

Índice

Índice	1
Introdução ao JavaScript Básico.....	2
Configurações.....	2
Usando o Java Script	2
Comandos: <code>script language - alert - /script</code>	3
Variáveis.....	4
Regras para criar uma variável:	4
Comandos: <code>Window.prompt - alert com variável</code>	5
Comandos	6
1. <code>document.write(escreva sua informação);</code>	6
2. <code>document.bgColor=" cor em inglês";</code>	7
3. <code>document.fgColor=" cor em inglês";</code>	7
4. <code>document.title="Título da página";</code>	7
5. <code>document.write (document.lastModified);</code>	7
6. <code>window.open("URL","nome_da_janela","Parâmetros");</code>	7
7. <code>window.location.ref="nome e estensão da outra página ";</code>	8
8. <code>window.location.reload();</code>	8
9. <code>window.close();</code>	8
10. <code>window.status=" ... mensagem ...";</code>	8
Eventos	9
Objetos de Formulário	9
1-) Botão de comando:	9
2-) Botão de radio:	10
3-) Caixa de verificação:.....	10
4-) Caixa de Texto:	10
Operadores do JavaScript	11
Comandos: <code>if e if ..else</code>	13
Comando <code>case</code>	13
Laços de Repetição em JavaScript	14
1-) Estrutura <code>for</code>	14
2-) Estrutura <code>while</code>	15
3-) Estrutura <code>do..while</code>	16
Funções	16
Função simples	16
Função com parâmetro	17
Cálculos em JavaScript.....	18
Uso de uma função de conversão	20

Fonte de pesquisa – site – eadcetec.kit.net/javascript

APOSTILA ELABORADA através da pesquisa feita no endereço acima pela professora
Marli H.G.Cazarin El Kadre

Introdução ao JavaScript Básico

Javascript é linguagem de script para construção de páginas da Web utilizando recursos dinâmicos. Podemos criar efeitos especiais, controlar os dados digitados em um formulário e criar algumas animações. A construção de uma página usando Java, deve ser desenvolvida junto com HTML

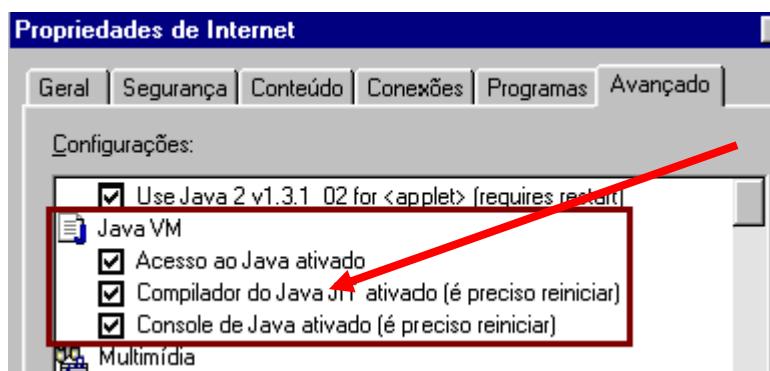
Configurações

Antes de iniciar o curso, é preciso verificar se o seu navegador está configurado para executar os scripts em Javascript.

- Com a internet aberta, clique no menu ferramentas
- Clique na opção: opções da internet



- Na guia avançado alterar o item Java VM, ticando toda as 3 opções, como mostra a figura abaixo:



Usando o Java Script.

O código em Javascript poderá ser inserido em qualquer ponto de sua página, desde que dentro das tags <html>..</html> e iniciando-se com a tag <script language = "Javascript"> e terminando com a tag </script>. Exemplos:

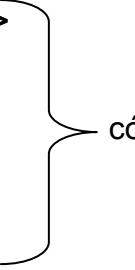
<script language="Javascript">	tag para indicar o início da codificação JAVA Script
-----------------------------------	--

..... código scripts	
</script>	tag para indicar o fim da codificação JAVA Script
</head>	
<body>	
Corpo da página	Codificação normal do HTML
</body>	
</html>	

```

<html><head><title>Programando em Javascript - aula 1</title>
</head>
<body>
<script language="Javascript">
    ..... código scripts
</script>
</body>
</html>

