# 前言

MaxScript足够强大，相比C#或.Net，MaxScript更加适合做3dsMax的二次开发，随着3dsMax版本的更新，各块功能的接口也一直有更新支持，近几年都不会被其他的语言超越。作为TA要掌握MaxScript是非常必要的，同时我们也需要进阶到C++来做一些更底层的能为美术解决问题的功能。

MaxScript主要就是用来解决重复劳动，把各个不在一个界面的常用功能集合在一起，把繁琐的操作通过一个按钮或者快捷键实现，积累一些Functions。

遇到问题多多与人交流，有问题就直接问，不用思前想后那么多，人们都会喜欢去探讨有价值的问题。

## 常见的MAXScript交流网址

[http://forums.cgsociety.org](http://forums.cgsociety.org/)

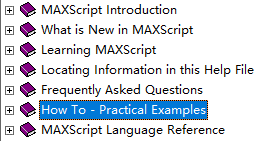
<http://forums.cgsociety.org/forumdisplay.php?f=98>

<http://area.autodesk.com/>

<http://www.cgplusplus.com/bbs/>

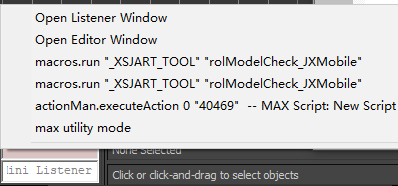
## 学会侦听和查找文档

通过官方提供的MAXScript帮助文档，借助MAXScript Listener来快速入门。



# 界面 User Interface

## MAXScript Listener脚本侦听



### 打开方式

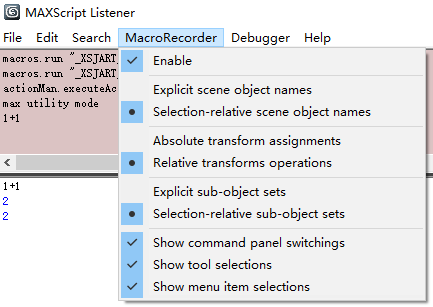
左下角-右键-Open Listener Window

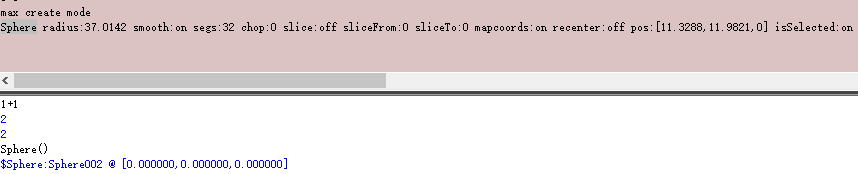
快捷键F11

工具面板上的MAXScript按钮

### 侦听设置

打开需要侦听的功能：大部分命令可以显示在红色区域





白色区域为调试命令和代码的窗口。

快捷清除窗口内容：Ctrl+D；写在代码里的命令：clearListener()。

### 报错提示



红色文字的报错信息。

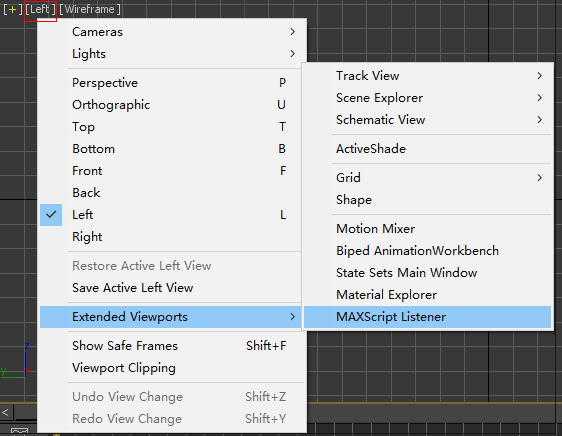
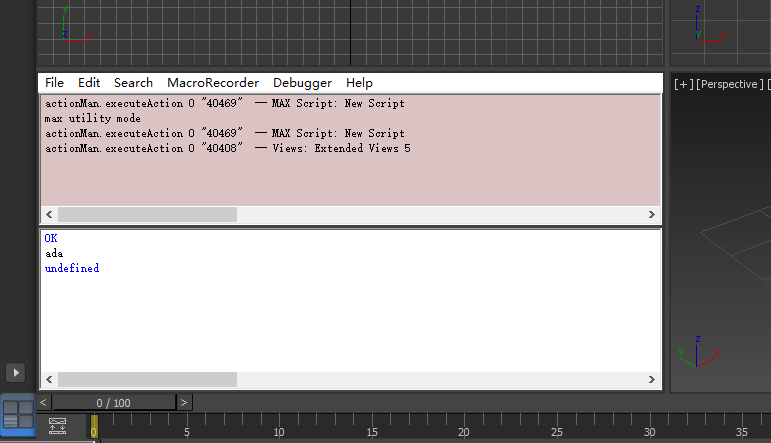
**undefined**



