

目录

目录

作业讲评

学习目标

switch语句实现多分支

do...while循环

如何确定循环的次数,重点

for循环实现九九乘法表

练习：使用双层循环打印等腰三角形

练习：打印平行四边形

循环中断关键字

break

continue

isNaN()判断是否为数字

window.prompt()用户输入对话框

今日总结

今日作业

作业讲评

```
F:\GZZ20H5直播6班\Javascript基础\03-Javascript基础\4-代码\01-三五筛选.html (4-代码) - Sublime Text
文件(F) 编辑(E) 选择(S) 查找(F) 查看(V) 转到(G) 工具(T) 项目(P) 首选项(N) 帮助(H)

01-三五筛选.html
4 <meta charset="UTF-8">
5 <title>Document</title>
6 <script type="text/javascript">
7     // 输出1~100之间所有的数,3的倍数就输出三,5的倍
    数就输出五,又是3又是5的倍数就输出三五,输出三,五
    ,三五以后,不输出对应的3,5,15,30等些数字了
8
9     // 第一步:先输出1~100
10    for(var i = 1; i <= 100; i++){
11        // 第二步:使用分支结构对结果进行筛选
12        if(i%3 == 0 && i%5 == 0){ // 即是3的倍数,
            又是5的倍数
13            document.write("三五 ");
14        }else if(i%5 == 0){ // 如果是5的倍数
15            document.write("五 ");
16        }else if(i%3 == 0){ // 如果是3的倍数
17            document.write("三 ");
18        }else{ // 输出数字
19            document.write(i+ " ");
20        }
21    }
22
23    // 原因是因为多分支结构的时候,多个条件表达式,
    会从上向下依次判断,满足条件就执行,结束if结构
24 </script>
25 </head>
26 <body>
```

Document

1 2 3 4 5 三 7 8 三 五 11 三 13 14 三 五 16 17 三 19 五 三 22 23 三 五 26 三 28
29 三 五 31 32 三 34 五 三 37 38 三 五 41 三 43 44 三 五 46 47 三 49 五 三 52 53
三 五 56 三 58 59 三 五 61 62 三 64 五 三 67 68 三 五 71 三 73 74 三 五 76 77 三
79 五 三 82 83 三 五 86 三 88 89 三 五 91 92 三 94 五 三 97 98 三 五

```
02-鸡兔同笼.html
4 <meta charset="UTF-8">
5 <title>Document</title>
6 <script type="text/javascript">
7     // 鸡跟兔子一共有35只,鸡跟兔子的腿一共有94只,请问鸡兔各多少只
8     // 分析
9     // ji+tuzi = 35;
10    // ji*2+tuzi*4 = 94;
11
12    /*假设  鸡的数量  兔子的数量  判断腿总数是否满足
13    0        35-0=35      0*2+35*4 == 94?
14    1        35-1=34      1*2+34*4 == 94?
15    2        35-2=33      2*2+33*4 == 94?
16    3        35-3=32      3*2+32*4 == 94?
17    ...
18    35       35-35=0      35*2+0*4 == 94?
19
20    穷举法:就是把所有的情况都列举出来,尝试从里面找到满足条件的情况
21    */
22
23    for( var ji = 0; ji<=35 ; ji++){
24        // 通过鸡的数量求出兔子的数量
25        var tuzi = 35 - ji;
26        if( ji*2 + tuzi*4 == 94){ // 判断两个值是否相等需要用 == ,不是一个= ,一个=是赋值符号
27            document.write("鸡的数量为"+ji+",兔子的数量为"+tuzi+"<br/>");
28        }
29    }
30 </script>
31 </head>
```

学习目标

- 能够掌握switch分支语句,昨天讲的if多分支语句
- 能够掌握do...while循环语句,昨天只讲了while语句跟for语句
- 能够使用循环关键中断字break,continue
- 能够使用window.prompt让用户输入内容
- 能够使用js内置函数isNaN()判断是否为数字

switch语句实现多分支

语法:

