**js书写方式**

**1.内部js：**

在任意位置（除了title标签内）书写

通常在body结束之前书写 先加载html和css

<script type ="text/javascript">
alert(1);
</script>

**2.外部js：**

  1.创建独立的 后缀名为.js的文件​

2.在html中引入js文件，任意位置

<script type= "text/javascript" src = ""></script>

​注：绝不能再外部引入样式中再写js语句

**3.内联：**

在标签的开始位置写js语句，一般需要触发方式

**注释**

  //单行注释

  /\*多行注释\*/

**js的三种输出方法**

1.alert(1) 弹窗

2.console.log() 控制台打印

3.document.write() 在html页面中打印

**变量：**

**js中一个可以改变值的容器**

var a = 10；

var是声明变量的关键字，不能缺省。

一个=号是把右边的值赋给左边。

**定义变量的规则：**

1.可以有字母、数字、下划线\_、美元符号$

2.不能以数字开头

3.不能以关键字和保留字作为变量名称

4.驼峰命名法

boxImgBtn小驼峰 用于声明变量、函数

BoxImgBtn大驼峰 用于类、构造函数的命名

**Number 数值型** 3 333 3.14 NaN

使用typeof检查NaN的数据类型会返回 Number

js中可以表示的数字的最大值

Number.MAX\_VALUE:1.7976e+308

Number.MIN\_VALUE:5-e324

超过最大值和最小值的会表示为Infinity和-Infinity

**String 字符串型** “你好”

**Boolean 布尔值** true或false 布尔值只有两个

**Undefined 未定义** 值只有一个就是undefined 声明一个变量但是不给变量赋值时 值就是undefined

**Null 空值** null这个值专门用来表示一个为空的对象

（symbol） es6新增的一个类型

**Object 对象类型**

**Array 数组类型**

**Function 函数类型**

使用typeof检查null的数据类型会返回object

**undefined与null的区别?**

\* undefined代表没有赋值

\* null代表赋值了, 只是值为null

**什么时候给变量赋值为null呢?**

\* var a = null //a将指向一个对象, 但对象此时还没有确定

\* a = null //让a指向的对象成为垃圾对象 被垃圾回收器回收

**六个数据类型：**Number、String、Boolean、Undefined、Null、Object

**五个基础数据类型：**Number、String、Boolean、Undefined、Null

**两种特殊数据类型：**Null、Undefined

**两种引用类型：**Array、Function

**查看数据类型：typeof** 返回值是一个字符串

typeof不能区分object和null、 object和Array

：number、string、boolean、undefined、function、object

**js语法中的细节**

每句话后面的分号可要可不要

变量可一次定义多个

弱引用类型 变量的类型划分不严格 可以存储不同的数据类型

**关键字和保留字**

关键字：已经赋予了特定的操作和含义

保留字：预备，未来可能会有特殊含义

不能作为变量名称



**数据类型转换**

**隐式类型转换**

var a = 1;

var b = "3";

进行 - \* / % 会进行隐式类型转换，将string类型转变成number类型

**显式类型转换**

Number（） 可以转换纯数字的字符串 如果不是纯数字的话 则返回NAN （Not a Number）非常严格

parseInt（）将数字开头的字符串类型转为整数的number类型

parseFloat（）将数字开头的字符串类型转为有小数的number类型 （前提条件是本来就有小数）

toFixed（保留几位小数） 将数字类型转换成字符串类型

eg. var num = 5.56789;
var n=num.toFixed(2);

var num = 177.234

console.log(num3.toFixed()) // 输出：177

console.log(num3.toFixed(2)) // 输出：177.23

console.log(num3.toFixed(6)) // 输出：177.234000

String()

toString(） 可以将数字指定为其他进制 返回转换后的进制 一开始默认是十进制 a.toString(2) 将a转换为二进制

NaN （Not a Number） 这不是一个数字

1.在进行非数字类型运算的时候就会出现

2.显式类型转换中，非纯数字开头或者非数字开头的也会出现

判断是否是数字 isNaN（）

如果参数值为 NaN 或字符串、对象、undefined等非数字值则返回 true, 否则返回 false。

**将其他的数据类型转换成字符串**

**方式1：调用被转换数据类型的.toString（）方法 不传参数 传参会转换为其他进制**

不会影响原变量，会将转换后的结果返回

eg. var b=a.toString();

注：null和undefined这两个值没有toString（）方法 会报错；

**方式2：调用String（）函数 并将被转换的数据作为参数传递给函数**

返回值为转换后的数据

使用String（）函数进行强制类型转换时，

对于Number和Boolean实际上就是调用.toString（）方法

但是对于null和undefined，就不会调用.toString（）方法

它会将null和undefined直接转换为"null"和"undefined"

**将其他的数据类型转换成数字**

**方式1：调用Number（）函数**

-字符串-->数字

1.如果是纯数字的字符串，则直接将其转换为数字

2.如果是字符串中有非数字的内容，则转换为NaN

3.如果字符串是一个空格或者一连串空格，则转换为0

-字符串-->数字

1.true转换为1

2.false转换为0

-null-->数字

转换为0

-undefined-->数字

转换为NaN

**方式2：专门用来对付字符串**

parseInt（）可以将一个字符串中的一个有效整数部分取出来然后转换为Number

parseFloat（）可以将一个字符串中的一个有效小数部分取出来然后转换为Number

如果不是数字开头的字符串则转换为NaN

如果对非String的数据类型使用parseInt（）和parseFloat（）

它会将其先转换为String再转换为Number

eg. var bool=true;

parseInt(bool)返回NaN;

**将其他的数据类型转换成布尔值**

**方式1：使用Boolean（）函数将a转换为布尔值**

var a = 123；a= Boolean（a）；

-数字-->布尔值

-除了0和NaN会被转换为false，其余都会转换为true

-字符串-->布尔值

-除了空串被转换为false，其余都会转换为true

-null和undefined-->布尔值

-都会被转换为false

-object-->布尔值

-对象都会被转换为true

**方式2：（隐式类型转换）**

-为任意的数据类型做两次非运算，即可将其转换为布尔值

-例子:
var a = "hello";

a = ! !a; // true