```



código em JAVA entre as tags
<body> </body>

OBS.1 → Como o HTML, uma página usando JAVA SCRIPT, deve ser digitada em um editor de texto, pode ser o BLOCO DE NOTAS e salvar com a extensão HTML ou HTM

Comandos: `script language - alert - /script`

1. `<script language="Javascript">` Comando para informar o navegador qual é a linguagem de scripts que deverá ser interpretada.
2. `alert("..... escreva sua mensagem aqui");` Esta função exibi uma caixa de diálogo com a mensagem que está entre aspas no final da instrução usa-se ponto-e-vírgula (;) para indicar ao navegador o final de instrução. Também pode ser escrito usando o comando window antes de alert. Ex:

```
window.alert("..... escreva sua mensagem aqui ....");
```

3. </script> Comando para finalizar o bloco de instruções em Javascript.

Exercício

- 1 – Criar uma página usando o comando alert para desejar um Bom Dia.

```
<html>
<head><title>exercício nº 1 </title></head>
<script language="Javascript">
alert("Bom dia? Tudo Bem?");
</script>
<body> Primeiro exercício em JAVA SCRIPT
</body>
</html>
```

Resultado



Variáveis

As variáveis permitem guardar dados na memória da máquina durante a execução de programas, páginas ou scripts. Sempre que forem usadas elas estarão prontas para entrarem em ação e receber ou enviar dados conforme a solicitação do usuário.

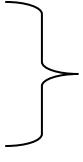
Regras para criar uma variável:

1. Um nome de variável deve sempre começar por uma letra
2. Não utilize símbolos especiais tais como:(! @ # \$ % ^ & * ; " ' , . / ? + = ~`)
3. Procure escrever os nomes das variáveis sempre em minúsculo uma vez que Javascript é sensível a letras maiúsculas e minúsculas
4. Procure dar nomes as variáveis que lembrem o seu conteúdo

Para declarar uma variável em Javascript, você poderá utilizar ou não a denominação **var** (sempre em letras minúsculas), não é preciso especificar o tipo de variável. Se criar uma variável e atribuir um conteúdo texto e na linha seguinte atribuir um conteúdo numérico, o próprio navegador interpreta a instrução e converte normalmente sem nenhum problema.

Exemplo-1

```
var nome="ETE Silvio de Mattos Carvalho";
nome="ETE Silvio de Mattos Carvalho";
```



As duas declarações
são iguais

Comandos: Window.prompt - alert com variável

1. **variável=window.prompt("mensagens para entrada de dados", "título");** - Comando para entrada de dados, usando uma variável que deve ser previamente declarada. Pode ser escrito omitindo-se a palavra window. Ex:

```
variável=prompt("mensagens para entrada de dados", " ");
```