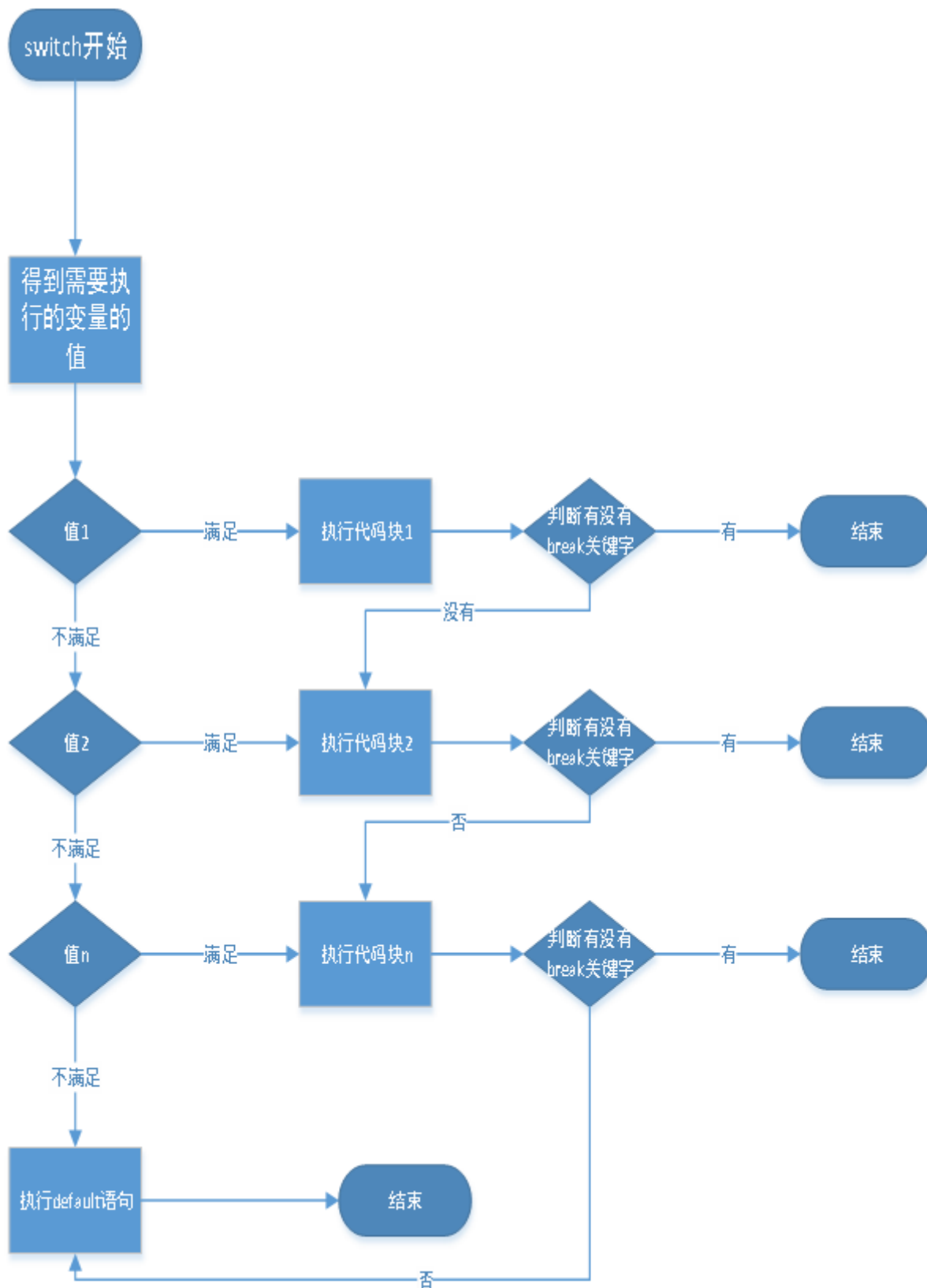
```
1  switch(变量名){
2      case 值1:
3          代码块1;
4          break;
5
6      case 值2:
7          代码块2
8          break;
9
10     case 值3:
11         代码块3;
12         break;
13
14     case 值n:
15         代码块n;
16         break;
17
18     default:
19         默认执行的代码块
20 }
```

结构说明:

1. switch语句它是拿小括号中变量的值去与每一个case后面的值进行全等比较
2. 如果比较成功(得到布尔true) 就会执行对应的代码块 还要去查找是否有break关键字
3. 如果有break关键字 那么switch语句就会结束
4. 但是如果没有break关键字 就会先执行下面的代码块 然后再判断是否有break关键字 它要找到break关键字才会结束执行。
5. 如果说变量的值与case中每一个值都不相等 那么它就会执行默认的代码块
6. break可以省略, 如果省略, 代码会继续执行下一个case

7. switch 语句在比较值时使用的是**全等操作符**, 因此不会发生类型转换(例如, 字符串'10' 不等于数值 10)

流程图:



举例:

