

## Flash 游戏开发教程：第一节

作者：周云(cloudy1982) 来源：闪吧

闲着也是闲着，开一贴给刚刚踏上 FLASH 游戏开发的爱好者们写点东西。

FLASH 游戏这东西吧，总入不了大流。国内拿 FLASH 做美术方面的应用比较多，而传统的游戏程序员又把 FLASH 当小儿科，不屑搞这个，事实上开发 FLASH 游戏也赚不了钱，七七八八的原因加起来，就变成了没多少职业程序员来开发 FLASH 游戏，学校也不会培养学生深入学习开发 FLASH，搞这块的人少，人少交流讨论的也少，正正规规的教材也少，当然优秀作品更少。

目前国内大部分的 FLASH 游戏开发爱好者没有好教材的指引，自己摸索着就上路了（NND,我也是这样开始的），开发流程和习惯都是各人一套，很多人因为没有接受正确的学习而走了很多歪路（MD，好像我也是）

于是乎，话题转回来，我就开一贴跟菜鸟们多交流吧.....

### 游戏制作前的准备

突然灵光一闪：老子要做个 XX 类型游戏！

于是立马打开电脑，打开 FLASH，找图片，建元件，F9一按，代码蹭蹭的打啊.....结果做了一半做不下去了。

以上情况常常发生.....

做游戏首先脑子里先要有整个游戏的规划，最好是拿笔画个流程简图，然后再下手。

游戏流程的规划是很重要的，我在做一个游戏前，草稿纸上来来回回要打个半天的草稿才开工，我一直认为游戏的规划部分是很难的，如果真等到全部想好了，做起来其实是个很轻松的过程，只是耗时间罢了。

理论知识多说没用.....

我们不如来实践下，跟我一起做简单的游戏吧.....

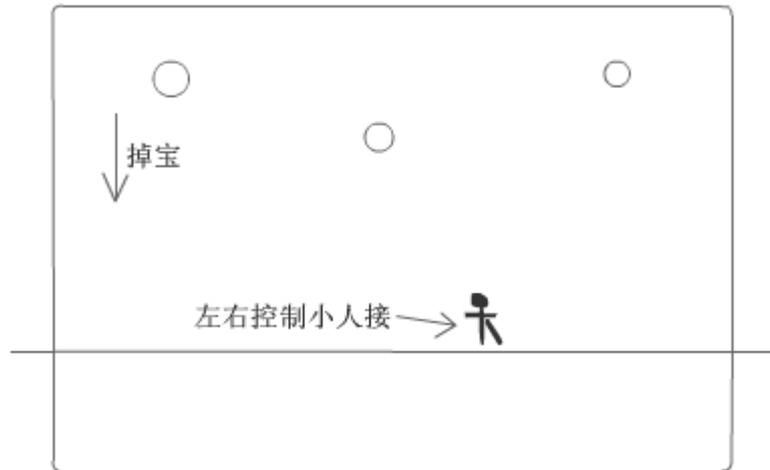
吃金币游戏-策划（我在草稿纸头上写下以下几点）

基本描述：

天上掉宝物，地下小人物由自己控制，去接宝得分

时间限制30秒

写到这里，顺手画了张草图



简单示意草图1

画好草图，我继续在草稿纸上又写下详细的几处描述： 网页教学网

宝物分四种：金币10、金币50、增加时间的宝物、瞬间获得屏幕所有金币的宝物

要有两个人能选，一个人物太无聊

人物样子不能太傻（这条请忽略）

最好来点技能~！（晕- -不是说简单游戏么.....）

要能记录我最高得分，有点挑战性！

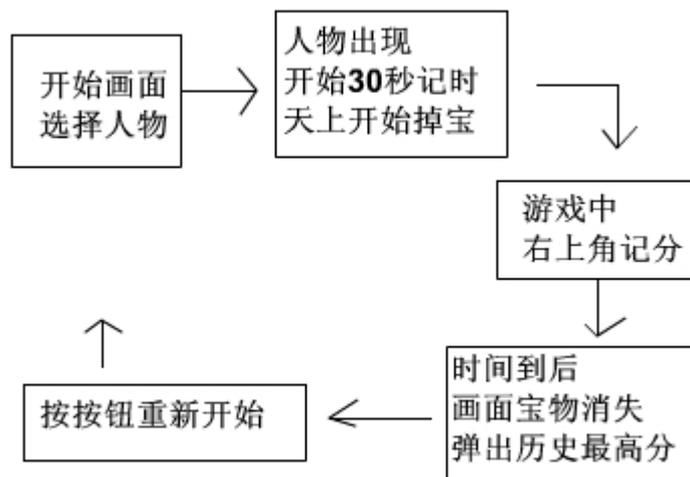
玩到后面最好难度慢慢增加，就让下落物品速度加快吧！

即使小游戏，音乐音效也应该有的

（占位中...）

写到这里，我又想了想，又画了张草图

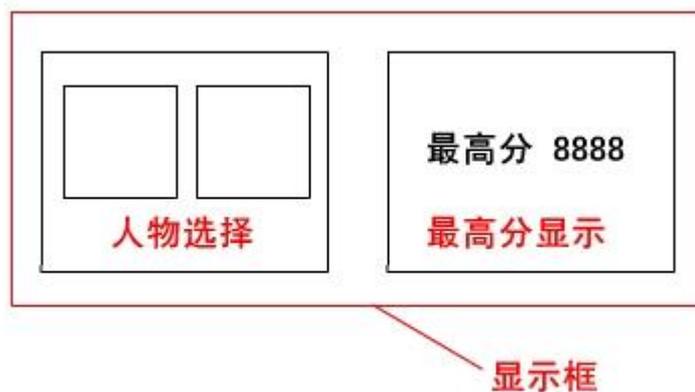
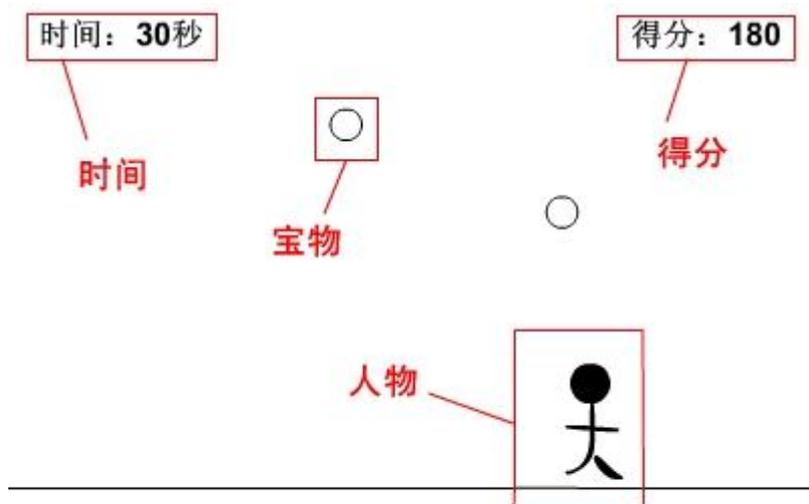
大概想了下这个小游戏的运行过程



