



万水电脑创意设计精品丛书

Flash MIX 编程魔法石

于立轩 主编
杨旗工作室 策划



中国水利水电出版社
www.waterpub.com.cn

万水电脑创意设计精品丛书

Flash MX 编程魔法石

于立轩 主编

杨旗工作室 策划

中国水利水电出版社

内 容 提 要

在 Macromedia 公司出品的 Flash MX 制作软件中,大大地加强了程序对 Flash 作品的控制,本书全面、详细地分析了 Flash MX 使用的 ActionScript 脚本程序设计语言的每条语句以及语句的具体应用,本书最主要的特点就是其讲解内容的详尽性。

全书共分 8 章,讲述的内容主要包括:Flash MX 编程基础、ActionScript 编辑器的使用、ActionScript 的语法规则、ActionScript 的运算符、Actions (动作) 语句、函数 (Functions)、属性 (Properties)、对象 (Objects)、Flash UI 组件 (Components)、实用案例解析。

对于 Flash MX ActionScript 的初学者,通过本书可以轻松掌握 ActionScript 脚本程序的编制与实战应用;对于 Flash MX ActionScript 的中高级用户,可以将本书作为 ActionScript 语句的速查手册,还可以从本书中学习一些更深层次的程序设计理念及 Flash MX 编程的技巧。

图书在版编目 (CIP) 数据

Flash MX 编程魔法石/于立轩主编. —北京:中国水利水电出版社, 2003
(万水电脑创意设计精品丛书)

ISBN 7-5084-1385-7

I. F… II. 于… III. 动画—设计—图形软件, Flash MX IV. TP391.41

中国版本图书馆 CIP 数据核字 (2003) 第 007828 号

书 名	Flash MX 编程魔法石
主 编	于立轩
出版、发行	中国水利水电出版社 (北京市三里河路 6 号 100044) 网址: www.waterpub.com.cn E-mail: mchannel@public3.bta.net.cn (万水) sale@waterpub.com.cn 电话: (010) 68202266 (总机)、68331835 (发行部)、68359168 (万水)
经 售	全国各地新华书店
排 版	北京万水电子信息技术有限公司
印 刷	北京市天竺颖华印刷厂
规 格	787×1000 毫米 16 开本 19.5 印张 426 千字
版 次	2003 年 3 月第一版 2003 年 3 月北京第一次印刷
印 数	0001—5000 册
定 价	28.00 元

凡购买我社图书,如有缺页、倒页、脱页的,本社发行部负责调换

版权所有·侵权必究

前 言

Flash 电影以其炫目的动画特效加上特有的娱乐体验，已经令无数网迷和追随者为之痴狂。在网络上，大多数 Flash 电影的爱好者都喜欢自己制作一些 Flash 作品，并且为作品设计一些独特的风格，而且每个人都力争把自己的作品做到最好。在创作过程中有时候会遇到一些技术上比较复杂的问题，这时如果在 Flash 电影作品中恰当地使用一些 ActionScript 语句可能会很简单地就把这些问题解决了。因为 Flash 的 Action 程序设计语言为我们提供了非常强大的程序设计功能，通过一些很简单的操作就能够达到一些非常复杂的效果。尤其是在 Flash MX 版本推出以后，Action 编程的功能更加强大，不但在面向对象编程方面增添了许多实用的功能，在网络通信方面也提供了十分强大的支持，而且在 Flash MX 中还为用户提供了一些用户界面组件（Flash UI Components），从而使程序的设计更加方便快捷。在 Flash 中，程序脚本主要使用 ActionScript 编辑器来编制。ActionScript 是 Flash 的内部语言，它可以在 Flash 电影中实现各种非常实用的交互功能，使作品可以与用户进行沟通，从而使 Flash 用户不再只是一个旁观者，而是作为一个参与者加入到游戏中来，与此同时也就为 Flash 作品赋予了灵魂。在作品中使用的 ActionScript 可以通过捕捉用户的意愿来控制电影某些特定动作的执行，例如可以通过监测由用户操纵的鼠标事件和键盘事件来控制电影的播放。