先定义一个变量保存今天星期几,然后判断星期几应该干什么

星期一打篮球
星期二打乒乓球
星期三唱歌
星期四打桌球
星期五游泳
星期六打代码
星期天睡觉

```
// if多分支实现
var week = 10;
if(week == 1){
    document.write("星期一打篮球");
}else if(week == 2){
    document.write("星期二打乒乓球");
}else if(week == 3){
    document.write("星期三唱歌");
}else if(week == 4){
    document.write("星期四打桌球");
}else if(week == 5){
    document.write("星期五游泳");
}else if(week == 6){
    document.write("星期六打代码");
}else if(week == 7){
    document.write("星期天睡觉");
}else{
    document.write("请确保week的值是1~7");
}
```

switch语句实现

```
var week = 3;
switch(week){
    case 1:
        document.write("星期一打篮球");
        break;

    case 2:
        document.write("星期二打乒乓球");
        break;

    case 3:
        document.write("星期三唱歌");
        break;

    case 4:
        document.write("星期四打桌球");
        break;

    case 5:
        document.write("星期五游泳");
        break;

    case 6:
        document.write("星期六打代码");
        break;

    case 7:
        document.write("星期天睡觉");
        break;

    default:
        document.write("请确保week的值是1~7");
}
```

关于if多分支语句与switch之间的区别

- switch语句它在功能上和if语句多分支很相似
- 如果是**范围**的判断推荐使用**if语句多分支**
- 如果是**固定值**之间的比较 推荐使用**switch**语句
- 因为**switch能够实现的功能** 我们其实完全可以使用if语句多分支来实现!

do...while循环

do..while循环和while循环非常像，二者经常可以相互替代，但是do..while的特点是**不管条件成不成立，都会执行一次。**

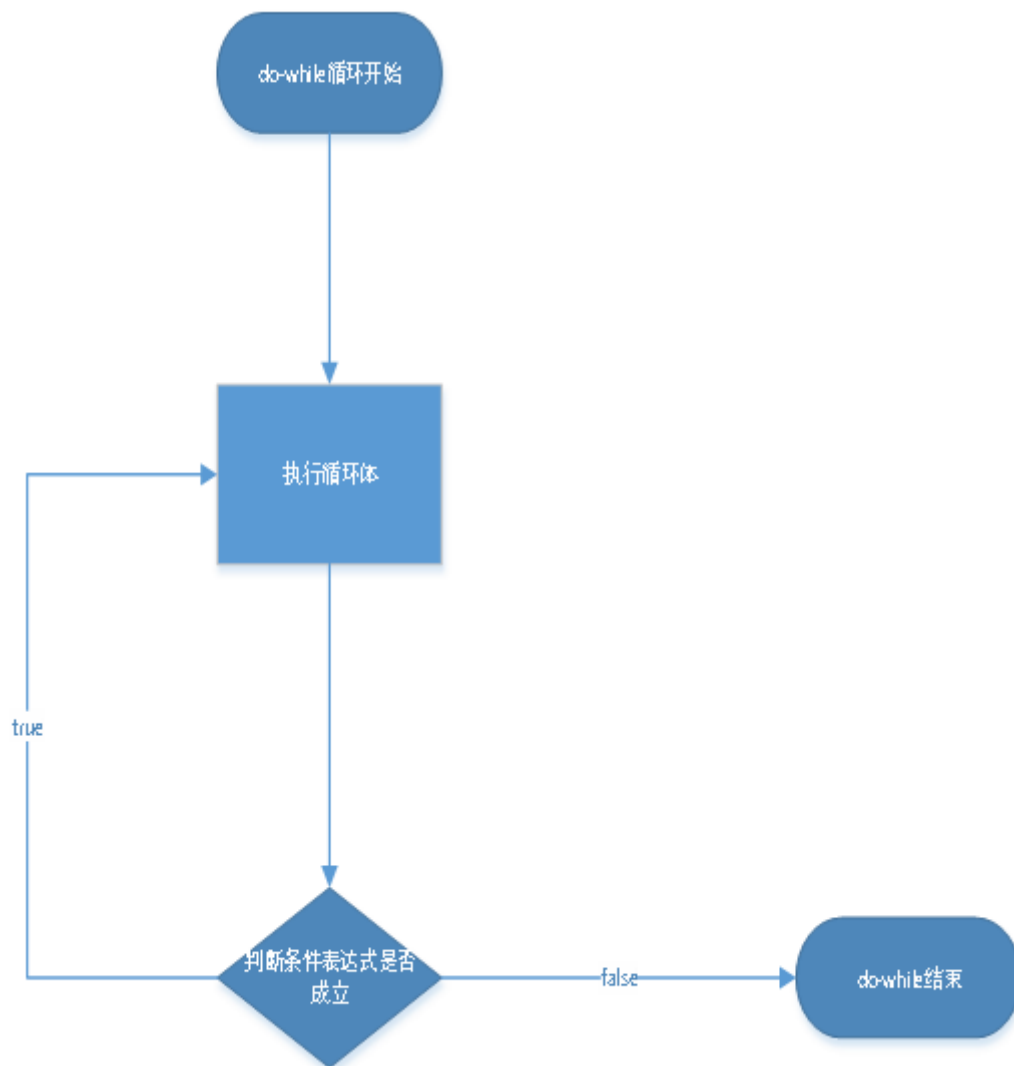
语法:

```
1  初始化变量;  
2  do{  
3    循环体;  
4    变量更新;  
5  }while(条件表达式);
```