简单示意草图2

鉴于我这个人脑子不好使，我又一样样统计了下一共有多少东西。

见下图：



● 别忘了还有背景图片和音乐……

人物

宝物

时间显示的东西

分数显示的东西

选择人物还有游戏结束后历史高分的两个提示板

背景图片和音乐音效

好了，游戏不管大小，我基本动手干活前这些基本的流程都要走。

为的是明确整个游戏的过程，然后统计下有多少东西要做，然后一样样做呗。

接下来就是搜集要用到的素材。

我找了张背景图，找了两张火影的卡通人物：鸣人和小李。



接下去就是先把要用到的动画元件做好。然后就开始考虑程序部分（后面几讲重点）。

**P.S.**看到这里，有人要骂了，小样是不是发个半吊子帖子来骗分啊？为了表明我不是骗子- -...我今天把这个小东西做了下。因为纯教学讲解用，所以只简单做了两个人物和几个道具，等大家学完后可自行扩充改写。（为了写这个帖子我今天特意做的，没功劳也有点苦劳啊~）版主给点分升级啦~很多图我都传不上~  
控制，键盘方向左右操作，上键小李长按是加速，鸣人是跳跃。

## Flash 游戏开发教程：第二节

作者：周云(cloudy1982) 来源：闪吧

请先阅读上一篇 Flash 教程：[Flash 游戏开发教程：第一节](#)

我们讨论下 AS 的问题。

先不急于动手打代码，我们先要想好这代码怎么个写法，怎么个安排。

好，现在小鸟们排好队，一个个来回答，你准备怎么写：

10个小菜鸟9个会回答：我 X 你的傻 X~！还能怎么写？F9按了开始写啊.....gotoandplay 啊，onrelease 啊，if 啊.....该写啥写啥.....

咳咳，事实上，也没错。

游戏也能这么出来。

不过我个人推荐我们该融入点面向对象编程的概念。

啥？NONONO.....我不是要开始讲复杂的东西.....

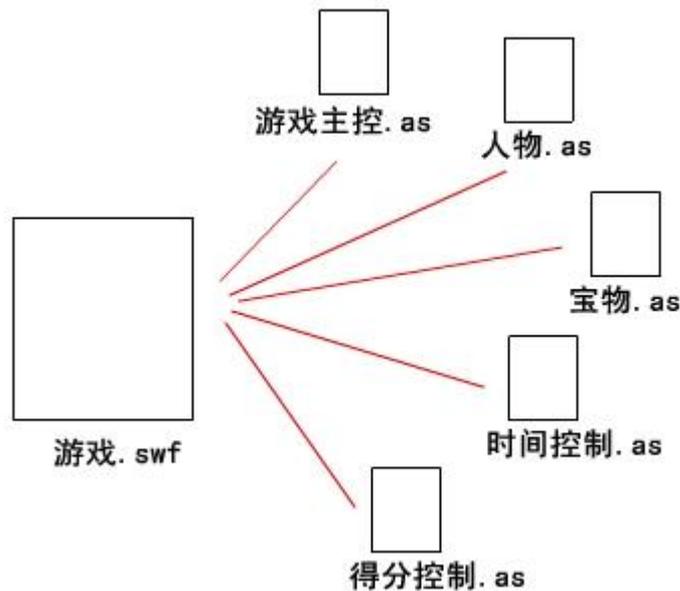
我尽量把话往简单了讲.....

我们上讲统计过游戏大致那几个东西是吧？我们现在就把那几个东西独立开来做。

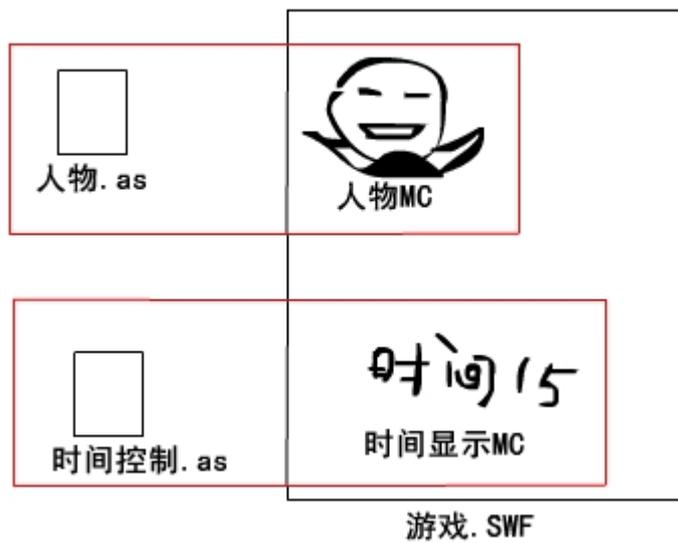
用你听过的，就是分几个 CLASS（类）来做。

比如，人物，我们就搞个人的 CLASS，宝物，我们就搞个宝物的 CLASS，和传统的 F9 按开一股脑的打代码不同，CLASS 类文件是另外写的 .AS 文件，然后游戏主的 .SWF 主文件调用的。

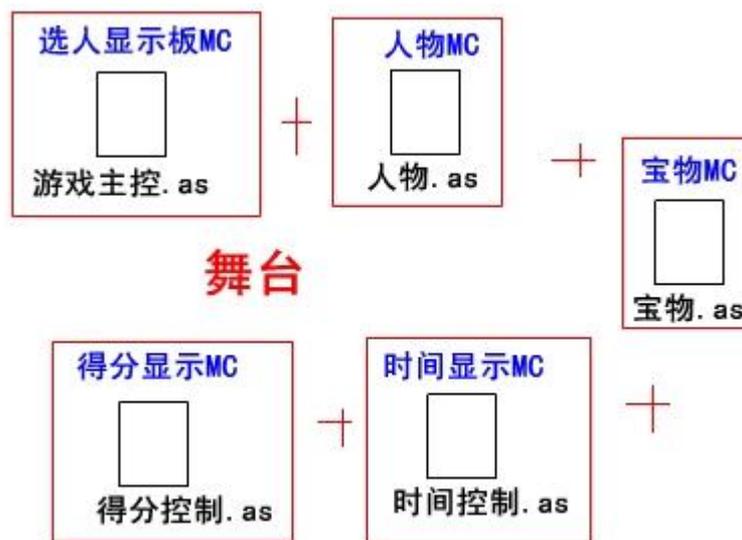
来张图：



每个 CLASS 都会被绑定在对应的 MC 上



上节统计过的，游戏一共就那么几个 MC，每人把自己对应的小蜜绑好合成一体，在舞台上一合成，就是个完整的能运作的游戏了



这样的结构有什么好处？

清晰，修改起来方便，知道哪的问题上哪个地方改

**CLASS** 封装性好，不受其它 MC 和 **CLASS** 的影响

因为**2**的原因，所以适合好几个人一起开发。比如今天你要去谈恋爱，你就拉了隔壁阿三，告诉他要做个怎么样的 **CLASS**，名称和哪几个变量以及实现的功能，然后阿三做好，你就拿来可以直接用了。

把整个游戏一个个零件拆开做，不容易犯错。

看上去比较专业，不懂行的一看，**NND**，搞的跟专业开发人员一样，拉风毙了

### Flash 游戏开发教程：第三节

作者：周云(cloudy1982) 来源：闪吧

请先阅读上一篇 Flash 教程：[Flash 游戏开发教程：第二节](#)

在具体讲解 CLASS 构造前，想花点篇幅帮小鸟们理解下 CLASS 是什么东东。

还是来举个例子，有一个机械战士的 MC，这个 MC 里有机械战士完整的构造，手脚、身体、武器……但这个机械战士仍然不能动，因为它只是一具机械，没有大脑程序告诉它该怎么走动，怎么攻击……

