数据类型 转换规则：

基本数据类型：String, Number, Boolean, Undefined, Null.

引用数据类型：Array, Object, Function.

JS是一门弱类型语言，不同类型的变量可以直接相加，大多数情况下都会自动转换，但是有时候需要手动强制转换。typeof() 返回变量的数据类型。 console.log(typeof(arr[i])).

1.非数值类型转换成数值类型：

Number(), parseInt(), parseFloat()

Number(mix):

mix的值如果可以转换成数字转换成数字 不能转换的转换成0或者NaN。

Boolean的true和false转换成1和0。Number转换成Number，Undefined转换成NaN，Null转换成0，String数字部分转换成数字（010会转成10，1.1会转换成1.1，十六进制会转换成对应的十进制），非数字部分转换成NaN。String是空的转换成0。

parseInt(), parseFloat()专门把字符串转换成数字。

parseInt(string):

将string转换成数值，不遵循四舍五入。

var num = ["123" , "124.4" , "234asd" , "asf456"]

123、124、234、NaN。

parseInt(string, radix):

以radix为基底，将string转换成十进制的整数。radix的取值为2-32。

2.非字符串类型转换成字符串：

toString(), string()

String(mix):

该函数可以将任意类型的mix的value转换成字符串。

toString()有两种用法：

（1）demo.toString(): 将demo转换成字符串类型，demo不能为null和undefined。

（2）demo.toString(radix): 将十进制的数demo转换成目标进制数。

123.0.toString(8) 将123转换成八进制的字符串。----不能写成123.toString(8)，浏览器解析会解析成小数。

3.将值转换成boolean类型：

（1）原始类型值的转换方法：

以下六个值的转换结果为false，其他为true：

undefined, null, -0, +0, NaN, ''(空字符串）

（2）对象的转换规则：

所有对象的布尔值都是true，甚至连false的布尔对象也是true: Boolean(New Boolean(false)).

隐式的数据类型转换：

Pass

JS正则表达式：

var patt = /frank/i

通常用于两个字符串方法：search(), replace().

（1）search()返回子字符串的起始位置。

var str = " hello Frank!";

var n = str.search(/Frank/i) // return 6

（2）replace(a, b)搜索字符串并将a转换为b。

var str = document.getElementById("demo").innerHTML;

var txt = str.replace(/microsoft/i, Frank);

document.getElementById("demo").innerHTML = txt;

i 执行对大小写不敏感的匹配；g 执行全局匹配；m 执行多行匹配；[]用于查找某个范围之间的任何字符（[abc][0-9])；\d 查找数字；\s 查找空格； n+ 匹配至少包含一个n；n\* 匹配包含0或多个n；n? 匹配包含0或1个n。

a.test(b) 如果a在b里，返回trun，反之false。

pass

JavaScript变量提升：

JS中，函数及变量的生命都将被提升至函数的最顶部。

只有声明的变量会提升，初始化的不会。 var n = 5;

JavaScript作用域：

（1）变量在函数内声明，变量为局部作用域。

局部变量，只能在函数内部访问。

因为局部变量只作用于函数内，所以不同的函数可以用相同的变量名。

（2）变量在函数外定义，即为全局变量。

全局变量有全局作用域：网页中所有函数和脚本均可使用。

var carname = "Volvo";

function myFunction(){

//此处可调用carname;

}

如果变量在函数内声明，但没有使用var关键字，该变量为全局变量。

//此处可调用carname;

function myFunction(){

carname = "Volvo";

}

JS变量生命周期在他声明时候初始化，局部变量在函数执行完毕后销毁，全局变量在页面关闭后销毁。

函数参数只在函数内起作用，是局部变量。

在HTML中，全局变量是window对象，所有数据变量都属于window对象。