

Macromedia 中国授权认证培训 (ATC) 专用教程

macromedia

FLASH **MX** **ActionScript**

标准教程



总策划

黄 刚 于 林 编



北京希望电子出版社
Beijing Hope Electronic Press
www.bhp.com.cn



Macromedia 中国授权认证培训 (ATC) 专用教程

macromedia®

FLASH MX ActionScript

标准教程



总策划

黄 刚 于 林 编



北京希望电子出版社
Beijing Hope Electronic Press
www.bhp.com.cn



内 容 简 介

本书详细介绍了 Macromedia 公司的 Flash MX ActionScript 的使用方法和编程技巧。

全书由 10 章、1 个附录组成, 主要内容包括: Flash MX ActionScript 基础知识、Flash MX ActionScript 动作、Flash MX ActionScript 运算符、Flash MX ActionScript 函数和常量、关于电影剪辑属性、Flash MX ActionScript 对象、按钮对象实例研究、电影剪辑对象实例研究、声音对象实例研究、制作和打包 Flash MX 作品等。附录中提供了丰富的 ActionScript 函数、常量以及对象词典。

本书结构清晰、内容由浅入深、知识全面丰富、语言精简、无专业术语、阅读感觉通俗易懂, 全书内容涉及广泛并提供了大量示例以方便读者更好的理解和领会。

对于 Flash MX ActionScript 的初学者而言, 本书可以是 Flash MX ActionScript 的快速入门指南, 而对于 Flash MX ActionScript 的中高级用户而言, 本书便是必要的备查手册。

本书既可作为 Macromedia 中国授权认证培训(ATC)专用教程, 也可作为社会 Flash MX ActionScript 培训班用教材。

本书由  授权北京希望电子出版社在中国大陆地区独家出版、发行。

本版 CD 包括 Flash MX 试用版软件及书中程序源代码。

版权所有, 侵权必究!

盘书系列名: Macromedia 中国授权认证培训(ATC)专用教程

盘 书 名: Macromedia Flash MX ActionScript 标准教程

总 策 划: Macromedia

文本著作者: 黄 刚 于 林 编

责 任 编 辑: 马宏华

CD 制 作 者: 希望多媒体开发中心

CD 测 试 者: 希望多媒体测试部

出版、发行者: 北京希望电子出版社

地 址: 北京市海淀区知春路 63 号 (100080)

网址: www.bhp.com.cn E-mail: lwm@hope.com.cn

电话: 010-62520290, 62521724, 62528991, 62630301, 62524940, 62521921, 82610344 (发行)
010-82675588-202 (门市) 010-82675588-501, 82675588-201 (编辑部)

经 销: 各地新华书店、软件连锁店

排 版: 希望图书输出中心 全卫

CD 生 产 者: 北京中新联光盘有限责任公司

文本印刷者: 北京双青印刷厂

开本 / 规格: 787 毫米×1092 毫米 1/16 开本 48 印张 870 千字

版次 / 印次: 2002 年 8 月第 1 版 2002 年 8 月第 1 次印刷

印 数: 0 001-3 000 册

本 版 号: ISBN 7-900118-41-1

定 价: 66.00 元 (本版 CD)

说明: 凡我社售出的产品如有残缺, 可执相关凭证与我社调换。

前 言

Macromedia 公司总部设立在美国加州三番市, 在全球 50 多个国家设有经营机构。Macromedia 公司在全世界拥有 300 万开发和设计用户以及广大的行业合作伙伴网络, 其丰富的客户机软件被 98% 的 Web 应用开发人员所广泛使用, 是企业、政府和教育市场客户的战略性 IT 提供商。

Macromedia 公司的主要软件产品包括: Macromedia Authorware, Macromedia ColdFusion, Macromedia Director, Macromedia Dreamweaver, Macromedia Fireworks, Macromedia Flash, Macromedia Flash Player, Macromedia FreeHand 和 Macromedia JRun 等。Macromedia 公司的软件产品被全球数以百万计的开发人员和设计人员所使用, 为他们提供了高效地创造互联网上最为有效的用户体验。其集成的工具、服务器和客户机技术家族支持提供从网站到 Rich Internet Applications 的跨多种平台和设备的广泛 Internet 解决方案。

