1.递归

递归属于CPU密集型的算法，对CPU要求比较高

JS属于单线程，只使用一个CPU内核，JS不适合递归中深层次嵌套

课后拓展：创建函数，传递任意一个数字，使用递归计算1~该数字之间所有整数的和

提示：倒着运算n~1

http://www.codece.com/archives/102

2.匿名函数

没有函数名称的函数

function () { }

(1)创建函数

|  |
| --- |
| //函数声明  function fn(){  } |
| //函数表达式  var fun=function(){  } |

函数表达式创建的函数通过变量创建，只存在变量的声明提升，不存在函数提升，必须先创建再调用

对比**函数名称**和**函数名称()**

函数名称() 调用函数，执行函数体中的代码，得到函数的返回结果。

函数名称 本质是一个变量，保存了一个函数。

(2)匿名函数自调用

|  |
| --- |
| (function(){  函数体 //创建的作用域，防止污染全局  })(); |

(3)回调函数

将函数以实参的形式(**函数名称或者匿名函数**)传递叫做回调函数，通过参数来调用

|  |
| --- |
| function dong(){ }  function ran(madai){  madai(); //调用传递的回调函数  }  ran( dong );  ran( function(){ } ) |

3.系统函数

Number()/parseInt()/parseFloat()

isNaN() 检测一个值转为数值后是否为NaN，会自动调用Number()

是NaN - true 不是NaN-false

isFinite() 检测一个值是否为有限值，只有Infinity是无限值

是- true 不是-false

0做除数返回Infinity 2/0

eval() 执行字符串中的表达式

4.对象

对象属于引用类型数据

对象是一组属性和方法的集合

一部手机是对象：属性有颜色、尺寸、品牌.. 方法有打电话、玩游戏、看视频....

然哥是对象：属性有颜色、身高、体重、年龄、性别... 方法走路、扛麻袋、摊煎饼、养兔子..

万物皆对象

(1)JS中的对象

自定义对象，用户创建的对象

宿主对象，根据不同的执行环境划分

内置对象，JS提供的对象

(2)自定义对象

对象字面量创建

内置构造函数创建

自定义构造函数创建

(3)使用对象字面量创建对象

使用大括号{ }创建空对象，属性由属性名和属性值组成，之间用冒号隔开；多组属性之间用逗号隔开；属性名的引号可加可不加，如果含有特殊字符必须加引号。

(4)访问属性

对象.属性名

对象['属性名']

如果属性名中含有特殊字符，只能使用第二种方式

如果属性不存在返回undefined

练习：创建一本图书对象，包含有图书的编号，标题，价格，作者；打印出图书的标题，对价格重新赋值，添加图书出版社属性；最后打印该对象

(5)使用内置构造函数

new Object()

创建一个空对象，需要单独添加每一个属性

练习：使用内置构造函数创建一个汽车对象，包含有汽车的品牌，型号，颜色，长度，宽度

(6)遍历属性

使用循环依次访问每一个属性

|  |
| --- |
| for(var k in 对象){  k 代表对象中的每一个属性名  对象[k] 代表每一个属性名对应的属性值  } |

练习：使用对象字面量创建班级对象，包含有班级编号，所在城市，人数，班长；遍历对象中的每一个属性

课后任务

(1)复习今天内容，整理思维导图

(2)练习

创建成绩对象，包含有若干个成绩，遍历对象得到每一个成绩，最后计算出总成绩已经平均成绩。

(3)预习JS第7天

http://www.codece.com/archives/98