本书共分 8 章，对 ActionScript 编辑器中涉及的所有语句均作了全面详细的分析。为了使初学者能够解读早期 Flash 版本中使用的 Action 语句，在本书中对那些已经不建议使用的旧版本中的语句以及其更新的语句格式都作了特别的注明。另一方面，对于 Flash MX 版本中新增的各项功能也作了十分详尽的说明。为了使读者能够更好地理解 ActionScript 语句在 Flash 电影中的作用，在阐述其功能的同时还配备了大量简单易懂的实例以表述其在实战中的具体应用。参照书中的讲解和实例，读者可以很轻松地掌握 Flash 编程的技巧，创作出大家都满意的好作品。本书讲述的内容主要包括：

- Flash MX 编程基础
- ActionScript 的运算符
- Actions 动作语句
- 函数（Functions）
- 属性（Properties）
- 对象（Objects）
- Flash UI 组件（Components）
- 实用案例解析

本书由杨旗工作室策划，于立轩担任主编，参加编写的成员还有杨旗、孙颖敏、韩建东、林浩、李冬声、肖木坤、曹英俊等。

对于 Flash MX ActionScript 的初学者，通过本书可以轻松掌握 ActionScript 脚本程序的编制与实战应用；对于 Flash MX ActionScript 的中高级用户，可以将本书作为 ActionScript 语句的速查手册，还可以从本书中学习一些更深层次的程序设计理念及 Flash MX 编程的技巧。

由于时间仓促及作者水平有限，书中疏漏和错误之处在所难免，敬请读者批评指正。

编者

2003 年 1 月

目 录

第 1 章	Flash MX 编程基础	1
1.1	Flash MX 编程特点	1
1.2	ActionScript 编辑器	1
1.3	ActionScript 常用术语	5
1.4	词汇和语法规则	7
1.4.1	保留关键字	7
1.4.2	大小写规则	7
1.4.3	双引号 “ ”	7
1.4.4	下划线 “_”	8
1.4.5	分号 “;”	8
1.4.6	圆点 “.”	8
1.4.7	圆括号 “()”	8
1.4.8	方括号 “[]”	8
1.4.9	大括号 “{ }”	9
1.4.10	“//” 与 “/* */” 注释	10
1.4.11	_global 标识符	10
1.5	ActionScript 的数据类型	11
1.5.1	数值 (Number) 类型	11
1.5.2	字符串 (String) 类型	11
1.5.3	布尔 (Boolean) 类型	12
1.5.4	对象 (Object) 类型	12
1.5.5	影片剪辑 (MovieClip) 类型	13
1.5.6	未定义 (Undefined) 类型	13
1.5.7	Typeof 命令	13
1.6	常量与变量	14
1.6.1	常量	14
1.6.2	变量	14
第 2 章	ActionScript 的运算符	18
2.1	算术运算符	18
2.2	逻辑运算符	21

2.3	位运算符	22
2.4	赋值运算符	25
2.5	比较运算符	29
2.6	其他运算符	31
第 3 章	Actions 动作语句	33
3.1	影片控制命令	33
3.1.1	play 命令	33
3.1.2	stop 命令	34
3.1.3	gotoAndPlay 命令	34
3.1.4	gotoAndStop 命令	34
3.1.5	stopAllSounds 命令	35
3.1.6	on 鼠标事件	35
3.1.7	startDrag 和 stopDrag 命令	37
3.1.8	onClipEvent 命令	39
3.1.9	prevFrame 命令	40
3.1.10	nextFrame 命令	40
3.1.11	prevScene 命令	40
3.1.12	nextScene 命令	41
3.1.13	duplicateMovieClip 命令	41
3.1.14	removeMovieClip 命令	42
3.1.15	setProperty 命令	42
3.1.16	with 命令	43
3.1.17	trace 命令	44
3.2	浏览器/网络命令	45
3.2.1	fscommand 命令	46
3.2.2	getURL 命令	47
3.2.3	loadMovie 与 loadMovieNum 命令	48
3.2.4	loadVariables 与 loadVariablesNum 命令	50
3.2.5	unloadMovie 与 unloadMovieNum 命令	51
3.3	条件选择结构	51
3.3.1	if_else 单分支选择	52
3.3.2	switch 多分支选择	53
3.3.3	“?:”条件运算符	55
3.4	循环结构控制	55
3.4.1	while 与 do_while 循环	55

