

# 1 Estruturas de control

As estruturas de control permiten modificar o fluxo de execución básico do *script*. Tamén nos permiten que certas partes do *script* non teñan que executarse sempre.

## 1.1 Sumario

- 1 Sentencias Condicionais
  - ◆ 1.1 IF
  - ◆ 1.2 IF...ELSE
- 2 Bucles
  - ◆ 2.1 FOR
  - ◆ 2.2 FOR...IN

## 1.2 Sentencias Condicionais

Con estas sentencias seremos capaces de tomar unha decisión baseándonos nunha expresión e facer que o programa elixa entre dous camiños de execución diferentes.

### 1.2.1 IF

A sintaxe da sentencia *if* é a seguinte:

```
if (condición) {  
    [sentencias]  
}
```

A condición debe ser unha expresión lóxica:

- ◇ Se o resultado de avaliación é certa, a sentencia executarase, e continuarase coa execución do programa.
- ◇ Se o resultado é falso, *JavaScript* ignora as sentencias e continuarase coa execución do *script*.

- **Exemplo con *if*:** Empregar a sentencia *if* para saber cal é o maior de dous números.

```
<html>  
<head>  
    <title>Estruturas de control</title>  
</head>  
<body>  
    <center>  
        <script language="javascript">  
  
            </script>  
        </center>  
    </body>  
</html>
```

### 1.2.2 IF...ELSE

As veces, a utilización da sentencia *if* en solitario non é suficiente, porque se precisa executar certas sentencias cando a expresión condicional non se cumpre. Para poder facer isto, engadimos un bloque *else* ó bloque *if*, a sintaxe quedará deste xeito:

```
if (condición) {  
    [sentencias]  
} else {  
    [sentencias]  
}
```

- **Exemplo *if...else*:** Empregar a sentencia *if...else* para saber cal é o maior de dous números.

```
<html>  
<head>  
    <title>Estruturas de control</title>
```

```

</head>
<body>
  <center>
    <script language="javascript">

    </script>
  </center>
</body>
</html>

```

## 1.3 Bucles

A utilización de bucles dentro dun *script* serve para moitos propósitos: para contar, para percorrer estruturas de tipo *array*,...

### 1.3.1 FOR

O bucle *for* ten a seguinte sintaxe:

```

for ([exp_inicialización]; [exp_condición]; [exp_bucle]) {
  [sentencias]
}

```

As tres expresión cerradas entre corchetes existentes no paréntese, son opcionais, pero é preciso escribir os caracteres punto e coma (;) aínda que estas se omitan.

- ◊ **exp\_inicialización:** Utilízase para inicializar e declarar a variable de contador do bucle.
- ◊ **exp\_condición:** Define a condición que debe cumprirse para que continúe contando.
- ◊ **exp\_bucle:** Incrementa ou decrementa a variable empregada como contador do bucle.

- **Exemplo for 1:** Facer un *script* que imprima os números pares comprendidos entre 1 e 100.

```

<html>
<head>
  <title>Estruturas de control</title>
</head>
<body>
  <center>
    <script language="javascript">

    </script>
  </center>
</body>
</html>

```

- **Exemplo for 2:** Mellorar o programa anterior para que poña os do 1 ó 9 nunha liña, os do 10 ó 19 na seguinte liña, os do 20 ó 29 noutra,...

Para facelo empregar o **operador módulo (%)**, con este operador, tal e como sabemos, descubrimos se un número é múltiplo de outro observando se o módulo de ambos números é 0.

```

<html>
<head>
  <title>Estruturas de control</title>
</head>
<body>
  <center>
    <script language="javascript">

    </script>
  </center>
</body>
</html>

```

As sentencias for tamén se poden aniñar, mira o seguinte exemplo que o demostra:

- **Exemplo for 3:** Fai un programa que mostre o valor que se obtén ó sumarlle a un número todos os valores comprendidos entre el mesmo e o 0 (por exemplo do número 5: 5+4+3+2+1=15, debería mostrar **5 : 15**).

```
<html>
<head>
  <title>Estructuras de control</title>
</head>
<body>
  <center>
    <script language="javascript">

    </script>
  </center>
</body>
</html>
```

- **Exemplo for 4:** Reforma o exemplo anterior para que faga ese sumatorio para todos os números comprendidos entre o 0 e ese número (por exemplo, se pos un 5 que faga o sumatorio dende o número 0 ata o número 5).

```
<html>
<head>
  <title>Estructuras de control</title>
</head>
<body>
  <center>
    <script language="javascript">

    </script>
  </center>
</body>
</html>
```

- **Exemplo for 5:** Facer un script para que mostre en pantalla a táboa de multiplicar dun número dado como dato, debe de mostralo en formato táboa.

```
<html>
<head>
  <title>Estructuras de control</title>
</head>
<body>
  <center>
    <script language="javascript">

    </script>
  </center>
</body>
</html>
```

- **Exemplo FOR\_6:** Modifica o script anterior para que pida o número do que queres facer a táboa de multiplicar (emprega a función de JavaScript `prompt()`).

### 1.3.2 FOR...IN

Este bucle utilízase para executar un conxunto de sentencias sobre cada unha das propiedades dun obxecto. Produce unha iteración ó través de todas as propiedades dun obxecto. Para utilizalo, por tanto, fai falla coñecer o modelo de obxectos de JavaScript.

Este bucle ten a seguinte sintaxe:

```
for (var propiedade in obxecto) {
  [sentencias]
}
```

- **Exemplo FOR...IN\_1:**