一个函数最多可以有255个参数

内置函数

|  |  |
| --- | --- |
| 函数 | 说明 |
| eval() | 求字符串中表达式的值 |
| isFinite() | 判断一个数值是否为无穷大 |
| isNaN() | 判断一个数值是否NaN |
| parseInt() | 将字符型转换为整型 |
| parseFloat() | 将字符型转换为浮点型 |
| encodeURI() | 将字符串转换为有效的URL |
| encodeURIComponent() | 将字符串转换为有效的URL组件 |
| decodeURI() | 对encodeURL()编码的文本进行解码 |
| decodeURIComponent() | 对encodeURIComponent()编码的文本进行解码 |

常用内置函数的详细介绍

1. parseInt()函数

该函数主要将首位为数字的字符串转换成数字，如果字符串不是以数字开头，那么将返回NaN(即无意义的数字)。

1. parseFloat()函数

该函数主要将首位为数字的字符串转换成浮点型数字，如果字符串不是以数字开头，那么将返回NaN。

1. isNaN()函数

isNaN( Num ) Num:需要验证的数字

该函数主要用来检验某个值是否为NaN。

1. isFinite()函数

该函数主要用来检验某个表达式是否为无穷大。

1. encodeURI()函数

该函数主要用于返回一个URI字符串编码后的结果

1. decodeURI()函数

该函数主要用于将已编码为URI的字符串解码成最初的字符串并返回。

此函数可以将使用encodeURI()转码的网络资源地址转换为字符串并返回，也就是说decodeURI()函数是encodeURI()函数的逆向操作。

说明：URI与URL都可以表示网络资源地址，URI比URL的表示范围更加广泛，通常情况下，两者等同的。encodeURI()函数只对字符串中有意义的字符进行转义。

var num1="123abc";

var num2="abc123";

document.write("123abc转换结果为："+parseInt(num1)+"<br>");

document.write("abc123转换结果为："+parseInt(num2)+"<br>");

var num3="123.456abc";

var num4="abc123.456";

document.write("123.456abc转换结果为："+parseFloat(num3)+"<br>");

document.write("abc123.456转换结果为："+parseFloat(num4)+"<br>");

document.write("123.456abc转换结果是否为NaN："+isNaN(parseFloat(num3))+"<br>");

document.write("abc123转换结果是否为NaN："+isNaN(parseInt(num2))+"<br>");

document.write("1除以0的结果是否为无穷大："+isFinite(1/0)+"<br>");

document.write("转换为网络资源地址为："+encodeURI("http://127.0.0.1/save.html?name=测试")+"<br>");

document.write("转换为网络资源地址的字符串为："+decodeURI(encodeURI("http://127.0.0.1/save.html?name=测试"))+"<br>");

定义函数的方法：构造函数Function()和函数直接量

区别：

构造函数Function()允许在运行时动态创建和编译javascript代码，而函数直接量却是程序结构的一个静态部分，与函数语句相似。

每次调用构造函数Function()都会解析函数体，而且创建一个新的函数对象。如果对构造函数的调用出现在一个循环中，或者出现在一个经常被调用的函数中，这种方法的效率将非常低。而函数直接量不论出现在循环体还是嵌套函数中，既不会在每次调用时都被重新编译，也不会在每次遇到时都创建一个新的函数对象。

Function()创建的函数使用的不是静态作用域，相反地，该函数总是被当做顶级函数来编译。