在全球信息产业迅猛发展的今天, Macromedia 公司把发展的重点转向以中国大陆地区为主体的亚洲市场。从 1994 年正式进入中国市场以来, Macromedia 公司通过广泛的用户群体和行业合作伙伴将 Macromedia 产品的应用拓展到网上出版、多媒体开发、图形图像处理、远程视频、多平台发布等领域, 并逐步确定了 Macromedia 产品在中国 Internet 开发和互动多媒体领域的领先地位。

为了全面、系统地提升网络设计、开发者和多媒体设计者的应用水平, Macromedia 公司特别建立“Macromedia 中国授权认证培训体系”, 并设立了“Macromedia 中国授权认证培训管理中心 (Macromedia ATMC)”、“Macromedia 中国授权认证培训中心 (Macromedia ATC)”、“Macromedia 中国授权认证考试管理中心 (Macromedia AXMC)”和“Macromedia 中国授权认证考试中心 (Macromedia AXC), 为网络设计、开发者、交互式多媒体设计者和有兴趣进入这一领域的爱好者提供专业的培训和授权认证考试服务。

为确保向参加 Macromedia 中国授权认证培训的学员提供最完善的教学服务, Macromedia 公司制定了一系列管理、教学、服务制度规范, 以形成一整套严谨的教学培训管理体制。目前全国共设有百余家“Macromedia 中国授权认证培训中心 (Macromedia ATC)”, 及若干家

“Macromedia 中国授权认证考试中心 (Macromedia AXC)”，学员或考生可以方便地根据自己的所在地选择授课、考试地点，接受统一完备的教学服务。

此次出版的《Macromedia 中国授权认证专用教程》，是惟一由 Macromedia 授权使用的培训教程，由 Macromedia 组织技术专家精心编写，并且它是专门针对教学培训计划而编写的，注重教学课程的科学性和可操作性，教程结构清晰、由浅入深、循序渐进，知识面丰富，语言精简、无专业术语、阅读感觉通俗易懂，大量的图文结合为读者节省了阅读时间、降低了学习难度，内容涉及面广并提供大量实例以方便读者的更好理解和领会，希望这套教程能对学习者提供帮助与指导，如果在学习过程中遇到任何问题，请访问我们的网站 (<http://www.macromedia.com/cn>) 得到技术方面的支持，或将你对本套教程的意见与建议发至 contact@macromediachina.com。



2002 年 7 月



Hotbook好书推荐

Macromedia 中国认证培训 (ATC)专用教程



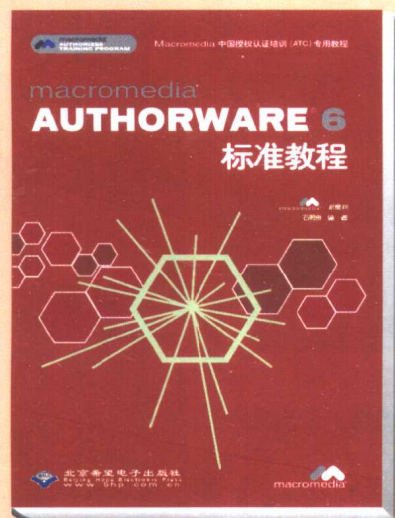
CX-83573
定价: 20 元
ISBN: 7980008359



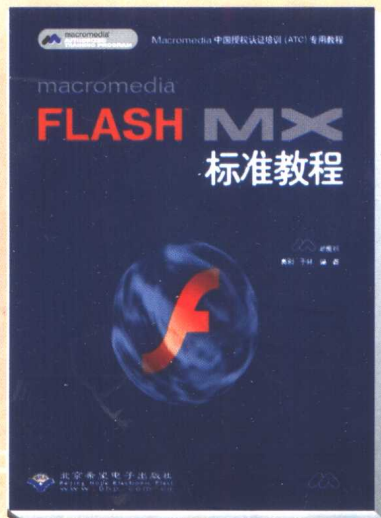
CX-83574
定价: 20 元
ISBN: 798000843x



CX-83575
定价: 26 元
ISBN: 7980008499



CX-83680
定价: 30 元
ISBN: 7900088601



CX-3768
定价: 39.00 元
ISBN: 7900101055



CX-3899
定价: 35.00 元
ISBN: 7900118330



北京希望电子出版社
Beijing Hope Electronic Press
www.bhpe.com.cn

社址: 北京海淀知春路 63 号卫星大厦三层
通信地址: 北京中关村 083 信箱北京希望电子出版社 (邮编 100080)
联系电话: 010-82675588(总机) 传真: 010-62520573