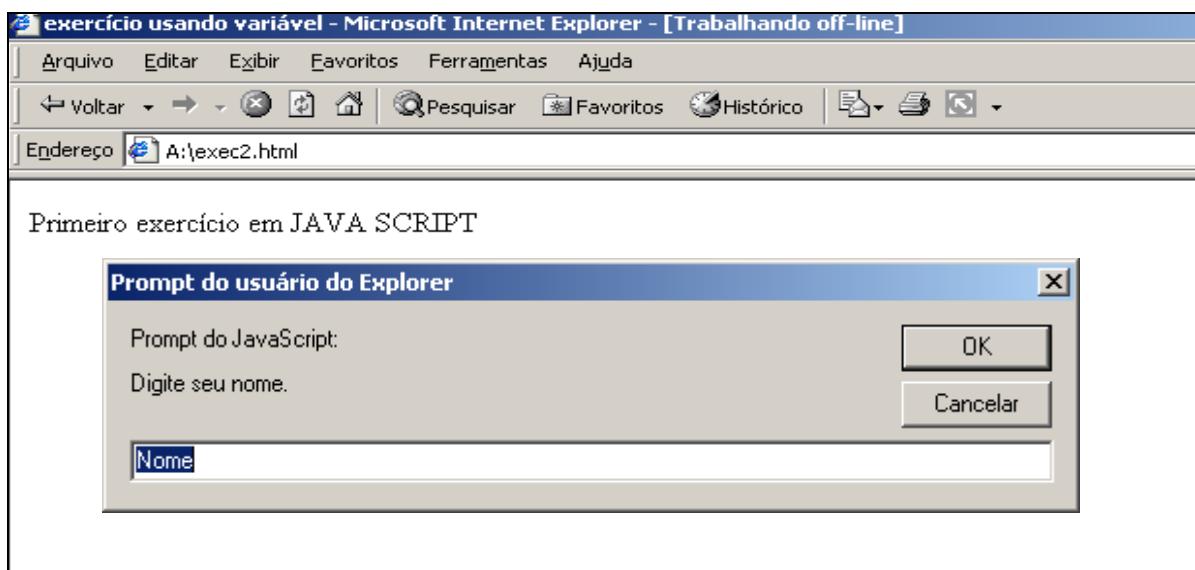
2. **alert("Mensagem "+nome da variável+ " continuação da mensagem");** Comando para mostrar uma pesagem utilizando uma variável. A variável pode vir em qualquer parte da mensagem, sempre entre aspas e com os sinais de + no início e fim.

Exercício.

Escreva um código em Javascript para permitir a entrada de um nome em uma variável e depois exibi-lo em uma caixa de diálogo.

```
<html>
<head><title>exercício usando variável</title></head>
<script language="Javascript">
var nome=window.prompt("Digite seu nome.", "Nome");
alert("Olá "+nome+" Seja Bem Vindo !!!");
</script>
<body> Primeiro exercício em JAVA SCRIPT </body>
</html>
```

Resultado



Comandos

1. **document.write(escreva sua informação);**

Este comando permite escrever qualquer informação na página que pode ser um texto ou o conteúdo de uma variável de memória. Ex:

```
document.write("Programando em Javascript");
```

```
document.write("Olá "+nome+" Seja Bem Vindo !!!");
```

2. document.bgColor=" cor em inglês";

Este comando altera a cor de fundo da página. Ex:

```
document.bgColor="pink";
```

3. document.fgColor=" cor em inglês";

Este comando altera a cor da letra na página. Ex:

```
document.fgColor="blue";
```

4. document.title="Título da página";

Define um título para a página. Ex:

```
document.title="Curso de Informática";
```

5. document.write (document.lastModified);

Exibe a data da última atualização da página. Ex:

```
document.write(document.lastModified);
```

6. window.open("URL","nome_da_janela","Parâmetros");

Comando para abrir uma página em uma nova janela, além de controlar sua posição, tamanho e os controles que deverão aparecer na nova janela. Ex:

```
window.open("URL","nome_da_janela","Parâmetros");
```

URL : é o endereço da página a ser aberta

nome_da_janela : é um nome dado a nova janela a ser aberta que mais tarde poderá ser referenciada em código, usando o atributo target.

Parâmetros: para colocar os controles de janela que deverão ser exibidos. São eles:

- a) status (barra de status): possui dois valores **yes(habilita a exibição)** e **no(desabilita a exibição)**
 - b) location (barra de endereço) : possui dois valores **yes(habilita a exibição)** e **no(desabilita a exibição)**
-

- c) toolbar (barra de ferramentas) : possui dois valores **yes(habilita a exibição)** e **no(desabilita a exibição)**
- d) menubar (barra de menus) : possui dois valores **yes(habilita a exibição)** e **no(desabilita a exibição)**
- e) scrollbars (barra de rolagem) : possui dois valores **yes(habilita a exibição)** e **no(desabilita a exibição)**
- f) left e screenX : definem a posição em pixels da janela a ser aberta a partir do lado esquerdo da janela do navegador. O left é utilizado no Internet Explorer e o screenX no Netscape.
- g) top e screenY: definem a posição em pixels da janela a ser aberta a partir do aldo de cima da janela do navegador. O top é utilizado no Internet Explorer e o screenY no Netscape.
- h) height (altura) : Controla a altura da nova janela. Este valor deverá ser informado em pixels.
- i) width (largura) : Controla a largura da nova janela. Este valor deverá ser informado em pixels.