而我们做的“机械战士控制.as”的 CLASS，绑定在这个战士身上。

这个 CLASS 可以看作为一个封闭的黑匣子，里面装载了指导机械人行动的程序，包括怎么移动、怎么攻击、怎么做事情 A、怎么做事情 B。

而这一切都是封闭在黑匣子里的，影响不到外界程序，外接程序也影响不到它。是个独立的东西。

这个黑匣子可以装载到任何一个机械战士身上，任何一个机械战士躯体的 MC 一旦绑定上这个 CLASS，就能立即行动起来。

而且我们可以给这个 CLASS 做个外界可以调用的接口。

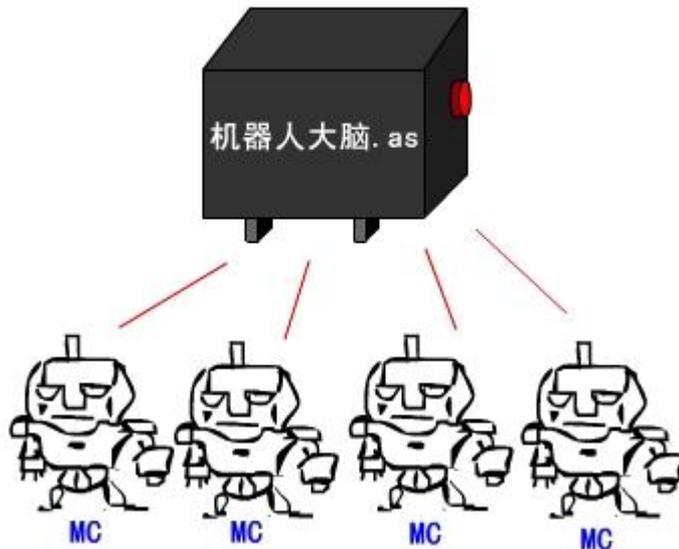
就像黑匣子上装了个可以给外界摁的按钮，你摁一次它就发一次绝招。



象上一节说的，CLASS 模式做游戏可以团队开发。我可以让别人帮我做 CLASS，做好我拿来用。我不需要知道他代码怎么写的，我只需要知道这个 CLASS 能让这个机械人按照什么规则行动，有什么按钮可以让我摁。

封装性的好处由此显示出来了。

再闯关游戏里，经常出现比如画面上主角一人对 N 个敌人的场面。这些敌人每个都可以绑定同一个 CLASS。就是说写了一个 CLASS 分别装载到 N 个实例化的 MC 中去，而不用每个敌人都各自写一段代码。



可能其中两个敌人的生命和攻击力不同。这也很简单，在 **CLASS** 里做一个接收外部传入生命和攻击力两个参数的功能，在敌人 **MC** 绑定 **CLASS** 后，再传生命和攻击力两个变量的值到 **MC** 里即可。

不过要说明，**CLASS** 并不是只能实例化在 **MC** 上，还可以装在例如 **Object** 之类的上面。

比如我们很常用的一个 **FLASH** 本来就有的 **CLASS: SOUND**

```
mySound=new Sound()
mySound.attachSound("a.mp3")
mySound.start(0,1)
```

这里我们就可以看作 **Sound** 类实例化了在 **mySound** 的上，而 **attachSound** 和 **start** 都是 **Sound** 类的“按钮”，可以让外部操控这个装载了 **Sound** 类的 **mySound**。我们并不知道 **Sound** 这个 **CLASS** 内部代码是如何的，但我们知道按钮可以用来干什么，用的也不是很爽吗？

前面说了 **CLASS** 的基本应用，可能有鸟鸟们要问了，我做 **function** 一样可以用啊，或者在 **MC** 里敲代码，跟你写 **CLASS** 一样。

那为什么还要写 **CLASS** 啊？

那下面我就来写点 **CLASS** 的另外个重要的特性-“继承” 网页教学网

先来个小例子：

隔壁阿三写了个类“机器人控制.as”，里面写了若干行的代码，包括“移动、跳跃”两个个功能。

我现在也想写个类“超级流氓机器人.as”，里面包括“移动、跳跃、攻击”三个功能。

和阿三写的类比较起来，“移动、跳跃”两个功能是一样的啦，我想写的类只多了个“攻击”功能～于是可以偷懒了。

在我写的类“超级流氓机器人.as”第一行加句简单的代码，就可以把阿三写的类完全 **COPY** 过来。（代码语法命令后面的课程会讲的）

然后我的“超级流氓机器人.as”文件里，内容就只有段写“攻击的代码”。 网页教学网看到这里，鸟鸟们又要问了，为什么不直接 **ctrl+c** 然后 **ctrl+v** 把阿三写的类复制过来，然后在里面加？别人写的代码，复杂的要死，反正我脑子转不过来，不想去分析别人的东西。分析别人写的代码，痛苦的来～

日后我再要写个“移动、跳跃、翻滚”的代码，我不是还要再 **ctrl+c** 然后 **ctrl+v**？多麻烦！

假使我用 **ctrl+c** 然后 **ctrl+v** 的方法做了，但突然发现“移动”功能里有个错误要改.....郁闷吧？你要一个个 **.as** 文件改了。而如果是继承的方法做的，你只要让阿三把他写的类修改了，你自己写的两个类就一点不需要改动了，因为“移动”部分的功能，是直接继承过来的嘛～

说到这里，顺带再提下，FLASH 里我们经常也在用类，只是你感觉不到。

做游戏用到最多，同时也是 FLASH 里最重要的类，就是~~~~~“MovieClip 类”！

鸟鸟们不知不觉的就一直在用 MovieClip 类，比如你有没有经常写：mc.onEnterFrame..  
onEnterFrame 就是 MovieClip 类里的一个方法，只要是个 MC（自动被绑上了 MovieClip 类），就都能用 onEnterFrame。 Webjx.Com

现在回到第一节说的那个天上掉宝地下人物接的游戏。

人物有两个：小李和鸣人

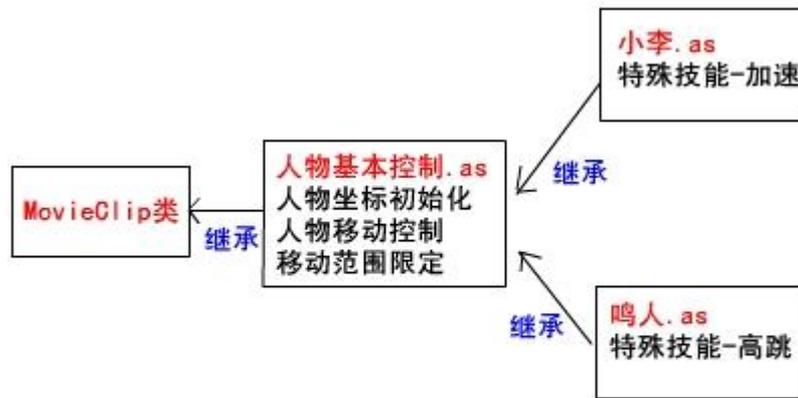
经过分析，这两个人物基本控制部分是一样的，然后就区别在两个人的各自的特殊技能不一样。

于是我们可以这么做：

先建个“人物基本控制.as”类

建个“小李.as”类，第一行一句代码继承“人物基本控制.as”类，然后在正文内容里写上小李特殊技能  
建个“鸣人.as”类，第一行一句代码继承“人物基本控制.as”类，然后在正文内容里写上鸣人特殊技能

见图：



大家可以看到，“人物基本控制.as”类也是继承了 MovieClip 类，因为要用到 MovieClip 类里的功能嘛～  
比如人物移动要用到 onEnterFrame, MC.\_x, MC.\_y 这点功能。

类的继承，还可以覆盖继承过来类里的方法。

比如 A 类有 move、jump、attack 三个方法

我要写个 B 类，也包括这三个方法，但是 attack 这个方法和 A 类的 attack 有点不同，而 move、jump 这两个方法则和 A 类里的一模一样。