3.4.2	for 循环	57
3.4.3	for.in 循环	58
3.5	转移语句	59
3.5.1	break 命令	59
3.5.2	continue 命令	60
3.6	打印控制命令	60
3.6.1	print 与 printAsBitmap 命令	60
3.6.2	printNum 与 printAsBitmapNum 命令	62
3.7	其他命令	63
3.7.1	#include 命令	63
3.7.2	#initclip 命令	64
3.7.3	#endinitclip 命令	64
3.7.4	evaluate 命令	64
3.7.5	setInterval 命令	64
3.7.6	clearInterval 命令	66
3.7.7	ifFrameLoaded 命令	66
3.7.8	tellTarget 命令	66
3.7.9	toggleHighQuality()	67
第 4 章	函数	68
4.1	自定义函数	68
4.1.1	function 命令	69
4.1.2	return 命令	69
4.1.3	call 命令	70
4.2	类型转换函数	70
4.2.1	Array()函数	71
4.2.2	Object()函数	72
4.2.3	Boolean()函数	72
4.2.4	String()函数	72
4.2.5	Number()函数	73
4.3	数学函数	74
4.3.1	isNaN()函数	74
4.3.2	isFinite()函数	75
4.3.3	parseInt()函数	76
4.3.4	parseFloat()函数	77
4.3.5	int()函数	78

4.4	取值函数	79
4.4.1	eval()函数	79
4.4.2	getProperty()函数	80
4.4.3	getTimer()函数	80
4.4.4	getVersion()函数	81
4.4.5	targetPath()函数	81
4.4.6	escape()函数	81
4.4.7	unescape()函数	82
4.4.8	random()函数	82
4.5	字符串函数	83
第 5 章	属性	86
5.1	setProperty 命令	86
5.2	getProperty 命令	87
5.3	_url 属性	87
5.4	_target 属性	87
5.5	_droptarget 属性	88
5.6	_name 属性	89
5.7	_totalframes 属性	90
5.8	_framesloaded 属性	90
5.9	_currentframe 属性	90
5.10	_quality 属性	90
5.11	_focusrect 属性	91
5.12	_soundbuftime 属性	91
5.13	_height 属性	91
5.14	_width 属性	92
5.15	_alpha 属性	93
5.16	_visible 属性	93
5.17	_rotation 属性	95
5.18	_x 属性和 _y 属性	95
5.19	_xscale 属性和 _yscale 属性	96
5.20	_xmouse 属性和 _ymouse 属性	97
5.21	_highquality 属性	97
5.22	.scroll 属性	97
5.23	.maxscroll 属性	98
第 6 章	对象	99

6.1	核心对象	99
6.1.1	_global	100
6.1.2	this.....	100
6.1.3	super.....	101
6.1.4	Date 对象	101
6.1.5	Math 对象	106
6.1.6	Array 对象	107
6.1.7	Number 对象	116
6.1.8	String 对象	118
6.1.9	Boolean 对象	126
6.1.10	Object 对象	128
6.1.11	Function 对象	132
6.1.12	Arguments 对象	133
6.2	影片对象	134
6.2.1	_root 属性	135
6.2.2	_parent 属性	135
6.2.3	_level 属性	136
6.2.4	Key 对象	136
6.2.5	Mouse 对象	139
6.2.6	Button 对象	142
6.2.7	MovieClip 对象	148
6.2.8	Color 对象	166
6.2.9	Sound 对象	167
6.2.10	Stage 对象	171
6.2.11	TextField 对象	174
6.2.12	TextFormat 对象	185
6.2.13	Selection 对象	187
6.2.14	System 对象	189
6.2.15	Accessibility 对象	191
6.3	客户机/服务器对象	191
6.3.1	LoadVars 对象	192
6.3.2	XML 对象	194
6.3.3	XMLSocket 对象	200
第 7 章	Flash UI 组件	203
7.1	复选框组件	204

