

# tutorial

Por Alexandre Cavalcanti

Adobe Certified Expert e Adobe Certified Instructor em Flash CS3. Consultor em design e desenvolvimento web, pós-graduado em design digital. Palestrante e professor dos cursos intensivos, de graduação e pós-graduação do Instituto Infnet desde 2000. Visite seu blog em [zarabatana.com/blog](http://zarabatana.com/blog).

## Actionscript 3.0 para webdesigners - Parte 1/4

Salve, salve, meu caro leitor! Durante as próximas edições, mostraremos como utilizar, de maneira simples e prática, a linguagem de programação Actionscript 3.0 nos seus projetos em Adobe Flash.

O objetivo deste tutorial é mostrar que programação não é um bicho de sete cabeças e que pode ajudar o webdesigner a automatizar rotinas repetitivas, se comunicar melhor com desenvolvedores e gerenciar projetos mais facilmente, evitando a perda de tempo com tarefas que o computador pode executar facilmente e deixando o foco no que realmente interessa. Criar.

Nesta primeira parte, veremos os conceitos básicos do Actionscript 3.0 através de exemplos simples e fáceis de seguir, utilizando o Adobe Flash CS4 Professional.

### Elementos básicos da linguagem

Antes de começarmos, faça o download dos arquivos desta parte do tutorial em: <http://www.revistawebdesign.com.br/index.php/downloads>. Para iniciar, abriremos o Adobe Flash CS4 Professional e definiremos nossa área de trabalho (*workspace*) como **Essentials**. Na versão CS4 há diversos layouts da área de trabalho listados na barra superior do software. Após o tutorial, escolha o que mais lhe agrada.

### Variáveis

Uma variável é um espaço na memória onde armazenamos um dado. Por exemplo, ao utilizarmos a expressão **x = 10**, guardamos na memória o número **10** e o identificamos como **x**. Toda vez que quisermos utilizar esse dado, basta chamarmos pelo nome x. Muito bem, vamos começar:

- 1 - Selecione **File** → **Open**, escolha o arquivo **AS3\_parte1\_ex1\_inicio fla** e clique **Open**.
- 2 - Selecione **File** → **Save As...** e salve o arquivo como **AS3\_parte1\_ex1 fla**.
- 3 - Selecione o quadro **1** da camada **AS** e clique no menu **Window** → **Actions**, ou pressione **F9**. Aparecerá o painel **Actions**, que é o local onde você escreverá o código Actionscript.
- 4 - Declare uma variável **angulo**, que irá armazenar um número inteiro. Esse número servirá, por exemplo, para a definição do ângulo de rotação de um MovieClip. Em seguida, atribua o valor **45** para a variável **angulo**. Para isso, digite o seguinte código:

```
var angulo:int;
angulo = 45;
```

**Observação:** os identificadores (nomes) das variáveis **não** podem conter espaços ou caracteres especiais, exceto  (sublinhado) e \$ (cifrão), nem começar por número e devem ser únicos no escopo onde são declarados.

- 5 - Feche o painel **Actions** e Selecione **File** → **Save**.

### Propriedades

Uma propriedade é uma característica de um objeto (nome, altura, largura, visibilidade, rotação, opacidade etc.) e com o Actionscript podemos alterá-las em tempo de execução. Neste exemplo, modificaremos a propriedade **rotation** (rotação) de um MovieClip que está no palco. Vejamos:

- 1 - Continuando no arquivo **AS3\_parte1\_ex1 fla**, selecione o quadrado que está no palco.
- 2 - No painel **Properties**, no campo Nome de Instância (*Instance Name*), digite: **quadrado\_mc**.
- 3 - Selecione o quadro **1** da camada **AS** e clique no menu **Window** → **Actions**, ou pressione **F9**. No painel **Actions**, especifique que o **quadrado\_mc** terá a sua propriedade **rotation** definida com o valor da variável **angulo**. Para isso, digite a seguinte linha abaixo do código já existente:

```
quadrado_mc.rotation = angulo;
```

- 4 - Selecione **File** → **Save**.
- 5 - Teste o filme, selecionando **Control** → **Test Movie**, e veja o resultado. Você modificou a rotação de um objeto em tempo de execução e com pouquíssimas linhas de código.
- 6 - Vamos utilizar mais uma propriedade. Declare a variável **opacidade** como número real e atribua o valor **.5** (meio) a ela. Em seguida, defina **opacidade** como valor da propriedade **alpha**, o que irá tornar o **quadrado\_mc** transparente em 50%. Veja como fica o código ao final do exemplo:

```
var angulo:int, opacidade:Number;
angulo = 45;
opacidade = .5;
quadrado_mc.rotation = angulo;
```

```
quadrado_mc.alpha = opacidade;
```

7 - Selecione **File** → **Save**.