Hotbook 好书推荐

您也想成为平面设计和三维动画大师吗? 请选用下列教材



CX-93737 定价: 48 元
(5 小时电脑动画电影 6VCD)
ISRC: CN970200010



CX-3988
定价: 38 元
ISBN: 7900101276



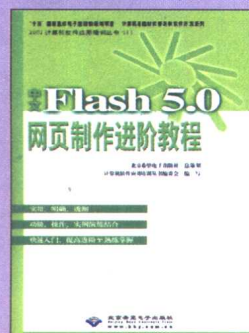
CX-3788
定价: 38 元
ISBN: 7900088385



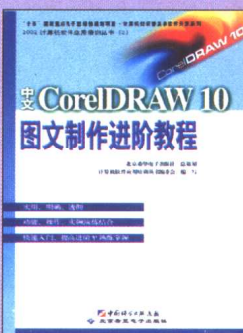
CX-83639
定价: 168 元(精装)
ISBN: 7900088393



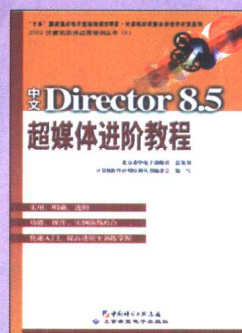
CX-3765
定价: 58.00 元
ISBN: 7900101071



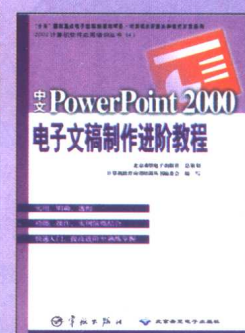
CX-83693
定价: 25.00 元
ISBN: 7900088741



CX-3701
定价: 25.00 元
ISBN: 7900088822



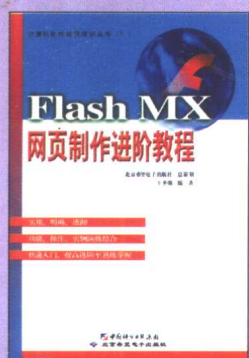
CX-3729
定价: 30.00 元
ISBN: 7900088946



CX-3722
定价: 26.00 元
ISBN: 7800348016



CX-3787
定价: 35.00 元
ISBN: 790010125X



CX-3771
定价: 25.00 元
ISBN: 7900101101



CX-83371
定价: 50.00 元
ISBN: 7900071288



CX-83544
定价: 39.00 元
ISBN: 7980007557



CX-83545
定价: 35.00 元
ISBN: 7980007581



CX-83586
定价: 33.00 元
ISBN: 7900088008



北京希望电子出版社
Beijing Hope Electronic Press
www.bhpe.com.cn

社址: 北京海淀知春路 63 号卫星大厦三层
通信地址: 北京中关村 083 信箱北京希望电子出版社 (邮编 100080)
联系电话: 010-82675588(总机) 传真: 010-62520573

目 录

第 1 章 Flash MX ActionScript 基础知识	1
1.1 关于 ActionScript	1
1.1.1 关于规划和调试脚本	2
1.1.2 关于面向对象的脚本编程	2
1.1.3 关于 MovieClip 对象	4
1.1.4 脚本流程介绍	5
1.1.5 ActionScript 运行时的控制	5
1.2 ActionScript 基本术语解释	7
1.3 ActionScript 语法基础	10
1.3.1 小点语法	10
1.3.2 大括号	10
1.3.3 分号	11
1.3.4 圆括号	11
1.3.5 大小写字母	11
1.3.6 批注	12
1.3.7 关键字	12
1.3.8 常量	13
1.4 ActionScript 中的数据类型	13
1.4.1 字符串值	13
1.4.2 数值	14
1.4.3 逻辑值	14
1.4.4 对象	15
1.4.5 电影剪辑	15
1.4.6 空值	16
1.4.7 未定义值	16
1.5 ActionScript 中的变量	16
1.5.1 命名变量	16
1.5.2 输入变量	16
1.5.3 设置变量使用范围	17
1.5.4 申明变量	18
1.5.5 在脚本中使用变量	18

