

Web Games

Prof. Luciano Alves

Estruturas condicionais

Estruturas condicionais

Na programação, um algoritmo é um conjunto de instruções que são executados sequencialmente. Vejamos um exemplo abaixo:

```
var soma: int;  
var numero1 : int;  
var numero2 : int;
```

```
numero1 = 10;  
numero2 = 3;  
soma = numero1 + numero2;
```

```
trace("Soma = " + soma);
```

Estruturas condicionais

Se observamos o exemplo anterior, temos uma execução sequencial das instruções (uma linha após a outra). Porém, na programação, existem situações em que haverá a necessidade de que um determinado conjunto de instruções seja executado sobre uma determinada condição (podendo ocorrer o que chamamos de desvio condicional). Para essa situação, utilizamos uma **estrutura condicional**.

Vamos conhecer agora as estruturas condicionais da linguagem Action Script 3.0.

Estruturas condicionais

A estrutura condicional if

A estrutura condicional **if** executa um bloco de comandos se uma determinada condição for satisfeita. Veja a sintaxe abaixo:

Sintaxe :

If (condicao)
<única instrução >

ou

If (condicao) {
<instrução 1>
<instrução 2>
:
<instrução n>
}

Estruturas condicionais

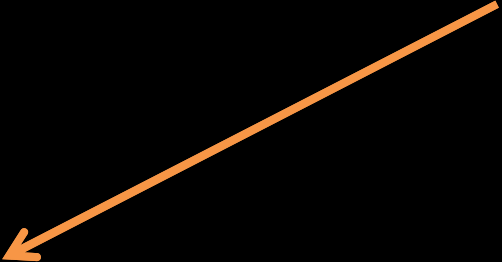
A estrutura condicional if

Exemplo:

```
var numero: int;
```

```
numero = 13;
```

```
if(numero >= 12)  
    trace("O numero é maior que 12");
```



Aqui a condição será VERDADEIRA pois a variável número assume o valor 13 e esse valor é maior que 12, logo, ele irá executar a instrução seguinte mostrando na tela a mensagem : O número é maior que 12.

Estruturas condicionais

A estrutura condicional if

Exemplo:

```
var numero: int;
```

```
numero = 5;
```

```
if(numero > 12)
```

```
trace("O numero é maior que 12");
```

Aqui a condição será FALSA pois a variável número assume o valor 5 e esse valor NÃO É MAIOR que 12, logo, ele simplesmente vai IGNORAR a instrução abaixo riscada em vermelho e vai continuar a executar as próximas instruções.

Estruturas condicionais

A estrutura condicional if/else

A estrutura condicional **if/else** executa um bloco de comandos se uma determinada for **verdadeira** e um outro bloco de comandos caso a condição seja **falsa**. Veja a sintaxe abaixo:

Sintaxe :

If (condicao)
<única instrução >
else
<única instrução>

ou

If (condicao) {
<instrução 1>
:
<instrução n>
} else {
<instrução 1>
:
<instrução n>
}

Estruturas condicionais

A estrutura condicional if/else

Exemplo:

```
var numero: int;
```

```
numero = 13;
```

```
if(numero > 12)
```

```
    trace("O numero é maior que 12");
```

```
else
```

```
    trace("O numero é menor ou igual que 12");
```

Aqui a condição será VERDADEIRA pois a variável número assume o valor 13 e esse valor é maior que 12, logo, ele irá executar a instrução indicada pela SETA VERDE e vai IGNORAR a instrução RISCADA COM A LINHA VERMELHA.

Estruturas condicionais

A estrutura condicional if/else

Exemplo:

```
var numero: int;
```

```
numero = 6;
```

```
if(numero > 12)
```

```
trace("O numero é maior que 12");
```

```
else
```

```
trace("O numero é menor ou igual que 12");
```

Aqui a condição será VERDADEIRA pois a variável número assume o valor 6 e esse valor NÃO É MAIOR que 12, logo, ele irá executar a instrução indicada pela SETA VERDE e vai IGNORAR a instrução RISCADA COM A LINHA VERMELHA.

Estruturas condicionais

Operadores relacionais

Conheça agora os operadores relacionais que podemos utilizar dentro de uma condição.

Operador relacional	Símbolo
Maior ou igual	>=
Maior	>
Menor ou igual	<=
Menor	<
Igual	==
Diferente	!=

Estruturas condicionais

Operadores relacionais (Exemplos)

```
if(numero >= 20)
```

```
:
```

```
if(numero_magico == numero_usuario)
```

```
:
```

```
if(soma != (valor1 + valor2))
```

```
:
```

Estruturas condicionais

Operadores lógicos

Até agora dentro das estruturas condicionais tínhamos somente avaliado uma única condição. E se eu agora quiser avaliar mais de uma condição dentro do *if* ? Pra isso usamos os **operadores lógicos**. Vamos conhecer agora os operadores lógicos da linguagem ActionScript.

Estruturas condicionais

O operador lógico And (&&)

O operador lógico **And** (ou **E** no bom português), representado pelo símbolo **&&**, retorna verdadeiro (true) se todas as condições avaliadas por ele forem verdadeiras. **Veja a tabela abaixo:**

Condição 1	Condição 2	Operador E
Verdadeiro	Verdadeiro	Verdadeiro
Verdadeiro	Falso	Falso
Falso	Verdadeiro	Falso
Falso	Falso	Falso

Estruturas condicionais

O operador lógico And (&&) : Exemplo

Veja um exemplo abaixo demonstrando o uso do operador lógico And (&&).

```
If ( (numero >= 10) && (numero <= 90) )
```

```
If ( (poder_atacar) && (sangue_inimigo > 50) )
```

Estruturas condicionais

O operador lógico Or(||)

O operador lógico Or (conhecimento OU no bom português), representado pelo simbolo ||, retorna verdadeiro (true) se pelo menos uma das condições avaliadas por ele for verdadeira. Veja a tabela abaixo:

Condição 1	Condição 2	Operador OU
Verdadeiro	Verdadeiro	Verdadeiro
Verdadeiro	Falso	Verdadeiro
Falso	Verdadeiro	Verdadeiro
Falso	Falso	Falso

Estruturas condicionais

O operador lógico Or (||) : Exemplo

Veja um exemplo abaixo demonstrando o uso do operador lógico Or (||).

```
if ( (opcao== "sim") || (opcao == "nao") )  
    :
```


Estruturas condicionais

O operador lógico Not(!)

O operador lógico **Not** (conhecimento como **NÃO** no bom português), representado pelo simbolo **!** , INVERTE o valor da condição retornando verdadeiro (true) se a condição for falsa (false) e retorna falso se a condição for verdadeira. Veja a tabela abaixo:

Condição	Operador NOT
Verdadeiro	Falso
Falso	Verdadeiro

Estruturas condicionais

O operador lógico Not (!) : Exemplo

Veja um exemplo abaixo demonstrando o uso do operador lógico Not (!).

```
If ( ! (numero > 12) )  
    :
```

A instrução acima equivale também a :

```
If (numero <= 12)
```