结构说明:

- 先执行一次循环体，然后再来判断条件表达式是否成立。如果条件表达式成立就继续执行循环体，如果不成立就会结束do while循环。
- **不管条件表达式是否成立，do...while循环都会执行一次循环体。**

流程图:



举例:使用do-while循环,输出1~100之间所有的数,包括1和100

// do...while的语法如下

```
/*初始化变量;  
do{  
    循环体;  
    变量更新;  
}while(循环条件表达式)*/
```

```
var i = 1;  
do{  
    document.write(i+" ");  
    i++;  
}while(i <= 100 )
```

小结:while与do...while之间的区别

- while循环是**先判断条件表达式再来执行循环体**
- do...while**先执行一次循环体 再来判断条件表达式是否成立 不管条件表达式成不成立 它先会执行一次循环体**

```
// while需要先判断循环条件是否成立,如果不成立,则不会执行循环体  
/*var i = 10;  
while( i<=5 ){  
    document.write( i );  
    i++;  
}*/
```

```
// do...while先执行一次循环体 再来判断条件表达式是否成立 do...  
while不管条件表达式成不成立 它先会执行一次循环体  
var i = 10;  
do{  
    document.write( i );  
    i++;  
}while( i<=5 )
```

如何确定循环的次数,重点

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <title>Document</title>
6   <script type="text/javascript">
7     // 我们在写循环的时候,需要确定循环的次数
8
9     // i += 4; 等同于 i = i+4
10    /*for(var i = 3;i < 10; i += 4 ){
11      document.write("hello<br/>");
12    }*/
13
14    // 如何确定的呢?
15    // 第一步:根据初始化变量得到开始循环的初始值
16    // 第二步:看循环条件表达式,到什么时候结束
17    // 第三步:看变量更新,看看变量是如何变化的
18    // i = 3; i = 7 i < 10
19    // 最后只有i=3和i=7能够进入循环体
20
21
22    for(var i = 5;i > 1 ; i--){
23      document.write(i+"<br/>");
24    }
25
26    // i = 5 4 3 2 i > 1
27  </script>
28 </head>
29 <body>
30
31 </body>
32 </html>
```

for循环实现九九乘法表

- 打印9行一列的表格

```
// 因为我们已经学完循环,循环就是解决一些重复的事情
var str = '<table border="1" width="600">';

// 使用for循环,循环9次,输出相同的内容
for(var i = 1;i <= 9;i++){
    // 什么东西需要重复做,什么东西就是循环体
    str += '<tr>';
    str += '<td>&nbsp;</td>';
    str += '</tr>';
}
str += '</table>';

document.write( str );
```

- 打印一行9列的表格

```
var str = '<table border="1" width="600">';
str += '<tr>';
for(var i=1;i<=9;i++){
    str += '<td>&nbsp;</td>';
}
str += '</tr>';
str += '</table>';
document.write( str );
```

- 打印9行9列的表格

```
// 打印9行9列的表格
var str = '<table border="1" width="600">';

// 这个时候,我们又发现了一样的代码,一样的代码又有规律,
// 我们就可以考虑使用循环完成
for(var j=1;j<=9;j++){
    str += '<tr>';
    for(var i=1;i<=9;i++){
        str += '<td>&nbsp;</td>';
    }
    str += '</tr>';
}

str += '</table>';
document.write( str );
```

- 打印对应行有对应列的9行表格,比如第一行,有1列; 第二行,有2列; 第三行,有3列

```
// 打印对应行有对应列的9行表格,比如第一行,有1列; 第二行,有2列; 第三行,有3列...