1.6	ActionScript 中的运算符	20
1.6.1	运算符的优先级	20
1.6.2	运算符的结合性	20
1.6.3	算术运算符	20
1.6.4	比较运算符	21
1.6.5	字符串运算符	21
1.6.6	逻辑运算符	22
1.6.7	相等运算符	22
1.6.8	赋值运算符	23
1.6.9	小点和数组访问运算符	23
1.7	使用动作实现对象控制	24
1.8	通过路径指定目标对象	25
1.9	控制脚本流程	26
1.9.1	检查条件	26
1.9.2	重复动作	26
1.10	调用和创建函数	27
1.10.1	调用函数	28
1.10.2	定义函数	29
1.10.3	向函数传递参数	29
1.10.4	在函数中使用变量	30
1.10.5	从函数返回值	30
1.11	Flash MX ActionScript 开发环境介绍	30
1.11.1	关于 Actions (动作) 面板	31
1.11.2	普通模式	34
1.11.3	专家模式	38
1.11.4	使用参考书面板	41
1.11.5	动作工具箱深入研究	43
1.11.6	自定义动作面板中的文本显示	49
1.11.7	提示区域	51
1.11.8	跳转栏	51
1.11.9	流畅的动作面板工作环境	52
第 2 章	Flash MX ActionScript 动作	53
2.1	可附加动作的载体	53

2.1.1	帧.....	54
2.1.2	按钮.....	54
2.1.3	电影剪辑.....	55
2.2	帧和场景的控制.....	55
2.2.1	goto 动作示例.....	56
2.2.2	play 动作示例.....	57
2.2.3	stop 动作示例.....	58
2.3	和浏览器/网络相关的动作.....	58
2.3.1	getURL 动作的用法.....	59
2.3.2	载入和卸载电影示例.....	60
2.4	电影剪辑控制.....	64
2.4.1	复制/删除电影剪辑示例.....	65
2.4.2	设置电影剪辑属性示例.....	66
2.4.3	拖动/停止拖动电影剪辑示例.....	71
2.5	设置变量动作示例.....	74
2.6	条件和循环示例.....	80
2.7	自定义及调用函数示例.....	87
2.8	其他动作.....	92
2.8.1	设置/清除时间间隔示例.....	93
2.8.2	追踪语句结果示例.....	97
2.8.3	设置脚本批注示例.....	97
第 3 章	Flash MX ActionScript 运算符.....	99
3.1	运算符分类示例.....	99
3.1.1	算术运算符.....	100
3.1.2	赋值运算符.....	101
3.1.3	比较运算符.....	102
3.1.4	逻辑运算符.....	105
3.1.5	其他运算符.....	107
3.2	运算符综合应用示例.....	108
3.2.1	准备制作游戏的素材.....	108
3.2.2	编写脚本控制游戏的运行.....	111
3.2.3	初始化游戏的运行环境.....	115
3.2.4	判断游戏的结果.....	119

第 4 章	Flash MX ActionScript 函数和常量	121
4.1	转换函数	123
4.2	获取时间示例	124
第 5 章	关于电影剪辑属性	130
5.1	电影剪辑属性详解	130
5.2	电影剪辑属性的利用示例	132
第 6 章	Flash MX ActionScript 对象	138
6.1	核心对象	138
6.1.1	日期对象	140
6.1.2	字符串对象	145
6.2	电影对象	151
6.2.1	颜色对象	151
6.2.2	按键对象	155
6.2.3	鼠标对象	162
6.2.4	文本域对象	164
第 7 章	按钮对象实例研究	167
7.1	按钮事件综述及示例	167
7.2	按钮属性示例	173
第 8 章	电影剪辑对象实例研究	178
8.1	电影剪辑对象的属性和事件	178
8.2	电影剪辑对象的方法示例	181
8.2.1	电影剪辑对象方法综述	181
8.2.2	迷宫示例	186
8.2.3	动态放大镜示例	192
8.3	电影剪辑对象的绘图方法	202
第 9 章	声音对象实例研究	207
9.1	声音对象的方法、属性和事件综述	207
9.2	在电影中动态载入 MP3 示例	210
9.3	制作 Flash 声音播放控件示例	213
第 10 章	制作和打包 Flash MX 作品	222
10.1	Flash 作品的版本和格式	222
10.2	制作 Flash MX 多媒体光盘	223
附录 A	ActionScript 词典	227