于是我们写 B 类，第一行先写句话继承了 A 类。然后在正文里把 attack 方法重写一次，就自动覆盖掉了继承过来的 A 类的 attack 方法。其它的方法就不用写了，因为直接继承过来了。

让我们来换个思路做上面那个例子。

直接写了“小李.as”类，然后让“鸣人.as”类继承小李类，正文里只需要把特殊技能方法重写遍就 OK 了。

见图：



看了那么多概念，小鸟们要想，咋还不教具体怎么写 CLASS 呢？

别急啊，下节里会简单的教些的。

其实我对这种死东西教学还是比较头大的，鸟鸟们也可以自己网上搜下具体 CLASS 撰写格式、规矩和变量、方法类型。肯定比我教的要详细。因为我在 CLASS 方面也只能算只比小鸟略微胖点点的中鸟啊，哈哈

## Flash 游戏开发教程：第四节

作者：周云(cloudy1982) 来源：闪吧

请先阅读上一篇 Flash 教程：[Flash 游戏开发教程：第三节](#)

本来不想写这节的，因为这节的内容很多书上或者网上资料都有，而且写的比我要详细正规得多。但后来想想，还是写吧！

而且后面几节要结合接宝的范例游戏，具体分析它的几个 CLASS，所以，还是用一节把 CLASS 的写法说一下。可能我对 CLASS 的写法也不是很正规，有老鸟看了后，最好也能指点一下。

先来个 CLASS 基本的模式。

文件名：Time\_control.as

```

1
2 class Time_control {
3     //第一部分：声明本 CLASS 要用到的变量
4     static var a:Number=2;
5     private var b:Number;
6     private var c:Number=1;
7     public var d:Number;
8     //第二部分：构造函数
9     function Time_control() {
10        init();
11    }
12    //第三部分：隐式设置-获取法
13    public function set _d(sss:Number):Void {
14        d = sss;
  
```

```

15     }
16     public function get _d():Number {
17         return (d);
18     }
19     //第四部分：私有和公有方法
20     private function init() {
21         //一般写点初始化的内容
22     }
23     private function aa() {
24
25     }
26     public function bb() {
27
28     }
29 }

```

复制代码

如果是要继承类，比如继承 **MovieClip** 类，第一行就改写成

```
class Time_control extends MovieClip {
```

下面简单解释下各个部分（我最不擅长解释说明这类东西）

**CLASS** 类都是以

```
class 类名称 {
```

```
内容
```

```
}
```

的形式，而且类名称和文件名要相同。

**CLASS** 的内容分四部分

第一部分

声明本 **CLASS** 所用到的变量，这里要提的是，和在 **SWF** 里定义变量不同。**SWF** 里你可以用比如 **a=1** 这种自动变量形式，而在 **CLASS** 里你必须写明 **a** 的类型是 **Number**，再定义变量值是**1**

第二部分

是 **CLASS** 的构造函数，什么意思呢？就是当 **CLASS** 的实例出现后，自动执行这个 **function** 里的内容，一般都是些初始化的内容。**function** 的名称要和 **CLASS** 类的名称一样哦~而且这里初始化一般都是用个 **init()** 函数，**init** 函数具体在后面回自己定义。为什么要这样而不直接把初始化内容写在构造函数内呢？因为当继承其它类的时候，构造函数是不能继承的，还得自己再写一次，而 **init** 函数是可以继承的，于是.....呵呵，懒人懒办法嘛~直接把内容写在构造函数里的话，每次继承就还要重写次，用自定义 **init** 函数的话，就可以偷懒咯~

第三部分

是定义隐式设置-获取的 **function**，什么意思呢？上节我们说过，**class** 可以看作一个封闭的盒子，跟外界无关。但有时候我们必须读取或者改变 **CLASS** 类内的某个变量。这时候就必须借助隐式设置-获取的 **function**。只有定义后，外界才能读取和设置 **CLASS** 内的变量。定义的方法就是比普通 **function** 多了 **get/set** 这个单词。而且必须要指定返回变量类型，无需返回则用 **Void** 表示。

第四部分

就是写 **function** 咯~这个就不具体说了，**function** 大家老写了

可能大家看到，**class** 里比平时大家写的，多了 **private** 和 **public**，还有 **static** 这三个词

啥意思捏？

简单简单的说下：

**private** 与 **public** 定义的变量和方法都能在 **CLASS** 内一样使用，没啥区别

而不同的地方就是，**public** 定义的变量和方法能够被外界读取设置和使用，而 **private** 则不能，只能在封闭的 **CLASS** 内使用。

比如 **public var d:Number=1** 设置了变量 **d** 是 **public** 型的

然后在第三部分写下：

```
30
31 public function set _d(sss:Number):Void {
32     d = sss;
33 }
34 public function get _d():Number {
35     return (d);
36 }
```

复制代码

当 **CLASS** 实例化在了一个 **MC** 后

**trace(MC.\_d)** 就得到 1

输入 **MC.\_d=2** 后，**MC** 内的 **d** 就变成 2 啦～而 **private** 设定的变量，你从外部既不能读取也不能赋值。自己仔细体会下

同理，例如：

```
public function 发绝招{
}
```

就像上几节图里那个封闭盒子上的发绝招按钮，实例化到 **MC** 后，就可以直接 **MC.发绝招()**

来让 **MC** 执行 **发绝招()** 这个 **function**

而 **static** 定义的变量，怎么跟大家描述呢.....它不存在于 **CLASS** 的实例化里。

而且 **static** 定义的变量，是被所有 **CLASS** 实例共享的。

还是来个例子吧：

舞台 **ATTACH** 个 **MC**，绑定的 **ccc.as** 类。类里有个 **static** 变量 **s=1**。**ccc** 类里初始化函数 **init()** 里让 **s++**，这时候 **s** 就等于 2 了是吧？

然后继续在舞台 **ATTACH** 个 **MC**，绑定的 **ccc.as** 类，这个 **MC** 的 **s** 就直接等于 2 了。然后它在初始化函数里同样执行了 **s++**，这时候两个 **MC** 里的 **s** 都同时变成了 3

这就是所有 **CLASS** 实例共享 **static** 变量。

怎么说呢，**CLASS** 撰写方面，我自己也不是很行，可能还存在少许的理解误区或者缺少技巧经验。希望老鸟也指导下。

关于这节，我建议新鸟鸟们最好自己再找点正规教材自己学习下。

单靠这节写的些鸟文，小鸟对 **CLASS** 还不能很好的认识。正常。

所有的知识只有通过实战才能真正的理解。

在下节里，我会详细的开始教那个接宝游戏怎么做。具体的代码怎么写。**CLASS** 怎么写怎么用。手把手的教鸟们学做这个范例游戏。

## Flash 游戏开发教程：第五节

作者：周云(cloudy1982) 来源：闪吧

请先阅读上一篇 Flash 教程: [Flash 游戏开发教程: 第四节](#)

搞了半天的鸟 CLASS, 我们又回到了做游戏的内容上来。

咱们学那么多知识, 目的也为了更好的来做游戏。这个游戏吧, 前面也有朋友说到了, 根本不用 CLASS, 做个单个 FLA 文件 F9 按了猛打代码, 然后 MC 里点开了东鸟点代码, 西鸟点代码, 也能完成。没错, 确实这样。

我自己相当长一段时间内也是不懂 CLASS 之流, 只用一个 FLA 文件, 代码全死敲里面, 也做出不少洋洋得意自以为是的小游戏。

我自己本身就不是科班出身的鸟, 当年差了十分没能进计算机系, 错过了接受正规编程教育的大好光阴。

(不过事后发现世界是奇妙的, 我做过几个单位从事正职编程的都不是计算机系毕业的)

但当小游戏逐渐变成大游戏后, 一个 FLA 里, root 里, MC 里代码开始多起来, 关系开始复杂起来, 于是制作者头也开始大起来, 直接影响到睡觉睡不好, 饭也吃不好, 肠胃不通, 消化不良, 上班没精神.....