// 以下代码中有两个变量,i跟就j, i是控制行数还是列数 j是控制行数还是列数
// j行,i列
var str = '<table border="1" width="600">';

for(var j=1;j<=9;j++){// j是控制行数,外层循环

    str += '<tr>';

    for(var i=1; i<=j ; i++){// i是控制列数,内层循环
        str += '<td>&nbsp;</td>';
    }

    str += '</tr>';

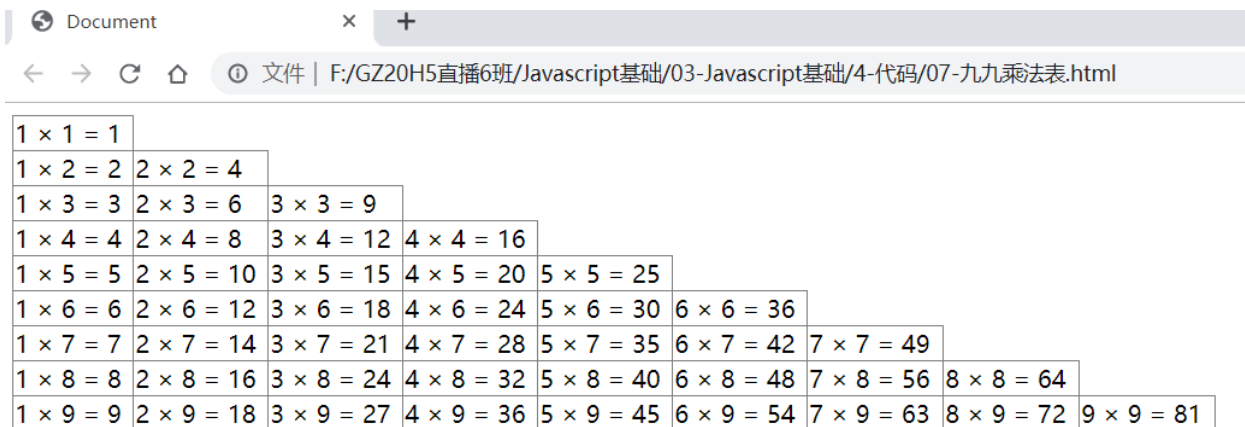
}
str += '</table>';

document.write( str );
```


- 打印九九乘法表

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <title>Document</title>
6   <script type="text/javascript">
7     // 输出九九乘法表所用表格
8     // 第一行一个单元格
9     // 第二行两个单元格
10    // 第三行三个单元格
11    // .....
12    // 第九行九个单元格
13
14    // rules="all"合并边框线
15    var str = '<table border="1" width="800" rules="all">';
16    for(var j=1;j<=9;j++){// j控制行数
17      str += '<tr>';
18      for(var i=1;i<=j;i++){// i控制列数
19        // 小技巧:先写固定,再写会变化的,先死后活
20
21        // ×是HTML中字符实体,代表乘号
22        str += '<td>'+i+' × '+j+' = '+i*j+'</td>';
23
24      }
25      str += '</tr>';
26    }
27    str += '</table>';
```

```
28
29 document.write( str );
30 </script>
31 </head>
32 <body>
33
34 </body>
35 </html>
```



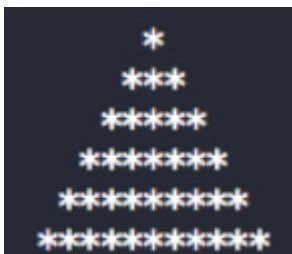
小结:

什么时候使用for循环什么时候使用while循环 不是绝对的

已知循环次数的时候多用for循环

未知循环次数的时候一般用while循环

练习：使用双层循环打印等腰三角形



- 1 分析：第一行，1个星星，5个空格
- 2 第二行，3个星星，4个空格
- 3 第三行，5个星星，3个空格
- 4 第四行，7个星星，2个空格

```
5 第五行, 9个星星, 1个空格
6 第六行, 11个星星, 0个空格
7
8 先打印星星, m = 2, i不变, n = 1, 公式= 2 * i + 1
9
10 后打印空格, m = -1, i不变, n = 5, 公式= -1 * i + 5
```

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4 <meta charset="UTF-8">
5 <title>Document</title>
6 <style>
7 /*设置网页中的字体为仿宋*/
8 body{
9 font-family: "仿宋";
10 font-size:50px;
11 }
12 </style>
13 <script type="text/javascript">
14 /*等腰三角形星星塔规律
15
16 行数 星星个数
17 1 1
18 2 3
19 3 5
20 4 7
21 5 9
22 6 11
23 ... ...
24 n 2*n-1*/
25 document.write("<h2>第一种居中方法</h2>");
26 document.write("<div style='text-align:center;border:5px solid red;width:280px;'>");
27 // 先输出星星
28 for(var i = 1; i <= 6; i++){// 外层循环控制行数
29
30 for(var j = 1;j<= 2*i-1 ; j++){// 里层循环控制星星个数
31 document.write("*");
```