7.2	组合框组件	208
7.3	列表框组件	213
7.4	普通按钮组件	223
7.5	单选按钮组件	225
7.6	滚动条组件	229
7.7	滚动窗组件	232
7.8	组件样式格式	234
第 8 章	实用案例解析	243
8.1	打字效果输出文本	243
8.2	文字特效	246
8.3	鼠标跟随文字	251
8.4	鼠标跟随元素	256
8.5	反弹运动	260
8.6	受控的反弹运动	265
8.7	碰撞飞碟	271
8.8	拼图游戏	278
8.9	用滚动窗组件制作拼图	284
8.10	时间机器	286
8.11	计算器	291
8.12	运动场上的足球	295
8.13	飘动的蜜蜂和花朵	297

第 1 章 Flash MX 编程基础

1.1 Flash MX 编程特点

Flash MX 使用内建的 Action 语句编辑程序和解释程序。程序代码可以直接在 Flash 的 ActionScript 脚本编辑器中书写，也可以在外部书写后再导入到 ActionScript 脚本编辑器中，但是导入的程序脚本必须符合 Flash 的 Action 语句的语法规则。使用 ActionScript 编制的脚本程序是 Flash 的灵魂，并且 ActionScript 编辑器也具有极其强大的功能。在 Flash 作品中添加 ActionScript 脚本，可以非常轻松地实现一些精美的 Flash 特效，同时还可以使作品实现人机交互的功能。

从 Flash 5 开始，Flash 的 ActionScript 脚本语言就开始支持面向对象的程序设计功能。所谓面向对象的程序设计，就是将一系列相关的信息组织起来，归纳在一个称为类（Class）的集合中。这些相关的信息称为这个类的方法或属性，而属于这个类的实例就是一个对象（Object）。每个对象都拥有其所属类的所有属性和方法。

Flash MX 的脚本编程技术在面向对象方面新增了许多功能，在 Flash MX 中新增了一些预定义的 Flash 对象，而且还开始引进了组件（Component）的概念。在 Flash MX 中内置了 7 个组件，包括普通按钮（PushButton）组件、单选按钮（RadioButton）组件、复选框（CheckBox）组件、列表框（ListBox）组件、组合框（ComboBox）组件、滚动条（ScrollBar）组件、滚动窗（ScrollPane）组件。这意味着闪客们可以更方便地使用更多的 Flash 用户界面组件资源。这些组件工具都是完全独立的，可以在用户间共享使用，就像 Delphi 的 VCL 控件或 VB 和 VC 的 Ocx 控件一样。

1.2 ActionScript 编辑器

在 Flash MX 的控制面板中打开“窗口”下拉菜单，单击“动作”选项，就可以激活 ActionScript 编辑器窗口；或者直接按快捷键 F9 也可以激活或者关闭 ActionScript 编辑器。在 ActionScript 编辑器的编辑区中，可以为帧、按钮、影片剪辑添加脚本程序。ActionScript 编辑器有两种程序编写模式：标准模式和专家模式。