于是, 就必须正规的, 能让脑子清醒的, 能轻松分辨游戏里各类关系的开发方法。

再于是, 我每天买瓶水, 学习了 CLASS 方面的知识。(日, 学 CLASS 和买水有啥关系??)

话题扯远了, 让我们再次回到接宝上来。

一楼那个我做的简单范例 SWF 大家看过了没?

什么? 没?

赶快去下了, 试着玩玩。给你5分钟, 玩好再继续往下看。

..... (5分钟的等待)

好, 试玩过了对吧?

我现在问你, 要你分析游戏里的逻辑关系, 你怎么分析?

给你5分钟思考。

..... (5分钟的等待)

是不是: 天上不间断掉宝-当地下人物碰到宝, 就获得相应的奖励。时间到就结束游戏。

是这样的。

但我思考问题, 是按:

1. 游戏时画面上有几大元素? (MC)

2. 每个 MC 要实现什么功能?

从每个 MC 的角度来思考分析游戏的构成。

你也要习惯这样, 因为我们以后要从 MC 的角度来编写 CLASS, 每类 MC 都会有个对应的 CLASS 类来告诉这个 MC 在游戏中该做什么。

各个 MC 拼合起来, 就成了一个游戏。

简单游戏和复杂游戏的一个区别, 就是游戏里元素多少, 元素多了, 逻辑复杂, 元素少了, 逻辑简单。

好, 请鸟鸟们继续思考, 接宝游戏里有几大元素 (MC)? 它们各自要做点什么?

给你10分钟.....

..... (10分钟的等待)

见图:



公布答案：

人物

宝物

分数（右上角）

时间（左上角）

游戏总控制（包括选人、最后最高分显示）

每类各自要做的事情：

人物：

- 1.能够左右按键移动控制移动
- 2.移动范围限定（不能跑出屏幕吧？）
- 3.按上键技能
- 4.一些初始化设定（刚开始出现的坐标，移动速度）

宝物：

按随机速度下落，落出屏幕消失

碰撞检测，是否碰到人物

碰到人物后产生加分、加时间等不同的效果

初始化设定（刚开始出现的坐标，移动速度）

分数：

- 1.显示分数

时间：

30秒倒计时，显示时间变化

到了0秒，通知‘游戏控制’执行‘结束游戏’的 **function**

游戏控制：

很重要的部分，要控制协调以上几个元素之间的关系，以及游戏的过程。

集合了很多种 **function** 供调用

以下规整下包含的 function

人物选择：按照人物选择框的选择，attach 人物 MC 到舞台（人物 attach 进来后，就会自动按照他自己的 CLASS 行动啦），同时调用‘掉宝控制’的 function 开始掉宝以及‘时间控制’里倒计时方法，开始30秒倒计时。

掉宝控制：按照随机时间间隔，随机的 attach 宝物种类到舞台（一样~！宝物 attach 进来后，就会自动按照他自己的 CLASS 行动）

金币消失：这个 function 要在时间到0时被调用，作用是让屏幕上的金币原地消失

游戏结束：这个 function 是被‘时间控制’里当时间=0的时候调用的，作用为判断比较当前得分和最高分，显示框出现显示最高分。

说了那么多，不知道大家思路跟不跟的上

按我说的仔细思考下吧

要直到能理解我说的，才能继续看下一小节。

下一小节开始，我要一个个放出各个 CLASS 的具体代码了，并进行详解，请时刻关注哦！

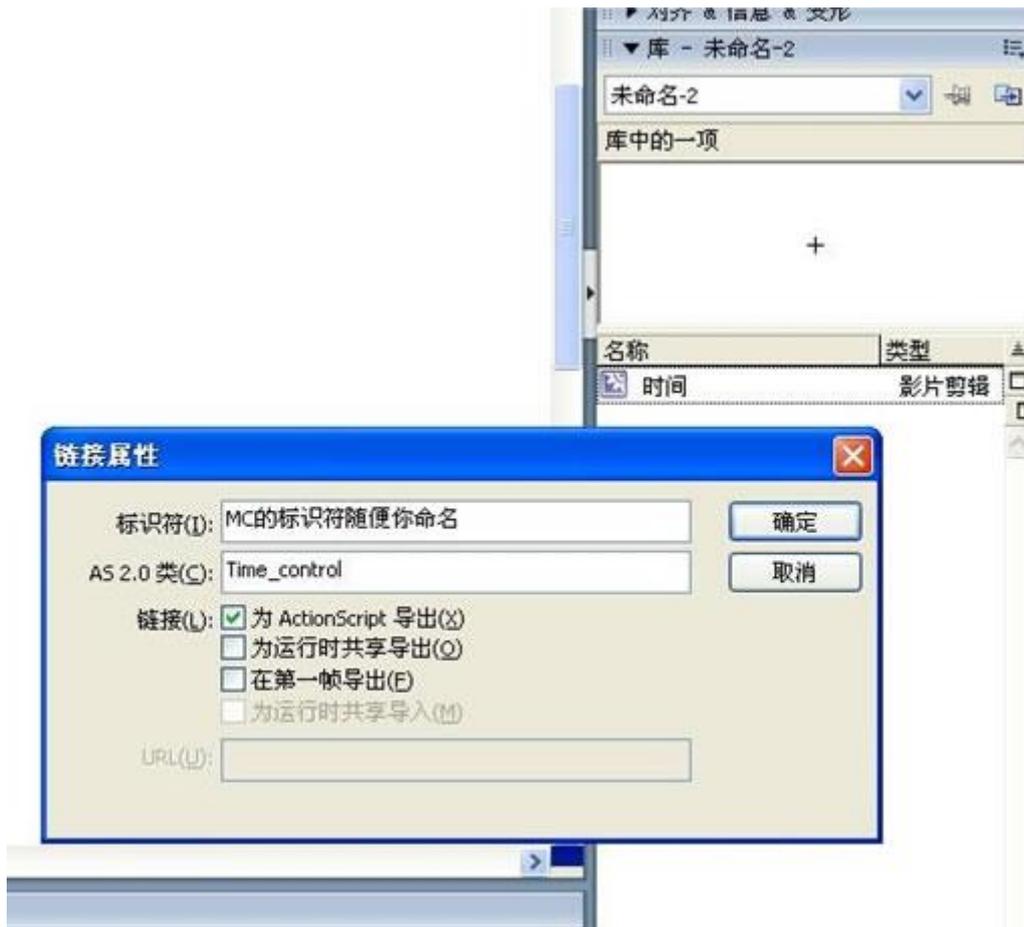
附：

编写好的 CLASS 怎么关联到 MC？

就拿上节例子 Time\_control.as 来说吧，要关联到库里的元件‘时间’

老鼠右击元件‘时间’，然后.....

