* 构造函数的了解。
* 构造函数：其实就是一个普通函数，对构造函数使用new运算符调用，就能生成实例对象。
* 构造函数的基本特点：首字母大写 new运算符创建。
* 数组的概念及定义

一组一般情况下相同类型的数据。

除了 Object 类型之外，Array 类型是ECMAScript 最常用的类型。而且 ECMAScript 中的 Array 类型和其他语言中的数组有着很大的区别。 虽然数组都是有序排列， 但 ECMAScript 中的数组每个元素可以保存任何类型。ECMAScript 中数组的大小也是可以调整的。

* 数组的创建方式
* 第一种是 new 运算符（构造函数）;

注意：给构造函数传递一个值也可以创建数组。如果传递的是数字，则会按照该数自创建包含 给定项数的数组；而如果传递的是其他类型的参数，则会创建包含那个值的只有一项的数组。

* + 第二种是字面量;

在计算机科学中，字面量是用于表达源代码中一个固定值的表示法。

字面量表示如何表达这个值，一般除去表达式，给变量赋值时，等号右边都可以认为是字面量。

字面量分为字符串字面量、数组字面量和对象字面量等。

* 数组的操作
  + 使用索引下标来读取数组的值--下标是固定的，从0开始
  + length:静态赋值/动态赋值（动态的，无需指定大小）
  + 数组的length属性 ，代表长度

数组的 length 属性不只是只读的。通过设置这个属性，可以从数组的末尾移除项或向数 组中添加新项。

* + 遍历访问数组 普通for循环和 for-in循环（结合数组或对象）

约定：所有的点操作可以改写成中括号的形式： obj.name <===> obj[name]

obj.name 用点操作符，后面只能跟字符串。

obj[name] 此时的name可以是变量也可以是字符串，如果字符串加引号。

* 检测数组：Array.isArray() 静态方法

Array.isArray()：确定括号中的某个值到底是不是数组

* 数组的方法(重点)
* 栈方法：栈是一种 LIFO（Last-In-First-Out，后进先出）的数据结构，也就是最新添加的项最早被移除。

push() 可以接收任意数量的参数，把它们逐个添加到数组末尾，并返回修改后数组的长度。

pop() 从数组末尾移除最后一项，减少数组的length值，然后返回移除的项。

* + 队列方法：队列数据结构的访问规则是 FIFO （First-In-First-Out，先进先出）。

unshift()方法能够在数组前端添加任意个项并返回新数组的长度。

shift() 方法能够移除数组中的第一个项并返回该项，同时将数组长度减 1。

* 重排序方法

reverse() 方法会反转数组项的顺序。

sort() 方法按字母表升序排列数组项。

注意：如果按照其他方式而非字母表顺序进行数组排列，必须给sort方法传递一个比较函 数做参数。

* + 操作方法

concat() 方法可以基于当前数组中的所有项创建一个新数组。具体来说，这个方法会先创 建当前数组一个副本，然后将接收到的参数添加到这个副本的末尾，最后返回新构建的数 组。在没有给 concat() 方法传递参数的情况下，它只是复制当前数组并返回副本。如果传 递给concat()方法的是一或多个数组，则该方法会将这些数组中的每一项都添加到结果数 组中。如果传递的值不是数组，这些值就会被简单地添加到结果数组的末尾。

slice() 方法它能够基于当前数组中的一或多个项创建一个新数组。可以接受一或两个参 数，即要返回项的起始和结束位置，不包括结束位置。

splice() 方法，对数组进行删除、插入、替换，是最强大的数组方法。

删除：删除任意数量的项，只需指定2个参数，要删除的第一项的位置和要删除的项数。

插入：可以向指定位置插入任意数量的项，只需提供 3 个参数：起始位置、0（要删除的 项数）和要插入的项。如果要插入多个项，可以再传入第四、第五，以至任意多个 项。

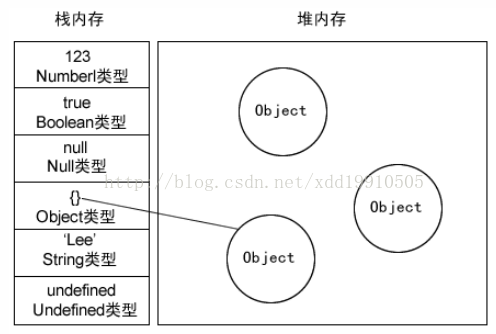
替换：可以向指定位置插入任意数量的项，且同时删除任意数量的项，只需指定 3 个参数： 起始位置、要删除的项数和要插入的任意数量的项。插入的项数不必与删除的项数相 等。

join()方法将数组中的所有元素都转化为字符串并连接在一起，返回最后生成的字符串。参 数为数组项之间的连接符号。

* 整数数组/字符串数组/对象数组/二维数组
* 应用

1. 定义一个含有30个整型元素的数组，按顺序分别赋予从2开始的偶数；然后按顺序每五个数求出一个平均值，放在另一个数组中并输出。试编程。
2. 随机点名程序。

* 数组的排序（冒泡、选择）
* 冒泡排序的原理：冒泡排序(Bubble Sort)是一种简单的排序算法，它重复地遍历要排序的数列，每次比较相邻的两个元素，如果他们的顺序不满足排序条件就把他们交换过来。遍历数列的工作是重复地进行直到不再需要交换，也就是说该数列已经排序完成。这个算法的名字由来是因为越小(或越大)的元素会经由交换慢慢“浮”到数列的顶端。
  + 选择排序的原理：选择排序(Selection Sort)是一种简单直观的排序算法。它的工作原理是每一次从待排序的元素中选出最小(或最大)的一个元素，存放在该序列的起始位置，这样一次遍历，只需一次交换，便可将最值放置到合适位置;重复前述操作直到全部待排序的数据元素排完。。
* JS堆和栈的了解
  + 栈：自动分配内存空间，系统自动释放
  + 堆：动态分配的内存，大小不定，也不会自动释放。



* + 基本类型和引用类型

基本类型：存放在栈内存中的简单数据段，数据大小确定。

引用类型（object）：存放在堆内存中的对象，实际保存的是一个指针（地址），这个指针指向的是另一个位置。当访问引用类型的值时，先从栈中获取该对象的地址指针，然后从堆内存中取得对应所需的数据。

* 值传递和址传递
* 对象的本质和意义

对象是javascript的数据类型。对象是一种复合值，它将很多值聚合在一起，可通过名字访 问这些值。对象也看做是属性的无序集合，每个属性都是一个名/值对。

* JSON表示对象的方法
* 应用

1. 随机生成一个五位以内的数，然后输出该数共有多少位，每位分别是什么。
2. 数组的冒泡排序（封装函数）
3. 数组的选择排序（封装函数）
4. 编写函数map(arr) 把数组中的每一位数字都增加30%
5. 编写函数has(arr , 60) 判断数组中是否存在60这个元素，返回布尔类型--indexOf

* 综合应用

1. 编写函数norepeat(arr) 将数组的重复元素去掉，并返回新的数组。
2. 有一个从小到大排好序的数组。现输入一个数，要求按原来的规律将它插入数组中。
3. 使用字面量形式创建一个对象，该对象存储一个学生的信息，该对象包含学号、身份证、年龄、性别、所学专业等属性信息，同时该对象包含一个自我介绍的方法，用来输出该对象的所有信息.