**学习目标**

1、掌握while语句的语法结构

2、掌握while语句的应用逻辑

3、掌握do-while语句的语法及应用

4、掌握for与while的区别

发散：

**for、**while、do-while这几种循环语句，它一定有四步：

1、初始值

2、结束条件

3、执行代码

4、初始值向结束条件靠拢，否则就是死循环

**while语句**

**格式：**

初始值

while(条件){

需要执行的代码；

步长 ;

}

说明：while就是当什么什么的时候，即**当括号里的条件成立的时候**，去执行大括号中的脚本，它是先判断条件，再执行

如：

var i=1; // 1、初始值

while(i<=100){ //2、结束条件

document.write(i+'<br />'); //3、执行语句

i++; // 4、向结束条件靠拢

}

说明：一定要在执行语句中，改变i的值，否则就是一个死循环。

拿它同for来比较。同for功能一模一样，只是语法不太一样。

**do-while 语句**

**语法：**

初始值;

do{

需要执行的代码；

增值

}while(条件);

说明：它是先执行，再判断条件。这种语法的循环至少要被执行一次。

如：1到10之间的偶数

var j = 1; // 1、初始值

do{

//3、执行语句

if(j%2===0){

document.write(j+'<br />');

}

j++; // 4、向结束条件靠拢

}while(j<=10); //2、结束条件

如：至少执行一次：

var j = 18;// 1、初始值

do{

//3、执行语句

if(j%2===0){

document.write(j+'<br />');

}

j++; // 4、向结束条件靠拢

}while(j<=10); //2、结束条件

例：1+2+3+4+…+100

while方法：

var num = 0;

var i=1;

while(i<=100){

num += i;

i++;

}

console.log(num);

do-while方法：不建议用do-while方法

var num = 0;

var i=1;

do{

num += i;

i++;

}while(i<=100);

console.log(num);

for方法

var num = 0;

for(var i=1;i<=100;i++){

num += i;

}

console.log(num);

总结：什么时候用while、do-while、for

for：适合已知循环次数的循环体

while：适合未知循环次数的循环体