7. window.location.ref="nome e estensão da outra página ";

Comando para redirecionar o internauta para outra url (página). Ex:

```
window.location.ref="exec1.html";
```

8. window.location.reload();

Este comando serve para atualizar a página atual no navegador. É como se você pressionasse a tecla F5 ou clicar no menu exibir e selecionar atualizar. Ex:

```
window.location.reload();
```

9. window.close();

Este comando fecha a página atual que está aberta no navegador e ao fechar será exibida uma caixa de diálogo do Internet Explorer pedindo para confirmar o fechamento da página.

10. window.status=" ... mensagem ...";

Com este comando, você pode exibir qualquer mensagem na barra de status do navegador. Ex:

```
window.status="Programação em Javascript";
```

Exercício:

Fazer várias páginas usando todos os comandos. Procure agrupar vários comandos em uma única página.

Eventos

Eventos são acontecimentos que ocorrem durante a navegação em uma página da web. Em Javascript existem basicamente dois tipos de eventos **Eventos de Sistemas e Eventos de Mouse**

1-) Eventos de Sistemas: São os que entram em ação automaticamente sem a intervenção do internauta. Temos: **onLoad** (ao carregar) e **onUnload** (ao descarregar) que são escritos na tag `<body>`. Ex:

```
<html><head><title>Eventos em Java Script </title></head>
<body onload="alert('a página está sendo carregada .... Seja Bem Vindo!!!')"
      onUnload = "alert('Até mais volte sempre.....')">
</body>
</html>
```

2-) Eventos de Mouse: São aqueles que acionam ações mediante o uso do mouse, abaixo uma relação dos eventos de mouse:

- **onClick** : Ocorre quando o botão do mouse for clicado
- **onDbclick**: Ocorre quando o botão do mouse sofrer um clique duplo
- **onmouseMove**: Ocorre quando o ponteiro do mouse passar sobre o objeto
- **onmouseOver**: Ocorre quando o ponteiro do mouse ficar acima do objeto
- **onmouseOut**: Ocorre quando você retirar o ponteiro do mouse do objeto
- **onSubmit**: Ocorre quando o internauta clica no botão enviar com objetos de formulários
- **onFocus**: Ocorre quando um objeto ganhar o foco com o click do mouse ou o uso da tecla TAB
- **onChange** : Ocorre quando o conteúdo de uma caixa de texto for alterado

Objetos de Formulário

Os objetos de formulário permitem a construção de páginas com botões de comando, botões de rádio, caixa de texto, caixa de seleção, para construção de um formulário para enviar dados a um destinatário.

Para construir um formulário todas as tags deverão ser escritas entre as tag `<form> </form>`.

1-) Botão de comando:

```
<html>
<head><title>botão</title></head>
<form>
<input type="button" name="bt1" value="Clique no botão" onclick="alert('Você
clicou no botão')">
</form>
</html>
```

input type="button" → botão do tipo alto relevo

Clique no botão

name="bt1" → nome do botão, pode ser qualquer um, desde que não seja repetido.

value="Clique no botão" → rótulo do botão

onclick → evento, ao clicar no botão o comando alert será executado

2-)Botão de radio:

```
<html>
<head><title>botão</title></head>
<form>
<input type="radio" name="opt1" onclick="alert('Você clicou no botão')">Clique
na opção
</form>
```

input type="radio" → botão do tipo botão de opção

Clique na opção

Observe que a frase “ Clique na opção” foi colocada fora da tag input

3-)Caixa de verificação:

```
<html>
<head><title>botão</title></head>
<form>
<input type="checkbox" name="chk1" onclick="alert('Você clicou no botão')">Clique
na opção
</form>
```

input type="checkbox" → botão do tipo caixa de checagem

Clique na opção

4-)Caixa de Texto:

```
<html>
<head><title>botão</title></head>
<form>
<input type="text" name="texto1" size="35"
       maxlength="35" value="Curso de Javascript"
       onChange="alert('Você alterou o conteúdo')">
</form>
</html>
```

input type="text" → caixa de texto

size = “35” → tamanho da caixa de texto em caracteres

maxlength = “35” → quantidade máxima de caracteres que podem ser escritos na caixa de texto.