8 - Teste o filme, selecionando **Control** → **Test Movie**, e note a transparência do objeto.

**Observação:** caso não tenha dado certo, abra o arquivo **AS3\_parte1\_ex1\_final fla** e compare com o exemplo que você fez.

## Funções

Uma função é um bloco de código que é executado somente quando chamado e que pode retornar um valor. Utilizando funções, você pode reutilizar um código Actionscript em diversas situações e de maneira configurável, através de atributos. Vejamos um exemplo:

1 - Selecione **File** → **Open**, escolha o arquivo **AS3\_parte1\_ex2\_inicio fla** e clique **Open**.

2 - Selecione **File** → **Save As...** e salve o arquivo como **AS3\_parte1\_ex2 fla**.

Obs.: nós agora temos dois objetos na tela (**quadrado1\_mc** à esquerda e **quadrado2\_mc** à direita) e iremos alterar a rotação de cada um deles de maneira independente. Para isso, vamos criar uma função chamada **girarObjeto**.

**Dica:** uma função sempre terá um caráter de ação e deve ser identificada utilizando-se um verbo seguido de um substantivo (ex.: girarObjeto). Valem as regras de identificação das variáveis.

3 - Selecione o quadro **1** da camada **AS** e clique no menu **Window** → **Actions**, ou pressione **F9**. No painel Actions, envolva o código que altera a propriedade **rotation** do **quadrado1\_mc** em um bloco **function** chamado **girarObjeto**, delimitado por chaves, e que não retornará valor (**void**). Logo após, faça a chamada do bloco escrevendo o nome da função, **girarObjeto**, seguido de parênteses, como mostrado no código abaixo:

```
var angulo:int;
angulo = 45;
function girarObjeto():void {
quadrado1_mc.rotation = angulo;
}
```

```
girarObjeto();
```

4 - Selecione **File** → **Save**.

5 - Teste o filme, selecionando **Control** → **Test Movie**, e veja o resultado.

**Observação:** o objeto **quadrado1\_mc** girou corretamente e agora iremos executar essa mesma ação no **quadrado2\_mc**, mas utilizando um ângulo diferente. Como iremos girar dois objetos, faremos duas chamadas. Logo, nós teremos que dizer, a cada chamada da função, qual objeto será girado e de quantos graus será esse giro. Esses dados serão passados para a função através de atributos, que no caso se chamarão: **objeto** e **angulo**. Os nomes dos atributos seguem o padrão das variáveis.

6 - Feche a janela do Flash Player e retorne ao painel **Actions**. Remova as linhas com a declaração e a atribuição da variável **angulo**. O ângulo agora será definido através do atributo **angulo** da função.

7 - Insira nos parênteses da função os dois atributos com seus tipos de dado. Isto é, o atributo **objeto** será do tipo **MovieClip**, e o atributo **angulo** será do tipo **int** (número inteiro).

8 - No bloco de código, substitua a referência ao objeto **quadrado1\_mc** pelo atributo **objeto**. Dessa forma, poderemos dizer qual objeto será girado na chamada da função. Observe também que o valor do ângulo agora será determinado na chamada da função, através do atributo **angulo**.

9 - Abaixo do bloco da função, escreva duas chamadas. Na primeira, o **quadrado1\_mc** irá girar **45** graus para a direita. Enquanto que, na segunda chamada, o **quadrado2\_mc** irá girar **15** graus para a esquerda. Veja como deverá ficar o seu código ao final do exemplo:

```
function girarObjeto(objeto:MovieClip,
angulo:int):void {
    objeto.rotation = angulo;
}
```

```
girarObjeto(quadrado1_mc, 45);
girarObjeto(quadrado2_mc, -15);
```

10 - Selecione **File** → **Save**.

11 - Teste o filme, selecionando **Control** → **Test Movie**. Note que os objetos compartilham a mesma funcionalidade, mas de formas diferentes, devido à utilização de atributos.

**Observação:** caso não tenha dado certo, abra o arquivo **AS3\_parte1\_ex2\_final fla** e compare com o exemplo que você fez.

Tá vendo? Programar não é tão complicado quanto parece. Espero que esses exemplos tenham servido para ilustrar alguns fundamentos do Actionscript. Na próxima edição, faremos com que essas funções respondam às interações com o usuário, entre outras coisas. A gente se vê lá. Até!