用户可以在编辑器的控制面板右上角的功能菜单中选择或切换编辑模式，也可以单击面板右上角的“视图选项”按钮来选择或切换编辑模式，如图 1-1 所示。

使用专家模式书写和编辑程序代码时，程序员可以在编辑区域手动书写和编辑每条语句

和每个参数；使用标准模式书写和编辑 ActionScript 程序代码时，程序员是不可以手动修改编辑区的。必须通过双击动作语句的方式在编辑区中添加语句，或者使用编辑区上方的加号按钮来向编辑器中添加语句。在编辑器自动给出的参数栏中为语句添加参数；在标准模式下，使用“减号”按钮或者键盘上的 Del 键删除当前被选中的整条语句。标准模式的编辑界面如图 1-2 所示。

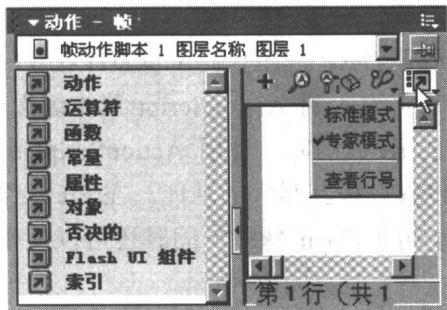


图 1-1



图 1-2

在专家编辑模式下书写程序代码对于那些比较熟练的程序员是十分方便的，但是刚刚入门的程序员使用时则比较容易产生语法错误；而使用标准编辑模式来书写 ActionScript 代码则较少发生语法错误，还可以十分方便地输入语句的参数，只是书写语句时比较烦琐一些。

在 Flash MX 中，程序源代码的编辑与调试更加方便快捷。在 Flash MX 的 ActionScript 中，增添了代码提示功能。当输入一个语句时，Flash 脚本编辑器会自动给出语句的书写格式、命令参数等提示信息，在很大程度上减少了语法错误的出现；当设置一个 on 键盘事件或者 onClipEvent 影片剪辑事件时，ActionScript 编辑器会自动给出事件列表供程序员选择，这也在某种程度上减轻了程序员的工作强度。

例：

当开始书写 for 语句的语句体时，ActionScript 编辑器立即弹出一个提示框，给出 for 语句的书写格式、输入参数等信息。在 ActionScript 编辑器的提示框中，包含在方括号 [] 中的参数表示其是一个可选参数。本例中提到的“1 属于 2”表示这条语句有两种书写格式，当前使用的是第一种格式，如图 1-3 所示。

当使用 onClipEvent 语句开始设置影片剪辑事件时，ActionScript 编辑器立即弹出影片剪辑事件列表，程序员可以直接选择影片剪辑事件，如图 1-4 所示。

在标准编辑模式下，在编辑区的左上角有一个“+”添加按钮，单击按钮可以通过选择为编辑区中添加一条语句；还有一个“-”删除按钮可以删除编辑区中当前选定的语句；在专家编辑模式下，只有“+”添加按钮而没有“-”删除按钮。图 1-5 所示就是专家模式下的添加按钮。

