

1. 循环

在js中，主要有三种类型的循环语句：

- for 循环
- while循环
- do...while循环

2. for循环

在程序中，一组被重复执行的语句被称为循环体，能否继续重复执行，取决于循环的终止条件，由循环体及循环的终止条件组成的语句，称之为循环语句

2.1. 语法结构

for循环主要用于把某些代码循环执行若干次，通常跟计数有关，其语法结构是：

```
for(初始化变量;条件表达式;操作表达式){  
    //循环体  
}  
// 初始化变量就是用var声明的一个普通变量 通常用于作为计数器使用  
// 条件表达式就是用来决定每一次循环是否继续执行，就是终止的条件  
// 操作表达式就是每次循环最后执行的代码，经常用于我们计数器变量进行更新（递增或者递减）
```

执行过程：

- 1、首先执行里面的计数器变量 `var i=1`，但是这句话在for循环里面只执行一次
- 2、去`i<=100`来判断是否满足条件，如果满足条件，就去执行循环体 不满足条件就退出循环
- 3、最后去执行`i++` `i++`是单独写的代码递增，第一轮结束
- 4、接着去执行`i<=100` 如果满足条件，就去执行循环体，不满足条件就退出循环 第二轮

断点调试：

断点调试是指自己在程序的某一个行设置一个断点，调试时，程序运行到这一行就会停下，然后你可以一步一步往下调试，调试过程中可以看各个变量当前的值，出错的话，调试到出错的代码行即显示错误，停下。

断点调试可以帮助我们观察程序的运行过程

浏览器按f12 -->source-->找到需要调试的文件-->在程序的某一行设置断点

watch：监视，通过watch可以监视变量的值的变化，非常的常用

f11:程序单步执行，让程序一行一行的执行，这个时候，观察watch中变量的变化

代码调试的能力非常重要，只有学会了代码调试，才能学会自己解决bug的能力。初学者不要觉得代码调试麻烦就不去调试

2.2. for 循环重复某些相同的操作

for循环因为有了计数器的存在，我们还可以重复的执行某些操作，比如做一些算数运算

3. 双重for循环

3.1. 双重for循环概述

很多情况下，单层for循环并不能满足我们的需求，比如我们需要打印一个5行5列的图形、打印一个倒直角三角形等，此时我们就可以通过循环嵌套来实现

循环嵌套是指在一个循环语句中再定义一个循环语句的语法结构，例如在for循环语句中，可以再嵌套一个for循环，这样的for循环语句我们称之为双重for循环

语法结构：

```
for (外层的初始化变量;外层的条件表达式;外层的操作表达式){  
    for (里层的初始化变量;里层的条件表达式;里层的操作表达式){  
        //执行语句;  
    }  
}
```

- 我们可以把里层的循环看做外层循环的语句
- 外层循环循环一次，里面的循环执行全部

4. while 循环

while语句可以在条件表达式为真的前提下，循环执行指定的一段代码，知道表达式不为真时结束循环

while语句的语法结构如下：

```
while:(条件表达式){  
    //循环体代码  
}
```

执行思路：

- 先执行条件表达式，如果结果为TRUE，则执行循环体代码，如果为FALSE，则退出循环，执行后面的代码
- 执行循环体
- 循环体代码执行完毕后，程序会继续判断执行条件表达式，如条件仍为TRUE，则会继续执行循环体，直到循环条件为FALSE时，整个循环过程才会结束

5. do while 循环

do while 语句是while语句的一个变体，该循环会先执行一次代码块，然后对条件表达式进行判断，如果条件为真，就会重复执行循环体，否则退出循环

语法结构：

```
do {  
    //循环体；  
}while (条件表达式)
```

执行思路：

跟while不同的地方在于：

- do while 是先执行一次循环体再判断条件，如果条件表达式结果为真，则继续执行循环体，否则退出循环
- do while 循环体至少执行一次

6. continue break

6.1. continue关键字

continue关键字用于立即跳出本次循环，继续下一次循环（本次循环中continue之后的代码就会少执行一次）

例如：吃5个包子，第三个有虫子，就扔掉第三个，继续吃第4个第5个包子

代码实现：

```
for (var i=1;i<=5;i++){  
    if (i==3){  
        continue;//只要遇见continue就退出本次循环，直接跳到i++  
    }  
    console.log('我正在吃第'+i+'个包子');  
}
```

6.2. break 关键字

break用于立即跳出整个循环（循环结束）

例如：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

代码实现如下：

```
for (var i=1;i<=5;i++){  
    if (i==3){  
        break;  
    }  
    console.log('我正在吃第'+i+'个包子');  
}
```