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;i的变化){ 循环体 }

break 和 continue

|  |
| --- |
| var i=0;  var sum=0;  do{  i++;  if(i%2==0){  continue;  }  if(i%5==0){  break;  }  sum+=i;  }while(i<10) |

循环嵌套

学习一门编程语言的基本步骤

(1)了解背景知识：历史、现状、特点、应用场景。

(2)搭建开发环境，编写hello world

(3)变量和常量

(4)数据类型

(5)运算符

(6)逻辑结构

(7)通用小程序

(8)函数和对象

(9)第三方库、框架

(10)实用的项目

学习目标

函数相关(重点)

变量的作用域(重点)

函数的作用域(重点)

递归调用(掌握)

1.函数

parseInt()/parseFloat()/typeof()...

分为系统函数和自定义函数

自定义函数

function：功能体，函数，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——要封装的反复执行的代码  } |

调用

函数名称();

练习：使用函数封装10+20的结果，并打印出来，调用3次。

练习：使用函数封装计算1~100所有数字的和，打印结果，调用3次。

(2)创建带有参数的函数

|  |
| --- |
| function 函数名称(参数列表){ //形参->形式上的参数  函数体  } |

调用

函数名称(参数列表); //实参->实际的参数

**参数列表**：可以是0个或者是多个数据，之间用逗号隔开；创建函数时的参数称为形参，调用函数时的参数成为实参，调用的时候，实参的值会赋给形参。

形参本质上就是一个声明了的变量，但未赋值。

练习：创建函数getSum，传递1个参数，计算1~任意数字之间的和。

练习：创建函数getRun，传递2个参数，打印任意两个年份之间的所有闰年个数。

(3)带有返回值的函数

|  |
| --- |
| function 函数名称(参数列表){  函数体  return 返回值;  } |

调用

函数名称(参数列表)

return表示函数执行后，所返回的结果。

注意：

①如果没有return或者return后没有返回值，结果都是undefined。

②return可以返回任意类型的数据

③return后的所有代码都不会被执行

练习：创建函数add，传递3个参数，返回3个数字的和。

练习：创建函数getMax，传递2个参数，返回最大值。

练习：创建函数getMax，传递3个参数，返回最大值。

练习：创建函数isRun，传递1个参数(年份)，是闰年返回true，不是闰年返回false。

练习：创建函数getStatus，传递1个参数(订单状态码)，返回对应的中文；

1-待付款 2-待发货 3-运输中 4-已签收 5-已取消 其它状态码-无法查询

**对比return、break和continue**

return 返回结果，用于终止函数的执行，常用于函数中。

break 用于结束循环，结束switch语句；

continue 跳过当前循环，继续下一次循环。

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：

**全局作用域**：在全局作用域下声明的变量可以在任意位置访问到。

**函数(局部)作用域**：函数作用域下声明的变量只能在函数内部访问。

注意：在函数内部使用var关键字声明的变量是局部变量，不适用var关键字声明的变量是全局变量。

(2)变量声明提升

JS程序执行前，使用var关键字声明的变量会提升到**所在作用域**的最前边；但赋值还是在原来的位置。

|  |
| --- |
| console.log(a);  var a=1; //var a;会提升到前边; 而 a=1还是在当前位置; |

3.函数的作用域

函数和变量类似，也分为全局作用域和函数(局部)作用域

全局作用域：在全局作用域下创建的函数可以在任意位置调用

函数(局部)作用域：在函数(局部)作用域下创建的函数只能在函数内部调用

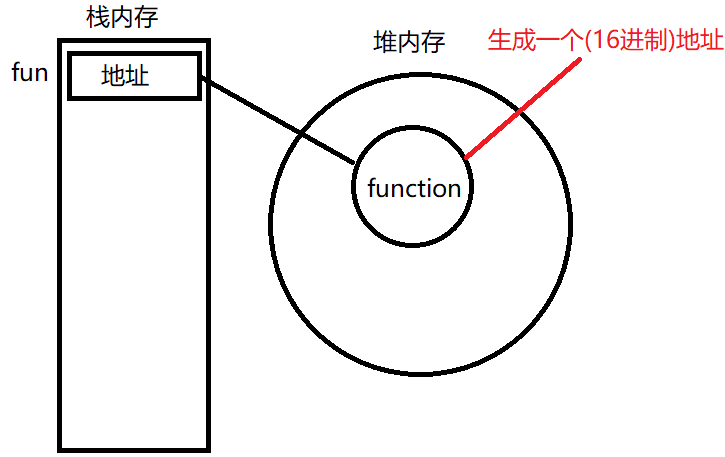
**函数声明提升**

和变量一样，JS在程序执行前，把使用function声明的函数提升到最前边。

|  |
| --- |
| function fun(){  return 1;  } |

函数调用 fun() 获取函数的返回值(return后的值)

函数名称 fun 保存的是函数在堆内存中的地址



4.递归的调用

在函数的内部调用自身

递归要有跳出条件，需要结合return使用。

斐波那契数列

1 1 2 3 5 8 13 21 34 55 89 144

课后任务：

(1)复习今天内容，整理思维导图

(2)练习：

创建函数，传递1个参数，使用递归计算斐波那契数列的第n项是多少

(3)预习JS的对象