```
32  }
33
34  document.write("<br/>");
35  }
36  document.write("</div>");
37
38
39
40  document.write("<h2>第二种居中方法</h2>");
41  for(var i = 1; i <= 6; i++){// 外层循环控制行数
42
43    // 输出空格
44    for(var k = i; k<= 5; k++ ){
45      document.write(" ");
46    }
47
48    // 输出星星
49    for(var j = 1;j<= 2*i-1 ; j++){// 里层循环控制星星个数
50      document.write("*");
51    }
52
53    document.write("<br/>");
54  }
55  </script>
56  </head>
57  <body>
58
59  </body>
60  </html>
```


第一种居中方法

```
  *
 ***
*****
*****
*****
*****
```

第二种居中方法

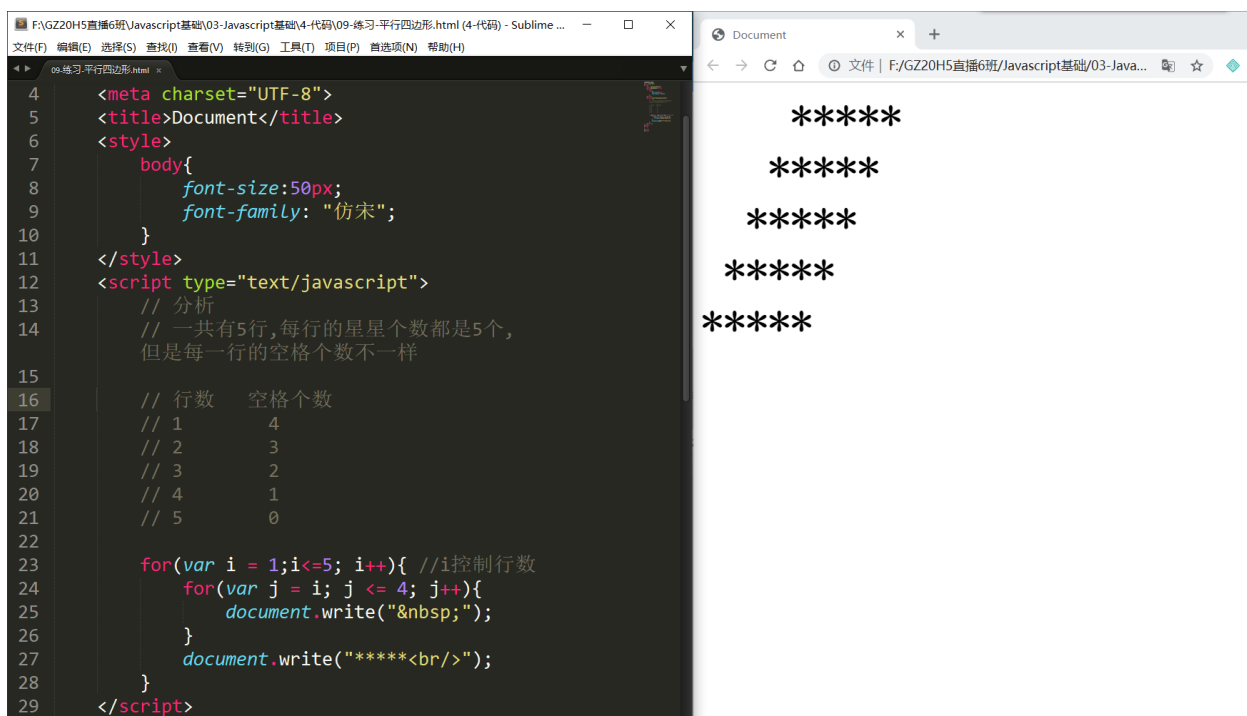
```
  *
 ***
*****
*****
*****
*****
```

练习：打印平行四边形

```
  ★★★★★
★★★★★
★★★★★
★★★★★
★★★★★
```

- 1 第一行：4个空格
- 2 第二行：3个空格
- 3 ...
- 4 第五行：0个空格

5 推公式: $m * i + n = -1 * i + 4$



The screenshot shows a code editor on the left and a web browser on the right. The code editor displays an HTML document with a JavaScript script. The script uses nested loops to generate a parallelogram of asterisks. The browser window shows the rendered output, which is a parallelogram of asterisks.

