正课:

1. String:

删除 切割

2. RegExp:

3. Math:

4. Date:

1. String:

替换:

衍生: 删除: 其实就是将关键词替换为""

切割: 将字符串按切割符，切割为多段子字符串

1. 简单切割: 切割符是固定的

var substrs=str.split("切割符");

返回值: 切割后的多段子字符串组成的数组

强调: 切割后的结果中，不包含切割符

固定套路: 打散字符串为字符数组:

var chars=str.split("")

2. 高级切割: 切割符不固定

var substrs=str.split(/正则/)

2. RegExp

什么是: 保存一条正则表达式，并提供用正则表达式执行验证和查找的API 的对象

何时: 只要用正则执行验证和高级查找，都要先创建正则表达式对象

如何:

创建: 2种:

1. // : var reg=/正则/ig;

何时: 只要正则表达式是固定不变的

字符冲突: //之中如果再出现/，应改为\/

2. new: var reg=new RegExp("正则","ig");

何时: 只要正则表达式需要用程序动态生成

字符冲突:""之中，如果包含\d,\w,\s，都要换成\\d,\\w,\\s

比如: /\d{6}/ "\\d{6}"

API:

1. 验证:

var bool=reg.test(str)

检查str是否满足reg的规则要求

坑: 正则默认只要能找到符合条件的子字符串，就算验证通过

解决: 今后，凡是验证，必须同时前加^后加$

2. 高级查找:

即查找每个关键词的内容，又查找每个关键词的位置

如何:

var arr=reg.exec(str)

在str中找下一个符合reg规则的关键词

返回值: 每次只返回一个关键词及其位置

arr:[

0: 关键词的内容,

index: 关键词的位置

]

如果找不到，返回null

3. Math:

什么是: 保存数学计算的常量，并提供数学计算的API

何时: 只要执行数学计算时

如何:

创建: 不能用new创建，就可直接使用的对象

所有API都用Math.直接调用

API:

1. 取整: 3种:

1. 上取整: 只要超过，就取下一个整数

Math.ceil(num)

2. 下取整: 舍弃小数部分

Math.floor(num)

parseInt(str) 可去掉str数字后的非数字字符

3. 四舍五入取整: 够5进一，不够舍弃

Math.round(num) 缺: 只能取整

优: 返回number类型

vs n.toFixed(2) 优: 可按任意小数位数四舍五入

缺: 返回字符串，不能直接做加法

自定义四舍五入的函数:

2. 乘方和开平方

乘方:Math.pow(底数,幂)

开平方:Math.sqrt(num)

3. 最大值和最小值:

Math.max(值1,值2,...) 获得多个值中的最大值

Math.min(值1,值2,...) 获得多个值中的最小值

缺点: 不支持数组类型参数

解决: Math.max/min(...arr)

数组前的...是特殊运算符: 可将数组打散为单个元素，再传给函数

4. 随机数:

Math.random() 0~1之间的一个随机小数

在任意min~max之间取一个随机整数:

parseInt(Math.random()\*(max-min+1)+min)