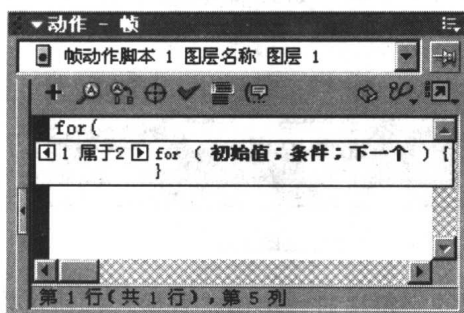


图 1-3

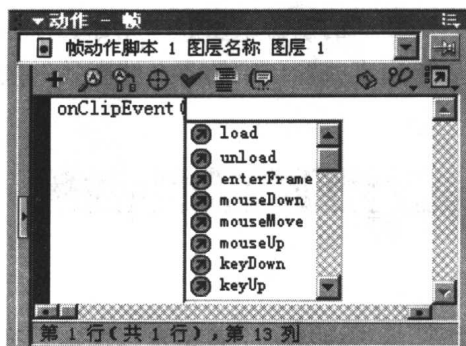


图 1-4

在标准编辑模式下，单击“-”删除按钮时会删除当前被选定的语句；在专家编辑模式中则没有“-”按钮的设置，如图 1-6 所示。



图 1-5



图 1-6

无论是使用标准编辑模式还是使用专家编辑模式，都可以使用“+”添加按钮来添加语句，如图 1-7 所示。

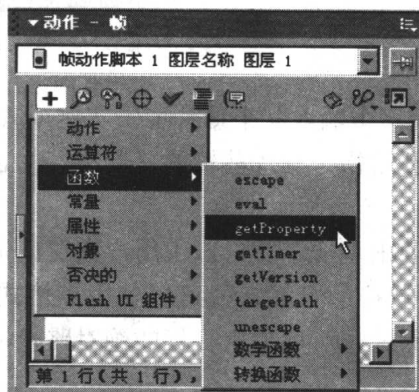


图 1-7

Flash MX 的 ActionScript 编辑器可以支持程序代码的查找和替换功能，这就为程序代码的编辑提供了很大的方便，尤其是在需要替换更改大量的信息时。例如当需要在脚本中修改一个变量名时，如果这个变量名在脚本中使用了上百次，那么使用替换功能会非常有效地提高工作效率。ActionScript 编辑器的查找和替换功能按钮放置在编辑区域的上方，如图 1-8 所示。

单击“查找”按钮后会弹出一个窗口，如图 1-9 所示，供用户输入要查找的字符串，当

查找到一个相同的字符串以后，光标会停在这个字符串上面，如果需要继续查找下一个字符串，可以单击查找窗口中的“继续查找”按钮。

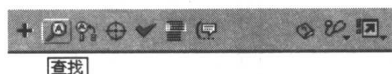


图 1-8

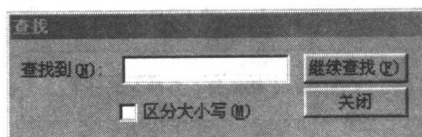


图 1-9

当需要将查找到的字符串选择性地替换为给定的字符串时，单击“替换”按钮可以在当前的程序代码中查找并替换这些字符串，如图 1-10 所示。

单击“替换”按钮后会出现一个替换窗口，如图 1-11 所示，用户在窗口中输入需要查找的字符串和替换字符串，当找到一个查找的字符串时，光标会停在这个字符串上面，如果需要使用替换字符串替换它，单击“替换”按钮，如果不想替换它，单击“继续查找”按钮继续查找下一个字符串。可以单击“全部替换”按钮将查找到的字符串全部替换为给定的替换字符串。



图 1-10

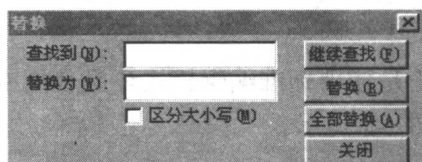


图 1-11

在编写代码时，当需要在程序中输入某个影片剪辑实例或者按钮实例的路径时，使用“目标路径”按钮可以非常方便地获得指定影片剪辑的目标路径并且在指定的位置输入所选择实例的路径，如图 1-12 所示。

单击“目标路径”按钮后会出现一个窗口，由用户在窗口中显示的路径结构中选择指定的影片剪辑以获得目标路径，获得的目标路径可以使用点格式表示也可以使用斜杠格式表示，可以使用目标实例的相对路径也可以使用目标的绝对路径，使用圆点格式时的设置如图 1-13 所示。



图 1-12



图 1-13

使用斜杠格式表示目标路径时的设置如图 1-14 所示。

在编辑区右上角有一个“脚本参考”按钮，单击这个按钮可以提供语句使用的帮助信息。在选中某条 ActionScript 语句时，单击这个按钮可以获得关于这个语句的一些参考信息，帮助用户使用这些 ActionScript 语句。“脚本参考”按钮如图 1-15 所示。

