
Funções

Hoje veremos:

- como são criadas e usadas as **funções** em seus programas JavaScript



Objetivos da aula:

Nessa aula vamos tratar das diversas formas de **Funções** isso é:

- Funções **sem parâmetros**
- Funções **com parâmetros**
- Com e sem **retorno de valor e**
- Como **ativar funções** a partir de um hiperlink



Funções

- são **seqüências de comandos**, **realizando uma tarefa** específica, a qual se atribui um **nome**
- Usando funções, você poderá ter partes de código **definidos uma vez** e **executados muitas vezes**



chamada à **funções** predefinidas

- A linguagem apresenta diversas funções já **predefinidas** como as já vistas:
 - **document.write();**
 - **confirm();**
 - **alert();**
 - **prompt();**



chamada ou invocação à função

- Quando se inclui o **nome** da função no programa diz-se que estamos fazendo uma **chamada à função** (ou **invocação à função**).
- Quando o programa é **executado**, isso produz um **desvio** no seu curso para executar os comandos da função.



definição de uma função :

- A **definição de uma função** em JavaScript usa a palavra-chave `function` e segue a seguinte sintaxe **mínima**:

```
function nome()  
{  
    ... comandos ...  
}
```

- Uma vez que foi definida, a função pode ser chamada ou invocada usando seu `nome`: `nome()`;

exemplo :

```
<HTML>
<HEAD><TITLE>Exemplo de function </TITLE></HEAD>
<BODY><script language="javascript">
function EscrevePulandoLinha(texto)
  {
    document.write("<P>" + texto + "</P>");
  }
EscrevePulandoLinha ("Isso ai... ");
EscrevePulandoLinha ("entendeu? ");
</script></BODY>
</HTML>
```

corpo e argumentos da função

- O interior ou corpo da função pode ser composto por qualquer número de comandos, que devem ser sempre contidos pelas chaves.
- Uma função pode receber dados que são os chamados **parâmetros** ou **argumentos**.
- A "passagem de parâmetros" é feita dentro do () que segue ao nome da função.

parâmetros **ou** argumentos :

- Quando há mais de um parâmetro ou argumento eles aparecem separados por vírgulas.
- As Funções devem receber o número de parâmetros que esperam.
- Se forem passados menos parâmetros que o esperado, é atribuído o valor `undefined` aos valores que faltam
- Funções **sem parâmetros** são as que não esperam receber qualquer valor.

passados menos parâmetros

```
<HTML><HEAD>
<TITLE>Exemplo de function com 2 argumentos</TITLE></HEAD>
<BODY><script language="javascript">
function Multiplica(n1,n2)
{
document.write(n1+"*" +n2+"=" + n1*n2);
document.write("<br>");
}
Multiplica(1,2);
Multiplica(1,5);
Multiplica(2);
</script></BODY></HTML>
```

Funções com retorno de valor

- Quando uma função **retorna um valor**, sua chamada pode ser incluída no meio de uma expressão.
- O valor retornado após a execução será utilizado no cálculo desta expressão.
- A forma de fazer uma função retornar um valor é utilizar o comando `return` seguido de um valor ou de uma expressão.

exemplo de Funções com retorno

```
<HTML>
<HEAD><TITLE>Exemplo de function com valor de retorno</TITLE></HEAD>
<BODY><script language="javascript">
function Mult(n1,n2)
{
  return n1*n2;
}
document.write(1 + "*" + 2 +"=" + Mult(1,2)+"<BR>");
document.write(3 + "*" + 2 +"=" + Mult(3,2)+"<BR>");
</script></BODY>
</HTML>
```

Como ativar funções de um hiperlink

```
<HTML><HEAD>
<TITLE>Exemplo de function ativada em link</TITLE></HEAD>
<BODY><script language="javascript">
function Escreve(texto)
{
  document.write("<P>" + texto + "</P>");
}
Escreve("Clique para escrever");
</script>
<A HREF="javascript:Escreve('Oi!')">Aqui</A>
</BODY></HTML>
```

Visualização do Código Fonte

- Em alguns navegadores para ter acesso ao código original da página é necessário acrescentar `view-source:` antes da URL da página na **barra de endereços**.

- Se a página a ser vista tiver como URL

```
file:///C:/tmp/teste.html
```

deve-se colocar na barra de endereços

```
view-source:file:///C:/tmp/teste.html
```

Erros de Sintaxe

- Alguns navegadores não dão nenhum sinal que ocorreu o erro.
- Só é possível descobrir que alguma coisa está errada porque não vai acontecer o que esperávamos.
- Para fazer com que o navegador "mostre" o erro é necessário digitar "JavaScript:" na barra de endereços e teclar **<enter>**, o que causará a exibição da janela com as informações de como se corrigir o erro.

finalizando....

Faça os exercícios sugeridos no texto desta aula.

Não deixe de reler o material se ainda não se sentir seguro!

Faça sua auto avaliação e até a próxima aula, onde veremos os **objetos** (uma forma de agrupar elementos) .