见图：



如果是自己拖到舞台上，“第一帧导出”就不选

如果是 attach 上舞台，则就要选哦

## Flash 游戏开发教程：第六节

作者：周云(cloudy1982) 来源：闪吧

请先阅读上一篇文章 [Flash 教程：Flash 游戏开发教程：第五节](#)

范例接宝游戏中人物类的 CLASS 讲解

首先请下载源文件：[CLASS.rar](#)

下载好，打开一看，如下：

一个 FLA 主文件： run\_game fla

三个目录： player

items

game

我把.as 类文件分门别类的存放，这样看上去清晰，更重要的是别人看了会觉得我很专业。

目录 player 里有： Player\_1.as

Player\_2.as

目录 items 里有： Jin.as

Bao1.as

Bao2.as

目录 game 里有： Game\_control.as

Score\_view.as

Time\_control.as

.as 具体意义看文件名就应该能知道，这里要说的是，一般默认的规则，放 CLASS 目录用小写字母命名，.as 的类文件名的首字母用大写

这节详细讲下人物类的 CLASS，首先是 Player\_1.as,人物小李的类

首先去源文件里点进人物1的 MC，看下结构。

由两帧组成，一是“stand”，二是“run”

人物上面有一层有个圆形的 MC，我把它命名为 hit，靠这个 mc 来做碰撞检测。

金币里面也有这个名叫 hit 的 MC，我判断人物和金币是不是相碰是用这个语句的： 金币

MC.hit.hitTest(人物 MC.hit)

传统碰撞检测都这么做。

又开始扯话题了，马上拉回来

现在拿出 Player\_1.as 的代码

```
1 class player.Player_1 extends MovieClip {
2     private var hit:MovieClip; //人物 MC 里出现的东西都要声明，hit 这个 MC 也不例外
3     public var speed:Number; //这个是人物移动的速度
4     //构造函数
5     function Player_1() {
6         init(); //初始化的人物打包在 init()里了
7     }
8     //隐式设置-获取法
9     public function set _speed(sss:Number):Void {
10        speed = sss;
11    } //可供外界用 MC._speed=X 修改 speed 的数值
12    public function get _speed():Number {
13        return (speed);
14    } //可供外界读取 speed 的数值
```

```

15 //私-公有方法
16 //---初始化方法---
17 private function init() {
18     _y = 310;
19     _x = 300;//人物来到舞台后的初始位置
20     speed = 6;//移动速度
21     onEnterFrame = Move;//人物来到舞台后就开始 onEnterFrame，内容见 Move 这个
function
22     }
23 //---人物移动---
24 private function Move() {
25     if (Key.isDown(37)) {
26         _xscale = -100;
27         _x -= speed;
28         this.gotoAndStop("run");
29     } else if (Key.isDown(39)) {
30         _xscale = 100;
31         _x += speed;
32         this.gotoAndStop("run");
33     } else {
34         this.gotoAndStop("stand");
35     }
36     //以上是左右键的定义，应该看得懂吧？
37     if (Key.isDown(38)) {
38         speed = 12;
39     } else {
40         speed = 6;
41     }
42     //以上是上键的定义，按着后速度就加倍咯
43     range();//范围定义，内容见下面
44 }
45 //---限定移动范围---
46 private function range() {
47     if (_x>540) {
48         _x = 540;
49     }
50     if (_x<10) {
51         _x = 10;
52     }//不让人物出界
53 }
54 }

```

复制代码

这个 CLASS 文件就是这样。按照前几节我说的结构，鸟鸟们对照这看下。能不能看懂。这里要说的是，第一行的 `player.Player_1`，其中前面的 `player` 表示路径。“就相当于”，`player.Player_1`就相当于

playerPlayer\_1，关联到 MC 的时候也要这么写的。

小李的 CLASS 写完了

然后写鸣人的 CLASS

## Flash 游戏开发教程：第七节

作者：周云(cloudy1982) 来源：闪吧

请先阅读上一篇 Flash 教程：[Flash 游戏开发教程：第六节](#)

先补充个游戏里的结构：

\_root 下有三个空 MC： pl,item 和 xs

pl 这个空 MC 是用来 attach 人物的

item 这个空 MC 是用来 attach 宝物的

xs 这个 MC 是用来 attach 显示得分的

为什么要这么做呢？.....因为管理方便呀~以后你会慢慢体会到的。

严重警告：第7课未搞懂看明白者，请回上节反复研究，直至搞懂。不然继续往下看会让你体内真气逆行，任、督二脉自锁，全身犹如蚂蚁咬，饭吃不下，觉睡不着，放 P 恶臭，对 FLASH 丧失学习信心.....

(.....)

好了，能看到这行的，说明你已经是个消化第7节内容的鸟了。

这一节我也话不多说，直接放代码了。用你第7节学到的内容自我检验下。

金币10和50共用的 CLASS

```
1
2 class items.Jin extends MovieClip {
3     static var basic_speed:Number = 3; //下落基础速度
4     public var score:Number = 10;    //得分
5     public var speed:Number;        //下落速度
6     private var hit:MovieClip;      //声明碰撞检测的 MC
7     private var depth:Number;      //当宝物被人物获得，显示得分的 MC 的深度
8     //---构造函数---
9     function Jin() {
10        init();
11    }
12    //隐式设置-获取方法
13    public function set _score(kkk:Number):Void {
14        score = kkk;
15    }//得分10还是50，传入这个参数就能解决
16    public function _bspeed(){
17        basic_speed= 3;
18        trace(basic_speed)
19    }//这个要说明下，静态变量 basic_speed 每实例化一个 MC 都会自我增加，所以基础速度越来越快。这个方法是让 basic_speed 变回原始的3。用于每回合开始时初始化使用。
20    public function set _speed(sss:Number):Void {
21        speed = sss;
22    }
23    //共有-私有方法
24    private function init() {
```

```

25     _x = 50+random(450);
26     _y = -15;
27     speed = basic_speed+random(50)/10;//下落速度是由基础速度+随机增量
28     basic_speed+=0.03
29     //trace(basic_speed)
30     onEnterFrame = Move;
31 }
32 private function Move() {
33     _y += speed;
34     if (_y>380) {
35         this.removeMovieClip();
36     }
37     ///////////////////////////////////////////////////
38     if (hit.hitTest(_root.pl.player.hit)) {
39         getItem();
40     }
41     /////////////////////////////////////////////////// 碰撞检测哦~
42 }
43 public function getItem() {
44     _root.music_control.snd("jin");//声音控制，以后会讲到
45     _root.score += score;
46     depth = _root.xs.getNextHighestDepth();
47     var a:MovieClip = _root.xs.attachMovie("getScore", "getScore"+depth,
depth++);//ATTACH 显示得分的 MC
48     a.aaa.score = "+"+score;
49     a._x = _x;
50     a._y = _y;
51     this.removeMovieClip();
52 }
53 }

```

复制代码

然后是两个宝物的 **CLASS**，继承了金币的类，只不过改写了被人物吃到后的效果  
爱心

```

54
55 class items.Bao1 extends items.Jin {
56     public function getItem() {
57         _root.music_control.snd("bao");
58         depth = _root.xs.getNextHighestDepth();
59         var a:MovieClip = _root.xs.attachMovie("getScore", "getScore"+depth,
depth++);
60         a.aaa.score = "TIME";
61         a._x = _x;
62         a._y = _y;
63         ///////

```

```

64         _root.time_c._time += 8;
65         ///////调时间控制类的 public 方法，后面会讲到
66         this.removeMovieClip();
67     }
68 }

```

复制代码

五角星

```

69
70 class items.Bao2 extends items.Jin {
71     public var score:Number = 30;
72     public function getItem() {
73         //_root.score+=score
74         _root.music_control.snd("bao");
75         ///////
76         for (var i in _root.item) {
77             if (_root.item._name.substr(0, 3) != "bao") {
78                 _root.item.getItem();
79             }
80         }
81         _root.score += score;
82         depth = _root.xs.getNextHighestDepth();
83         var a:MovieClip = _root.xs.attachMovie("getScore", "getScore"+depth,
depth++);
84         a.aaa.score = "+"+score;
85         a._x = _x;
86         a._y = _y;
87         ///////
88         this.removeMovieClip();
89     }
90 }