第1章 Flash MX ActionScript 基础知识

1.1 关于 ActionScript

Macromedia Flash 软件的用户都知道,要完成好的 Flash 作品,需要有两方面的能力。一是美工基础,二是编程技巧。在早期的 Flash 版本中,多注重的是图形图像方面的设计,脚本功能也比较简单,因此,没有编程基础的 Flash 用户,如果有很好的绘图能力,也可以凭借 Flash 软件的优势创造出很优秀的作品。但是,自从 Flash 5 推出 ActionScript 之后,编程技巧受到越来越多的重视。Flash 软件功能本身的主导方向也从以前的“网络动画解决方案”变成了 Flash MX 的“丰富客户机技术”。这一技术提倡的是跨平台和跨系统的可移植性、更强大的交互功能和更良好的数据通信处理。有鉴于此,Flash ActionScript 一经推出便广受关注,到 Flash MX ActionScript 推出时,它已经变得更加稳定,功能也更加强大,成为 Flash 用户必不可少的工具。

到目前为止,Flash 软件用户已经逐渐分裂为两种类型:一类是比较偏重于美工绘图的 Flash 设计人员,另外一类是偏重于编程实现的 Flash 开发人员。在制作诸如 Flash MTV 音乐动画这样的作品时,由于没有太多的交互功能和编程需求,美工基础好的 Flash 设计人员可以制作出很好的作品;而在建立诸如网络酒店预订系统这样的 Flash 项目时,需要开发人员具有更多的 ActionScript 编程技巧,使得这样的 Flash 项目既具有引人注目的外观,又具有强大的功能。

美术绘图能力和编程技巧并不是相互排斥的。相反,它们是一种互补的关系。同样地,Flash 设计人员和开发人员也并没有天然的鸿沟,开发人员可以通过 Flash 强大的绘图功能学习一些绘图技巧,而设计人员也可以通过学习 ActionScript 而向开发人员迈进。

ActionScript 的语法和样式和 JavaScript 非常类似。所以,ActionScript 是一种容易学习的语言。本书将为用户学习这种语言提供极大的帮助。

ActionScript 是一种工具,能帮助用户按照自己的意志更加精确地创建电影,实现自己的意图。用户不必等到完全理解每一个 ActionScript 元素才开始编写脚本。如果有一个清晰的目标,用户就可以开始使用简单动作编写自己的脚本,边干边学,直至完成复杂的任务。

像其他脚本语言一样,ActionScript 也有自己的语法规则、保留关键

字、提供运算符，并且允许用户使用变量存储和检索信息。ActionScript 包含内置对象和函数，也允许用户创建自己的对象和函数。本章将介绍有关 ActionScript 的各种基础知识，而在本书的其他各章中将详细介绍 Flash MX ActionScript 的所有条目。对于 ActionScript 的初学者而言，它可以是 ActionScript 的快速入门指南，而对于 ActionScript 的中高级用户而言，它可以是必要的备查手册。

1.1.1 关于规划和调试脚本

在为整个电影编写脚本时，脚本的数量和变化可能都是巨大的。例如，用户可能需要考虑以下问题：决定使用哪些动作，如何建立更有效的脚本结构，以及在哪些地方放置脚本？所有这些问题都需要经过仔细规划和测试。特别是当电影变得越来越复杂时，这些问题更加显得突出。

在开始编写脚本之前，用户需明确自己要达到的目标。例如，这些目标可能如下：

- 使用 Flash 创建整个站点。
- 询问站点访问者的姓名，并且在整个站点中重复使用该信息。
- 站点具有一个可拖动的导航工具栏，可以帮助用户跳转到站点的任意分支。
- 当某一个按钮被单击时，新的内容以淡入方式出现在舞台中央。
- 专门设立一个和用户联系的场景，其中包含已经填入用户姓名的表单。

在明确了制作动画要达到的目的之后，用户就可以开始创建对象并编写脚本来控制这些对象。

让脚本按照自己需要的方式工作，这通常比编写脚本更花费时间。用户需要经常不断地测试自己的脚本是否工作正常。为达到目的，最好的方法是从简单脚本开始，频繁测试，编写并测试完一部分可正常工作的脚本之后，就可以保存文件的一个版本，例如，myMovie01.flx，然后接着编写下一段脚本。这种工作方法可以帮助用户更有效地调试脚本错误，在编写更加复杂的脚本时也有坚实的基础。

