math.floor 去除小数，向下取整

间隔多少秒，执行一个内容（循环） setInterval（函数名，间隔时间'毫秒'） clearInterval 停止运行的间隔操作

间隔多少秒，执行一个内容（不循环，只执行一次） setTimeout（函数名，间隔时间'毫秒'）

需要停止循环时，要先将 循环函数声明一个函数名 例： 在外面var个 名字 名字= setinterval（xx，30）

js数组

var 变量 = new Array()； 声明一个数组 数组分类。排序 sort() 在html显示排序方法：document.write（数组.sort()）； sort()是可以传递一个函数的，该函数必须要用返回值

变量[0]中括号 = "声明的值" ; 0是第一项 length 长度 function 数组(a,b){ 返回值必须为正值，负值，和0

for循环打印数组： for（var i=0；i<arr.length;i++){ a<b 返回的是a在b的前面 负值

console.log(arr[i]); } for循环只有满足中间的条件才会向下执行，只有执行完for循环，才能向下进行 a==b 排序顺序不变 0

return a-b； （b-a排序顺序相反） a>b 那么a排在b的后面 正值

js获取html多项标签：getElementsByClassName();

在js中绑定html事件：var 变量 = document.getElementById( "id") id.onclick = 函数 没括号；

在js中访问某一行中的某一列 ： 名字.rows[行].cells[列] 也可以用（x行,y列）表示

二维数组 例：var 名字 = [[1,2],[3,4],[5,6],]; 也可以写为 =new Array（(),(),()）;

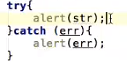
调用二位数组的第一个数组中的第一个元素 例： alert（[0],[0]）； 强调：0才是数组的第一位！！！

document.createDocumentFragment();文档碎片

pp1= pp2.slice(0); pp1等于从pp2 0开始截取的数组

x >0 ?a :b; x大于0吗？ 如果大于，x=a，否则x=b；

offsetLeft:直接在页面中获取某个单位的左间距 offsetTop：上间距 window.location.reload() 刷新本页面

检查错误，并抛出错误(例：用户输入了错误的用户名，可改变抛出方式) (图中try下面的大括号内容为错误区，catch下面为抛出错误的方式)