```

复制代码

宝物这类，和人物类有点相似的。

下节会把时间控制类和分数控制类一起讲掉。大家快把这节的内容消化了吧~

## Flash 游戏开发教程：第八节

作者：周云(cloudy1982) 来源：闪吧

请先阅读上一篇 Flash 教程：[Flash 游戏开发教程：第七节](#)

范例接宝游戏中得分显示类和时间控制类的 CLASS 讲解

事实上这两部分比较简单

所以并在一起讲解了

讲解前先点开 FLA 文件，把这两个 MC 看看，对 MC 构造了解了，才能写 CLASS 嘛~

看下得分显示类的代码：

```

1
2 class game.Score_view extends MovieClip {

```

```

3     private var w1:MovieClip;
4     private var w2:MovieClip;
5     private var w3:MovieClip;
6     private var w4:MovieClip;
7     private var w5:MovieClip;
8     private var w6:MovieClip;
9     private var i:Number
10    //构造函数
11    function Score_view() {
12        init();
13    }
14    //公-私有方法
15    public function init() {
16        onEnterFrame = Control;
17    }
18    private function Control() {
19        for (i=1; i<=6; i++) {
20            if (i<=int(String(_root.score).length)) {
21                this["w"+i].gotoAndStop(int(String(_root.score).substr(i*(-1),
22            1))+1);
23            } else {
24                this["w"+i].gotoAndStop(1);
25            }
26        }
27    }

```

复制代码

原理很简单，就是让这个 MC 一直 onEnterFrame 着，按照总得分来不断操控 MC 里的6个数字 MC 变化

仔细思考下 Control()里的代码，想通了其实很简单哦~~

不多说了，懂了就跟我看时间控制类的（不懂就继续看，直到看懂为止）

下面是时间控制类：

```

28
29 class game.Time_control extends MovieClip{
30     private var t:Number
31     public var time:Number
32     //-----
33     /*
34     function Time_control(){
35         init()
36     }

```

复制代码

\*/这段被注销掉了，因为我不要它一出现在舞台就开始倒计时。什么时候倒计时，游戏控制类会给本类发出命令（调用本类中的 Go()的 function）

```

37
38 //-----
39 public function set _time(ttt:Number):Void{
40     time=ttt
41 }
42 public function get _time():Number{
43     return(time)
44 }
45 //=====
46 private function TimeGo(){
47     t++
48     if(t%30==0) time--
49     if(time==0) {
50         delete onEnterFrame
51         _root.game_c.endGame()
52     }
53 }//倒计时代码，当时间=0，通知游戏控制类作出 endGame()的行为。
54 public function Go(){
55     t=1
56     time=30
57     onEnterFrame=TimeGo
58 }//初始化，并开始倒计时
59 }

```

复制代码

有了前两节的基础，这节显的很简单.....（感觉好像应该先教这节啊.....- -.....是不是难度顺序颠倒了??）

下节会讲最后的，也是很重要的类-游戏控制类，敬请期待~！

## Flash 游戏开发教程：第九节

作者：周云(cloudy1982) 来源：闪吧

请先阅读上一篇 [Flash 教程：Flash 游戏开发教程：第八节](#)

### 范例接宝游戏中得分显示类和时间控制类的 CLASS 讲解

事实上这两部分比较简单，所以并在一起讲解了。讲解前先点开 **FLA** 文件，把这两个 **MC** 看看，对 **MC** 构造了解了，才能写 **CLASS** 嘛~看下得分显示类的代码：

```

1
2 class game.Score_view extends MovieClip {
3     private var w1:MovieClip;
4     private var w2:MovieClip;
5     private var w3:MovieClip;
6     private var w4:MovieClip;
7     private var w5:MovieClip;
8     private var w6:MovieClip;
9     private var i:Number
10    //构造函数

```

```

11  function Score_view() {
12      init();
13  }
14  //公-私有方法
15  public function init() {
16      onEnterFrame = Control;
17  }
18  private function Control() {
19      for (i=1; i<=6; i++) {
20          if (i<=int(String(_root.score).length)) {
21              this["w"+i].gotoAndStop(int(String(_root.score).substr(i*(-1),
22  1))+1);
23          } else {
24              this["w"+i].gotoAndStop(1);
25          }
26      }
27  }

```

复制代码

原理很简单，就是让这个 MC 一直 **onEnterFrame** 着，按照总得分来不断操控 MC 里的6个数字 MC 变化。仔细思考下 **Control()**里的代码，想通了其实很简单哦~~不多说了，懂了就跟我看时间控制类的（不懂就继续看，直到看懂为止）下面是时间控制类：

```

28
29  class game.Time_control extends MovieClip{
30      private var t:Number
31      public var time:Number
32      //-----
33      /*
34      function Time_control(){
35          init()
36      }
37      */这段被注销掉了，因为我不要它一出现在舞台就开始倒计时。什么时候倒计时，游戏控制类
    会给本类发出命令（调用本类中的 Go()的 function）
38      //-----
39      public function set _time(ttt:Number):Void{
40          time=ttt
41      }
42      public function get _time():Number{
43          return(time)
44      }
45      //=====
46      private function TimeGo(){
47          t++
48          if(t%30==0) time--

```

```

49         if(time==0) {
50             delete onEnterFrame
51             _root.game_c.endGame()
52         }
53     }//倒计时代码，当时间=0，通知游戏控制类作出 endGame()的行为。
54     public function Go(){
55         t=1
56         time=30
57         onEnterFrame=TimeGo
58     }//初始化，并开始倒计时
59 }

```

复制代码

有了前两节的基础，这节显的很简单.....（感觉好像应该先教这节啊.....- -.....是不是难度顺序颠倒了??）下节会讲最后的，也是很重要的类-游戏控制类，敬请期待~！

## Flash 游戏开发教程：第十节

作者：周云(cloudy1982) 来源：闪吧

请先阅读上一篇 Flash 教程：[Flash 游戏开发教程：第九节](#)

### 范例接宝游戏中游戏控制类的 CLASS 讲解

终于讲到这节了，累死，也饿死，为了明年年初要拍婚纱，我每天中午就吃个苹果减肥，然后靠苹果的能量再给鸟鸟们写教程.....今天算是暴走式的更新速度了，从第7节写到了第10节，胃软导致了手软，手软导致了脚软.....版主看到，体恤下吧，加点分，我好早日升中级会员.....（咳咳）

本节讲的是最后一个类：游戏控制类

我自己每个游戏都会在舞台上有个 MC 帮我看场子，指挥调度手下的兄弟，提供各类凶器，把握整个进程.....这个游戏的游戏控制类：**Game\_control.as**

先点开 **FLA** 里的 **MC**，看看它的构造。有两个框，一个是选择人物的，一个是显示最高得分的。我用它们两的 **visible** 属性控制它们可见和不可见。**Game\_control.as** 把握着整个游戏的进程，提供了一些方法。

先看它怎么控制游戏的：

- 1.先初始化游戏参数，然后让选人框可见，让玩家选人。
- 2.玩家选好人，它就把人物 **attach** 到舞台上，同时通知‘时间控制类’这个小弟你可以**30**秒倒计时了，又同时开始在天上撒金币和宝物。
- 3.‘时间控制类’小弟一看时间到**0**了，马上又来通知大哥，时间到**0**了
- 4.大哥收到风声后，停止撒金币，让画面上已有的金币宝物全部消失，同时一番比较后显示出你的历史最高分，同时提供你个重新游戏的按钮
- 5.按重新游戏，大哥就回到第**1**步.....（好像所有小弟里，只有分数显示自管自工作，没有被大哥叫到）它包含了一些方法，供小弟和自己用：