1.1.2 关于面向对象的脚本编程

在面向对象的编程方法中，用户通过将信息分类成组来组织信息。用户可以创建某一类信息的多个实例，以便在脚本中使用。用户可以使

用 ActionScript 预定义的 class (类), 也可以创建自己的类。ActionScript 内置的类位于动作面板的 Objects (对象) 文件夹中。

在创建 class (类) 时, 用户需为由该类所创建的每个对象定义所有属性 (特性) 和方法 (行为)。这和现实世界中对象的定义方法一样。例如, “人” 具有诸如性别、身高和头发颜色之类的属性, “人” 的行为包括说话、走路、投掷等, 那么在本叙述中, “人” 就是一个 class (类), 而每个单独的人就是该类的实例, 即 Object (对象)。

ActionScript 中的对象可以包含数据, 也可以是舞台上的电影剪辑。所有电影剪辑都是预定义 MovieClip 类的实例。每个电影剪辑实例都具有 MovieClip 类的所有属性 (例如 _height, 高度; _rotation, 旋转; _totalframes, 总帧数等) 和所有方法 (例如, gotoAndPlay, 转到并播放; loadMovie, 载入电影; startDrag, 开始拖动等)。

要定义 class (类), 用户需创建一个特殊的被称为 “构造器” 的函数。Flash 预定义的类已经定义了构造器函数。例如, 如果需要电影中一个骑车人的信息, 则可以创建一个名为 Biker 的构造器函数。该函数包含 “时间” 和 “距离” 属性, 也包含 “速度” 方法, 该方法可以将骑车人的移动速度告诉用户, 如下例:

```
function Biker(t, d) {  
    this.time = t;  
    this.distance = d;  
}  
function Speed() {  
    return this.time / this.distance;  
}  
Biker.prototype.rate = Speed;
```

在编写构造器函数之后, 用户就可以创建 Biker 类的副本, 即: 实例。以下代码就可以创建 Biker 类对象的实例 emma 和 hamish:

```
emma = new Biker(30, 5);  
hamish = new Biker(40, 5);
```

实例之间还可以相互通信。例如, 对于 Biker 类的对象来说, 用户可以创建一个 shove (推) 的方法, 让一个骑车人推另外一个骑车人。这样, 当 hamish 实例靠得太近时, emma 实例就可以调用 shove 方法, 将 hamish 推开。要将信息传递给方法, 可使用参数。例如, shove (推) 方法就可以携带 who (被推者) 和 howFar (推多远) 参数, 如下例:

```
emma.shove(hamish, 10);
```

在面向对象编程方法中，类可以按指定顺序相互接收属性和方法，这就是所谓的继承。用户可以使用继承扩展或重定义类的属性和方法，从其他类继承属性和方法的类被称为“子类”，将属性和方法传递给其他类的类被称为“超类”，一个类可以同时是“子类”和“超类”。

1.1.3 关于 MovieClip 对象

ActionScript 内置的类在动作面板的动作工具箱中被称为“对象”。用户可以将某一对象看作某一类的实例。对象允许用户访问某些类型的信息。例如，Date（日期）对象包含的 `getFullYear`，`getMonth` 之类的方法就允许用户从系统时钟内读取时间信息。Sound（声音）对象包含的 `setVolume`，`setPan` 等方法就允许用户控制电影中的声音。MovieClip（电影剪辑）对象包含的 `play`，`stop` 和 `getURL` 等方法就允许用户控制电影剪辑实例，而 `_alpha`，`_framesloaded`，`_visible` 等则可以设置电影剪辑的属性信息。

MovieClip（电影剪辑）对象是 Flash 电影中最重要的对象，因为它们就是一个小型化的电影：它们有各自独立运行的时间轴。例如，如果主电影的时间轴只有一帧，而该帧中的电影剪辑包含 10 帧，则当主电影播放时，电影剪辑中的每一帧都能播放。这一特点使实例既可以相互交流，又可以以一个完全独立的对象存在。

每个电影剪辑实例都有自己的惟一名称，这样用户就可以在动作中对它们进行指定。例如，在舞台上可能包括多个实例（举个例子来说，包括 `leftclip` 和 `rightclip`），而用户只想一次播放一个，要分配动作指定某一个电影剪辑播放，则需要使用到实例的名称。例如：

```
leftClip.play();
```