未定义的变量，系统出错时可能会返回一个undefined。

**Tip**

在Max视窗中可以扩展侦听窗口，但是不会保存设置。打开新的Max文件后，需要重新设置。

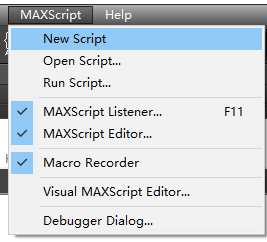
 

## MAXScript Editor windows脚本编辑窗口

### 打开方式

工具面板上的MAXScript按钮

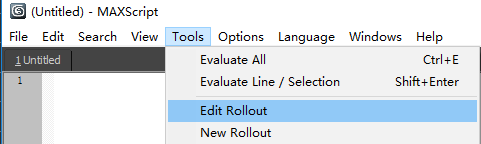
菜单上栏上的MaxScript选项

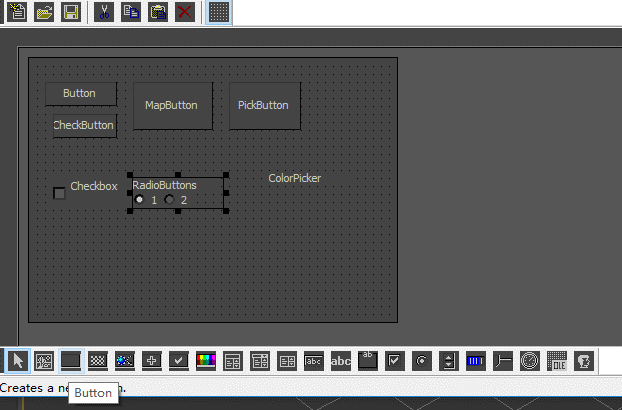


### 可视编辑界面

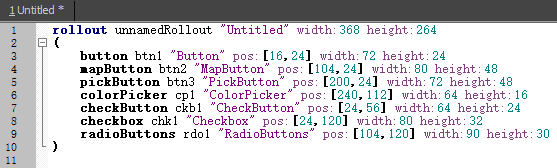
运行脚本通过Ctrl+E执行，或者Tools菜单上的Ecaluate All/Selection进行调试。