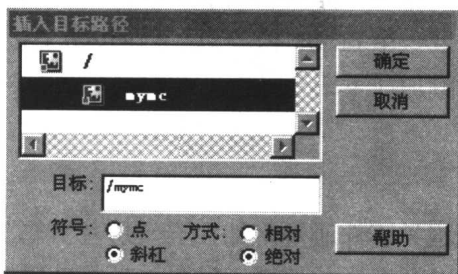


图 1-14



图 1-15

1.3 ActionScript 常用术语

1. 常量

常量 (Constants) 就是在程序执行的过程中其值不改变的量，它可以是一个具体的数，也可以是一个符号，即符号常量。

2. 变量

变量 (Variables) 是一个定义了的标识符，这个标识符在内存中定义了一定的内存空间，为变量赋值就是将赋予的值存储到变量标识符所定义的内存空间中。在程序的执行过程中可以根据需要重新为变量赋值。

3. 运算符

运算符 (Operators) 又被称作操作符，使用运算符可以对 Flash 中定义的不同类型的数据执行不同的运算操作。在 Flash MX 中提供了算术运算符、逻辑运算符、位运算符、比较运算符、赋值运算符等多种运算符。

4. 表达式

表达式 (Expressions) 是由常量、变量和操作符组合而成的一个式子，可以使用 Flash MX 提供的运算操作符对不同类型的表达式进行计算求值。在语句中表达式通常使用圆括号()括起来。

5. 动作

动作 (Actions) 是一条语句的主要部分，正是这一部分决定了这条语句将执行什么样的操作。再加上为这个动作设置的动作参数，就组成了一个完整的 ActionScript 语句，可以完成一项具体的功能。在本书中，也形象地把动作称为命令。

6. 事件

在 Flash MX 中, 事件 (Events) 主要有两种。一种是由 on 命令控制的事件, on 命令控制的事件处理程序可以附加在一个按钮 (Button) 或者影片剪辑 (MovieClip) 上面, 当鼠标在这个按钮或者影片剪辑的区域发生不同操作时, 或者键盘的某个按键被按下时, 就立即触发一个 on 事件并执行为这个事件事先设置的处理程序。而需要附加在按钮上面的代码就是这个事件和事件处理语句。例如可以为按钮设置一个 on 事件, 当鼠标在按钮上按下时, 立即触发执行指定的事件处理语句, 可以在按钮上附加如下语句:

```
on(press){  
    trace("the button be pressed! ");  
}
```

还有一种事件是 onClipEvent 影片剪辑事件, 这类事件通常附加在一个影片剪辑上面。影片剪辑事件包括影片剪辑的载入、卸载影片剪辑以及在影片剪辑上发生的鼠标或键盘事件等。当指定的影片剪辑上面发生了影片剪辑事件后, 如果在这个影片剪辑的脚本里面设置了 onClipEvent 命令, 就立即执行事先设置的事件处理命令。

7. 事件处理程序

当发生某个事件时, 就立即调用这个事件的事件处理程序 (Handlers), 如果没有设置这个事件的事件处理程序, 那么发生事件后不执行任何操作。

8. 函数

函数 (Functions) 是可以实现一项具体功能的程序段, 使用函数称为函数调用。可以在程序中不同的地方多次调用一个函数, 有些函数还需要输入一些初始值来参与函数的操作。

9. 参数

参数 (Arguments) 是为函数传递初始值的一些变量。当函数需要调用初始值时, 就使用函数的参数传递这些初始值。

10. 类

类 (Class) 是属于一个类型的所有数据的总称, 类中的每一个实例都继承这个类的所有属性, 用户需要使用构造函数来定义新的类。

11. 实例

实例 (Instances) 是属于一个具体类的对象, 它继承所属类的所有方法和属性, 但是每个实例都作为一个独立的个体而存在。

12. 对象

对象 (Objects) 是包含一定的方法和属性的类的实例, 用户可以自己构造对象也可以使用 Flash MX 预定义的对象。

13. 方法

方法 (Methods) 是对象可以使用的函数, Flash MX 中每个对象都继承其所属对象类的所有方法, 用户可以为对象添加自定义的方法。