- 1.初始化数据：**init()**
- 2.根据玩家选择，在舞台上 **attach** 人物：**chose(n:Number)**
- 3.天上播撒金币和宝物：**onGame()**
- 4.屏幕上宝物消失：**jin\_disappear()**
- 5.结束游戏，比较得分和最高分，让显示框显示最高分：**endGame()**

对照着我们来看下代码：

```

1
2 class game.Game_control extends MovieClip {
3     private var t:Number = 1;
4     private var game_begin:MovieClip
5     private var game_end:MovieClip
6     private var depth:Number
7     private var topScore:Number=0
8     public var time:Number;
9     //构造函数
10    function Game_control() {
11        init();
12    }
13    //隐式设置方法
14    public function set _time(ttt:Number):Void {
15    }
16    //公-私有方法
17    public function init() {
18        time=35
19        _root.score = 0;
20        _root.createEmptyMovieClip("pl", 3);
21        _root.createEmptyMovieClip("item", 2);
22        _root.createEmptyMovieClip("xs", 1);
23        //初始化里建立空 MC，同时也就是把以前的 MC 删除覆盖
24        game_begin._visible=true
25        game_end._visible=false
26        _root.music_control.bg(1)
27    }
28    public function chose(n:Number) {
29        game_begin._visible=false
30        _root.pl.attachMovie("ren"+n, "player", 1);
31        _root.time_c.Go()//时间控制小弟~号开始倒计时啦~
32        _root.music_control.bg(2)
33        onEnterFrame = onGame;
34    }
35    private function onGame() {
36        t++;
37        if (t%time == 0) {
38            depth=_root.item.getNextHighestDepth()
39            if (random(8) == 0) {
40                var a:MovieClip=_root.item.attachMovie("bao"+(1+random(2)),
"bao"+depth, depth++);
41            }
42            else if (random(5) == 0) {
43                var a:MovieClip=_root.item.attachMovie("jin2", "jin"+depth,

```

```

depth++);
44         a.score=50
45     } else {
46         _root.item.attachMovie("jin1", "jin"+depth, depth++);
47     }
48     time = 20+random(30);
49     }//这段自己仔细看下，就明白怎么随机时间间隔里播撒出随即类型的宝物了
50 }
51 private function jin_disappear() {
52     for(var i:String in _root.item){
53         _root.item._speed=0
54         _root.item._speed()//这个方法 Jin.as 里写过的，还记得吗？哈哈
55         _root.item._alpha-=5
56         _root.item._xscale+=2
57         _root.item._yscale+=2
58         if(_root.item.alpha<0) {
59             _root.item.removeMovieClip()
60             delete onEnterFrame
61         }
62     }//大小和 alpha 同时变化直到宝物消失~
63 }
64 public function endGame(){
65     onEnterFrame=jin_disappear
66     game_end._visible=true
67     if(_root.score>topScore) {
68         topScore=_root.score
69         game_end.sc=topScore
70     }//判断和记录历时最高分，同时显示最高分框显示~
71 }
72 }

```

复制代码

事实上在人物选择框，两个人物是按钮，用来调用‘游戏控制’类里的 **chose(n:Number)**方法。当 **chose(n:Number)**被调用时，后面一连串的进程就都自动展开了……今天一下子讲的太多了，能消化掉吗？游戏主体也就这样了。剩下点小东西，比如声音控制，下节课讲。

## Flash 游戏开发教程：第十一节

作者：周云(cloudy1982) 来源：闪吧

请先阅读上一篇 Flash 教程：[Flash 游戏开发教程：第十节](#)

### 声音控制&我对 FLASH 创作游戏的看法

声音控制，我个人不习惯用类来控制。

我一直是在游戏第一帧先用 **Sound** 类把库里的声音关联起来，然后做个 **Object**，在里面定义 **function** 来控制声音的。当然你也可以写个类来控制，这随你的，用痰盂还是马桶，我们谁也不能强求谁，对吧？把第一帧的声音控制代码拿出来：

```

1
2  //////////---声音控制---////////////////

```

```

3  music_bg1 = new Sound();
4  music_bg1.attachSound("bg1");
5  music_bg2 = new Sound();
6  music_bg2.attachSound("bg2");
7  music_bg_jin = new Sound();
8  music_bg_jin.attachSound("bg_jin");
9  music_bg_bao = new Sound();
10 music_bg_bao.attachSound("bg_bao");
11 //库里的声音文件都关联起来~
12 music_control = new Object();
13 music_control.bg = function(n) {
14     for (i=1; i<=2; i++) {
15         _root["music_bg"+i].stop();
16     }
17     _root["music_bg"+n].start(0, 999);
18 };
19 music_control.snd = function(n) {
20     _root["music_bg_"+n].start(0, 1);
21 };
22 //music_control 这个 Ob 里定义了 bg()和 snd 两个方法, bg 用来放长段背景音乐, snd 用来放
    音效
23 _root.music_control.bg(1)
24 //看, 就是这么用的

```

复制代码

现在前面几节那些 CLASS 里写到的声音调用语句知道是怎么回事了吧?

-----分割线-----

游戏开发基础的课程, 至此已经全部写完了。

范例游戏有很多地方可以扩充, 比如加点掉宝种类, 再加几个人物和技能, 玩到一半掉下只猪.....等等等等, 我只是抛砖引玉下而已~要玩我做的游戏? 等我找找.....随便先找个

[http://www.mhhf.net/games\\_786.html](http://www.mhhf.net/games_786.html), 如果喜欢, 我可以再开贴教这类 STG 射击型的, 或者动作闯关型的, 或者 FMS 联机型的, 或者 XX 型的, 或者 YY 型的都可以.....

关于做游戏, 事实上正规的团队应该有三种人:

- 1.策划
- 2.美工
- 3.程序员

但实际上, FLASH 这种东西, 相比其它, 真的属于不赚钱的。所以搞的专业人士不多。大多数情况下, 都是闪客单干。所以很多时候开发都不正规, 比如代码不注释~在一个 FLA 里, 帧里, 隐蔽的很多 MC 的小边小角里, 敲了很多代码, 如果有人接手, 必然会分析到晕倒.....这种不正规, 在长时间单干下, 是不会被意识到并且改正的。但如果参加过团队制作, 就会立即意识到这个问题。一个好程序员, 或者懂的很多的牛人, 并不会必定是个好的游戏开发者。

我自己见过好几个自己认为技术很牛的人, 他们并不做游戏, 或者游戏做出来并不受欢迎, 他们自己分析原因是没好美工, 导致画面粗糙, 不吸引人。而我则认为更主要的原因是缺少好的游戏策划。在我心里, 游戏

受欢迎程度取决的因素：策划>程序>美工。就比如细节方面，好的策划知道怎么样的设定更体贴玩家，知道什么元素能抓牢玩家，什么能激起玩家欲望，什么能让玩家 **HIGH**。老实说，这比懂 **CLASS** 重要得多了。

如果你跟我一样是一名独立创作 **FLASH** 游戏的制作者，那你先要成为一名很好的游戏玩家。要能从玩家的角度来设计制作游戏。当然，策划游戏，是需要天赋的。我们不能要求每个人样样都完美，毕竟每个人都有自己擅长的和不擅长的。继续努力吧。

**【本教程结束】**