通过Tools菜单上的Edit Rollout窗口来制作工具界面。





### Rollout生成的代码

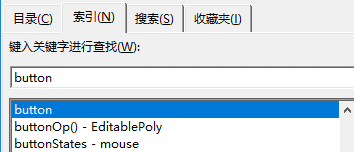


### 文本对齐的快捷键

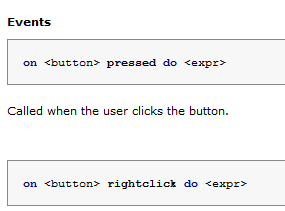
TAB

SHIFT+TAB

### 通过帮助文档的索引，搜索控件的使用



控件的使用可以查询文档，控件的事件**Events**用于执行代码操作。



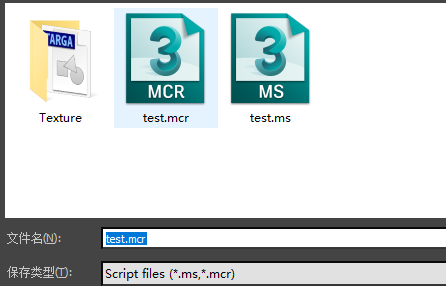


## 3、MAXScript脚本文件

### **运行脚本**：Ctrl+E

### **保存脚本**：Ctrl+S

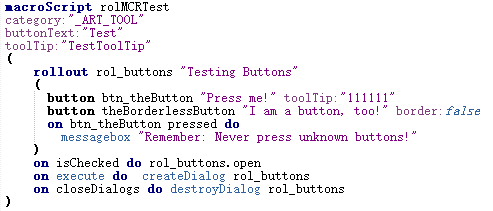
### 脚本格式：ms、mcr



### Ms脚本文件

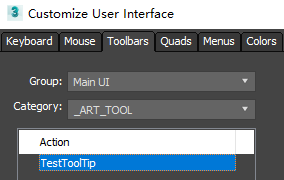
普通的运行脚本，通过拖动到max执行使用。

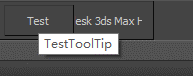
### Mcr脚本文件



Mcr脚本拖动到Max中不会直接执行，需要配置快捷键或工具栏按钮等方式使用。



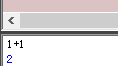




### Mse加密文件

加密：encryptScript "../test.ms"，会在脚本文件的目录下生成一个同名的.mse加密文件。

## 4、Debugging

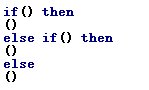


在脚本侦听窗口中以“行”为单位执行指令

在脚本编辑窗口中新建.ms文件进行调试

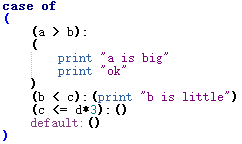
# 语法 Conditionals and Loops

## 1、if <expr> then <expr> [else <expr>]





## 2、case [<expr>] of (<cases>)



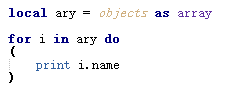
## 3、do <expr> while <expr>

**while <expr> do <expr>**

**exit**

## 4、for <var\_name> (in | =) <sequence> (do |collect) <expr>

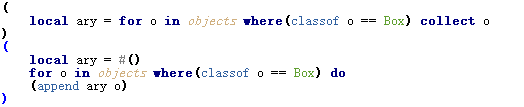
**exit [with <expr>]**



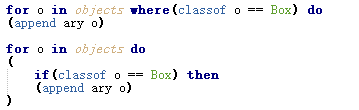
以下两种写法功能相同：



以下两种写法功能相同：



以下两种写法功能相同：



## 5、try <protected\_expr> catch <on\_err\_expr>

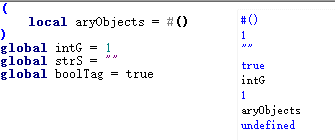
## 6、a and b or c and d and not e

# 三、变量与函数 Functions

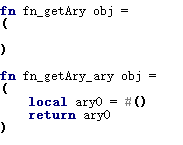
局部变量 **local** 全局变量**global**

局部函数 **fn** 全局函数**FN**

## 1、变量不需要定义类型，以命名来进行区分（变量命名不区分大小写）



## 2、函数命名规范“fn\_命名\_返回类型”



## 3、undefined 没有定义的物体

## 4、“$”代表当前选择的物件，通常用于调试

showproperties $

## 5、Collection Types ：Selection、Objects等

# 四、数组 Arrays

## 1、array

**#(<expr> { ,<expr> } )**

**#() --empty array**

常用函数：**append <array> <value>**

数组的index不是从0开始，从1开始计数

## 2、bitarray

**bitarray #{ <selection> }**

bitarray通常会和array互相转换使用

# 五、作业

作业：使用MaxScript创建mcr调用mse

作业：角色检查工具

检查选择的Poly：

1、每条需求做成一个ms文件，使用fn函数，加密mse

2、做一个mcr文件，调用mse