# day05

复习

while(循环条件){ 循环体 };

do{ 循环体 }while(循环条件);

for(初始值;循环条件;增量){ 循环体 };

循环嵌套

break

continue

var i=0,sum=0;

do{

i++;

if(i%2===0) continue;

if(i%5===0) break;

sum+=i;

}while(i<10);

console.log(sum);

## 1.函数

parseInt()/parseFloat()/alert()/prompt()/isNaN...

函数分为系统函数和自定义函数

function:功能体，函数(方法)，可以接受若干个数据，返回处理的结果。用于封装反复执行的代码——饺子机。

(1)创建普通函数

function 函数名称(){

函数体——封装的代码

}

调用：

函数名称();

练习：创建函数，封装计算1-100之间所有整数的和并打印出现；调用3次。

练习：创建函数，封装两个数字相加的和；调用多次

(2)创建带有参数的函数

function 函数名称(参数列表){//用于接收传递的数据

函数体

}

调用：

函数名称(参数列表);//实际要传递的数据

参数：创建函数时的参数称为形参，调用时的参数称为实参，

同时实参会赋值给形参，多个参数之间用逗号隔开。参数可以有0个或者多个，如果形参未被赋值，值为undefined。

练习：创建函数，计算1~任意数字之间所有整数的和，调用多

次。

练习：创建函数，计算任意两个年份之间的闰年个数并打印，

调用多次。

(3)创建带有返回值的函数

function 函数名称(参数列表){

函数体

return 值;//返回值，就是函数调用后的结果

}

调用:

函数名称(参数列表);

注意事项：如果函数中没有return或者return后没有加任何的值，返回undefined。return 后的代码不会再执行。

练习：创建函数，传递两个参数，返回任意两个数字的最大值

练习：创建函数，传递订单的状态码这个参数，返回对应的中文状态。

1-等待付款 2-等待发货 3-运输中 4-已签收 5-已取消 其他-无法追踪

练习：创建函数，返回任意三个数字的最大值 3 4 5

练习：创建函数，传递一个年份，要求返回是否为闰年，是->true,否则返回false

练习：创建函数，传递一个年份，如果是闰年返回366，不是闰年返回365

练习：计算1-任意数字之间所有整数阶乘的和

## 2.变量的作用域

(1)作用域

变量或者函数的可访问范围，分为两种

函数作用域：在函数中使用var声明的变量，只能在函数内部访问

全局作用域：在全局使用var声明的变量，可以再任意合法位置访问

注意：在函数内不适用var声明的变量，是全局变量，可以在函数外访问到。

(2)变量声明提升

JS程序在执行前，会将使用var声明的变量提升到所在作用域的最前边，但赋值还是在原来的位置。

## 3.函数作用域

函数的可访问范围，分为全局作用域和函数作用域;在全局作用域下创建的函数可以在任何作用域下调用，函数作用域下创建的函数，只能在这个函数内部调用。

函数提升

和变量提升一样，JS在执行前，会将使用function关键字创建的

函数提升到所在作用域的最前边。在调用的位置执行函数体中的代码。

## 4.递归

在函数内部调用自身，本身是一个死循环。

递归的正确用法：要有结束的条件，结合着return。

练习：使用递归计算1-任意数字之间所有整数的和。

斐波那契数列

1 1 2 3 5 8 13 21 34 55...

从第3个数开始，每个数的值为前两项相加的和

课后任务：

(1)复习

(2)练习：使用递归计算斐波那契数列的第n项

使用循环计算斐波那契数列的第n项

12/30/50/100