```
4 <meta charset="UTF-8">
5 <title>Document</title>
6 <style>
7   body{
8     font-size:50px;
9     font-family: "仿宋";
10  }
11 </style>
12 <script type="text/javascript">
13   // 分析
14   // 一共有5行,每行的星星个数都是5个,
15   // 但是每一行的空格个数不一样
16   // 行数  空格个数
17   // 1      4
18   // 2      3
19   // 3      2
20   // 4      1
21   // 5      0
22
23   for(var i = 1; i <= 5; i++){ //i控制行数
24     for(var j = i; j <= 4; j++){
25       document.write("&nbsp;");
26     }
27     document.write("*****<br/>");
28   }
29 </script>
```


循环中断关键字

典型的, 当条件表达式不成立了, 那么循环体就会终止执行。也可以称之为循环它**寿终正寝**了。

循环提前终止:本来条件表达式是成立了, 循环体还可以继续的往下执行, 但是我们可以使用一些关键字让其提前终止。

break和continue 循环终止的关键字 **都需要配合 if语句来实现** 主要的作用是为了提升循环的效率!

break

break打断的意思, 当在循环体中遇到了break关键字以后, 立即跳出整个循环, 即循环结束, 开始执行循环后面的内容

举例: 小明吃5个包子, 吃到第3个发现里面有半个虫子, 其余的不吃了

```

12
13 // 举例：小明吃5个包子，吃到第3个发现里面有半个
14 // 虫子，其余的不吃了
15 for(var i = 1; i <= 5 ; i++){
16     document.write("小明吃第"+i+"个包子<br/>");
17
18     if(i == 3){
19         document.write("OMG,有半个虫子~<br/>");
20         // 直接结束整个循环，
21         // 后续循环体就不用执行了
22         break;
23     }
24 }

```

小明吃第1个包子
 小明吃第2个包子
 小明吃第3个包子
 OMG,有半个虫子~

continue

continue继续的意思，立即跳出当前循环，继续下一次循环

举例：小明吃5个包子，第3个有虫子，就扔掉第3个，继续吃第4个第5个包子

```

23
24 // 举例： 小明吃5个包子，第3个有虫子，就扔掉第3
25 // 个，继续吃第4个第5个包子
26 for(var i = 1; i <= 5 ; i++){
27
28     if(i == 3){
29         document.write("小明吃第3个包子，
30         // 发现有虫子，扔掉<br/>");
31         continue;
32         // 结束本次循环，continue后面的代码，
33         // 就不执行了，进入下一次循环
34     }
35     document.write("小明吃第"+i+"个包子<br/>");
36 }
37 </script>
38 </head>
39 <body>
40 </body>
41 </html>

```

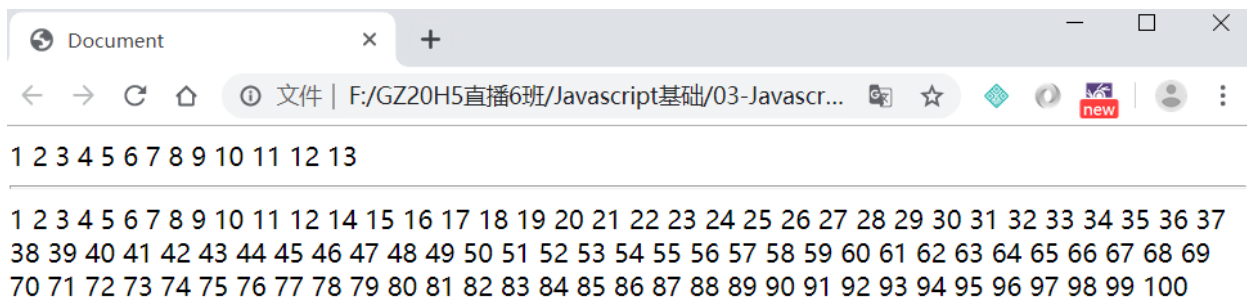
Document

小明吃第1个包子
 小明吃第2个包子
 小明吃第3个包子,发现有虫子,扔掉
 小明吃第4个包子
 小明吃第5个包子

练习：

- 输出1~100之间的所有的数，但要求碰到13这个数的时候就停止输出13之后的数字,13正常输出
- 输出1~100之间的所有的数，但要求碰到13这个数的时候,13不输出,13之后的数字继续输出