使用实例名称还可以在电影播放时复制、删除或拖动电影剪辑。使用 ActionScript 可以动态设置和检索电影剪辑属性的值，改变或读取这些属性则可以改变电影剪辑的外观和特性，这也是创建电影交互性的关键。例如，以下脚本示例就使用了 `setProperty` 动作，将 `navigationBar` 实例的透明度设置为 10：

```
setProperty("navigationBar",_alpha,10);
```

Flash 电影中的按钮和文本域也是可以使用 ActionScript 控制的对象。

1.1.4 脚本流程介绍

Flash 将从第一行语句开始执行 ActionScript 语句，一直按次序执行到最终语句或 ActionScript 指定跳转到的语句。

将执行次序发送到其他地方而不是下一语句的动作包括：if 语句、do..while 循环和 return 动作。

if 语句被称为条件判断语句或“逻辑分支”，因为它是基于某些条件的评估判断来控制脚本流程的。例如，以下代码就将检查变量 mynumber 中的值是否小于或等于 10，如果检查结果返回 true（例如 mynumber 的值为 5），则变量 alert 被设置文本字符串值，并在输出的文本域中显示其值。

```
if (myNumber <=10){  
    alert ="该数字小于或等于 10";  
}
```

用户也可以添加 else 语句来创建更加复杂的条件判断语句。在以下示例中，如果条件判断返回 true（例如，变量 mynumber 的值为 3），则第一组大括号之间的语句将执行，alert 变量将显示第二行的值。如果条件判断返回 false（例如，变量 mynumber 的值为 30），则第一组代码区将被跳过，else 语句后面的大括号之间的语句将执行。

```
if (number <=10){  
    alert ="该数字小于或等于 10";  
}else {  
    alert ="该数字大于 10";  
}
```

循环语句将重复执行某一动作若干次，直到满足某一条件才结束。在以下示例中，电影剪辑将被复制 5 次。

```
i =0;  
do {  
    duplicateMovieClip ("myMovieClip","newMovieClip"+i,i);  
    newName =eval("newMovieClip"+i);  
    setProperty(newName,_x,getProperty("myMovieClip",_x)+(i *5));  
    i =i +1;  
}while (i <=5);
```

1.1.5 ActionScript 运行时的控制

在编写脚本时，用户可以使用动作面板，将脚本附加给时间轴中的

帧，也可以附加给舞台上的按钮或电影剪辑。附加给帧的脚本将在播放磁头到达该帧时执行，而附加给按钮或电影剪辑的脚本则在某一事件发生时执行。所谓的“事件”，可以是电影播放时鼠标移动、某个按钮被按下，也可以是某个电影剪辑被载入等。使用 ActionScript 可以发现事件发生的时间，然后根据该事件执行指定的脚本。

附加给按钮或电影剪辑的动作被一个特殊的动作包围，该动作被称之为 handler（处理程序），onClipEvent 和 on 动作都被称之为处理程序，是因为它们操控或管理着事件。用户可以为每个处理程序指定一个或多个事件。当由处理程序指定的事件发生时，电影剪辑或按钮动作才执行。如果要在不同的事件发生时执行不同的动作，则可以给一个对象附加多个处理程序。

onClipEvent 动作处理的是电影剪辑的事件，而 on 动作则处理按钮的事件。用户也可以和电影剪辑一起使用 on 动作，创建“按钮电影剪辑”——一个能接受鼠标事件的电影剪辑。

电影剪辑事件和按钮事件也可以通过 MovieClip 和 Button 对象的方法进行控制。

例如，如果舞台上包含一个按钮，则用户可以使用动作面板添加 trace() 动作，其代码如下：

```
on (release){
    trace("You clicked me!");
}
```

用户也可以使用方法创建相同的效果。示例如下：

```
myMovieClip.onRelease =function(){
    trace("You clicked me!");
}
```

表 1-1 列出了按钮的事件和方法。

表 1-1 按钮的事件和方法

事件	方法	说明
on(press)	onPress	鼠标左键按下
on(release)	onRelease	鼠标单击（左键按下并释放）
on(releaseOutside)	onReleaseOutside	鼠标左键按下之后在按钮之外释放
on(rollOver)	onRollOver	鼠标悬停在按钮之上
on(rollOut)	onRollOut	鼠标移动到按钮响应区域之外