```
11-练习-循环中断关键字.html
4  <meta charset="UTF-8">
5  <title>Document</title>
6  <script type="text/javascript">
7      // 输出1~100之间的所有的数，但要求碰到13这个数的时候就停止输出13之后的数字,13正常
      输出
8      for(var i = 1;i <= 100; i++){
9          document.write(i+" ");
10         if(i == 13){
11             break;
12         }
13     }
14
15     document.write("<hr/>");
16
17     //
    输出1~100之间的所有的数，但要求碰到13这个数的时候,13不输出,13之后的数字继续输出
18     for(var i = 1;i <= 100; i++){
19
20         if(i == 13){
21             continue;
22         }
23
24         document.write(i+" ");
25     }
26
27
28 </script>
```



isNaN()判断是否为数字

isNaN => is not a number

isNaN(变量名) 如果是NaN就返回true 如果不是的就是false

作用： 先将一个变量的数据类型自动的转换为Number 如果是NaN就会得到true 如果是一个数字的话就是得到false

举例：

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <title>Document</title>
```

```
6 <script type="text/javascript">
7 // isNaN => is not a number 是一个非数字吗
8 // isNaN(变量名或者值) 判断变量或者值是否为数字
9 // 如果不是数字,返回true
10 // 如果是数字,返回false
11 // isNaN判断规则: 先将一个变量的数据类型自动的转换为Number 如果是NaN就会得到
true 如果是一个数字的话就是得到false
12
13 // 小结:只要是在字符串中含有非数字的都会转换为NaN;
14
15
16 // 变量原来的值 转成Number的值
17 var v1 = 123; // 数字
18 var v2 = "123"; // 数字
19 var v3 = "abc123"; // NaN
20 var v4 = "123abc"; // NaN
21 var v5 = true; // 1
22 var v6 = false; // 0
23 var v7 = undefined; // NaN
24 var v8 = null; // 0
25 var v9 = ""; // 0
26 var v10 = " "; // 0
27 console.log( isNaN(v1) );// false
28 console.log( isNaN(v2) );// false
29 console.log( isNaN(v3) );// true
30 console.log( isNaN(v4) );// true
31 console.log( isNaN(v5) );// false
32 console.log( isNaN(v6) );// false
33 console.log( isNaN(v7) );// true
34 console.log( isNaN(v8) );// false
35 console.log( isNaN(v9) );// false
36 console.log( isNaN(v10) );// false
37 console.log(" ");
38
39 console.log( Number(v7) );// NaN
40 console.log( Number(v8) );// 0
41 </script>
42 </head>
43 <body>
44
45 </body>
```

window.prompt()用户输入对话框

这个方法是属性window对象的方法 所以window可以省略

这个方法是用来向浏览器中弹出一个用户输入对话框

[HTML DOM Window 对象参考手册](#)

定义和用法

`prompt()` 方法用于显示可提示用户进行输入的对话框。

语法

```
prompt(text,defaultText)
```

参数	描述
text	可选。要在对话框中显示的纯文本（而不是 HTML 格式的文本）。
defaultText	可选。默认的输入文本。

说明

如果用户单击提示框的取消按钮，则返回 `null`。如果用户单击确认按钮，则返回输入字段当前显示的文本。

在用户点击确定按钮或取消按钮把对话框关闭之前，它将阻止用户对浏览器的所有输入。在调用 `prompt()` 时，将暂停对 JavaScript 代码的执行，在用户作出响应之前，不会执行下一条语句。

第一个参数:text表示提示信息

第二个参数:defaultText 表示输入框的中默认文本 默认值

但是注意:这两个参数都可以省略不写!

这个方法有两个按钮 :确定按钮、取消按钮

当用户点击确定按钮时会得到一个String类型的数据

当用户点击取消按钮会得到一个关键字 `null`

举例:

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <title>Document</title>
```

```

6  <script type="text/javascript">
7  // window.prompt()用户输入对话框,因为是window对象下的方法,所以可以省略windo
w. 直接写成prompt()
8
9  // window.prompt(提示信息,输入框中默认文本);
10
11
12  // window.prompt这个方法有两个按钮 :确定按钮、取消按钮
13  // 当用户点击确定按钮时会得到一个"String"类型的数据
14  // 当用户点击取消按钮会得到一个关键字 空类型null
15
16  // 我们可以使用一个变量保存用户输入的值
17  var str = window.prompt("请输入您的银行卡密码",123456);
18  // document.write( str );// null
19  // document.write("<br/>");
20  // document.write( typeof(str) );// object
21
22  if( str == null ){// 如果点击了取消
23  document.write("有内鬼,停止交易");
24  }else{// 如果输入的内容,点击确定
25  document.write( "你刚输入的密码" + str );
26  document.write("<hr/>");
27  document.write( typeof(str) );
28  }
29
30  </script>
31  </head>
32  <body>
33
34  </body>
35  </html>

```

今日总结

xmind没有特别说明,就是要做

有特别说明的时候,才可以不用做

今日作业

今天开始,咱们的作业的参考答案不当天发了,我们第二天发,也是为了让同学们不要依赖性,免得很多同学直接复制老师答案或者对着老师的答案抄

大家晚上不懂的多问,每天都看到助教小姐姐在群里喊,大家有不懂,快点问我鸭~