OnChange=..... → evento – quanto modificar o conteúdo da caixa de texto.

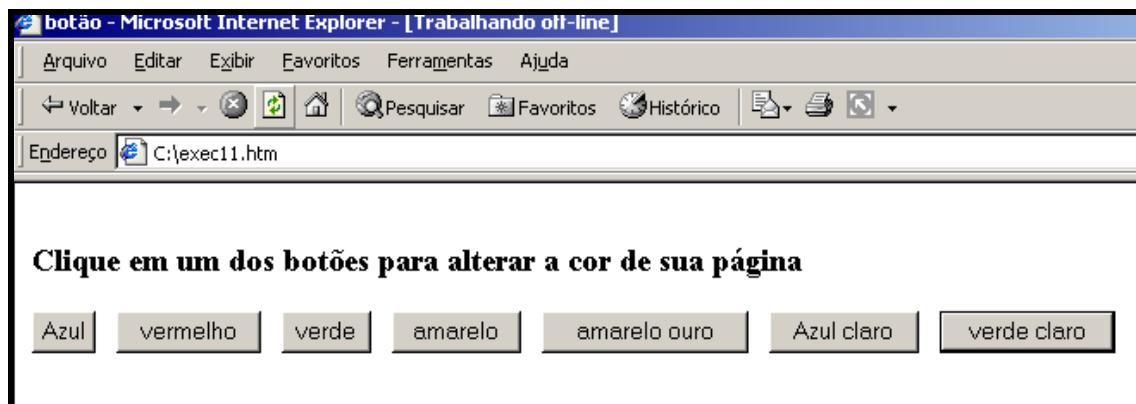
Exemplo:

Vamos criar um script que permita ao internauta escolher através de um clique no botão uma cor de fundo para sua página.

```
<html>
<head><title>botão</title></head>
```

```
<form>
<body>
<align="center"><br><h3> Clique em um dos botões para alterar a cor de sua
página</h3>
<input type="button" name="btazul" value="Azul"
onclick="document.bgColor=('blue')">&nbsp&nbsp
<input type="button" name="btvermelho" value="vermelho"
onclick="document.bgColor=('red')">&nbsp&nbsp
<input type="button" name="btverde" value="verde"
onclick="document.bgColor=('green')">&nbsp&nbsp
<input type="button" name="btamarelo" value="amarelo"
onclick="document.bgColor=('yellow')">&nbsp&nbsp
<input type="button" name="btamareloouro" value="amarelo ouro"
onclick="document.bgColor=('gold')">&nbsp&nbsp
<input type="button" name="btazulclaro" value="Azul claro"
onclick="document.bgColor=('cyan')">&nbsp&nbsp
<input type="button" name="btverdeclaro" value="verde claro"
onclick="document.bgColor=('lime')">&nbsp&nbsp
</form>
</script>
</body>
</html>
```

resultado



 → Este comando serve para colocar um espaço entre os botões.

Exercício

- 1 – Crie um script igual ao exemplo, usando no lugar do button o radio
- 2 – Crie um script para mudar a cor da página e mudar a cor da letra use o button e o radio.

Operadores do JavaScript

Os operadores permitem realizar cálculos, comparações e atribuir valores as variáveis de memória, durante o acesso a página.

1-) Operadores Aritméticos : Servem para realizar cálculos matemáticos.

Operador	Descrição	Exemplo	Resultado
+	Adição	1+5	6
-	Subtração	20-15	5
*	Multiplicação	5*3	15
/	Divisão	12/6	2
%	Módulo(Resto da divisão inteira)	12%6	0
++	Incremento	x=5 x++	x=6
--	Decremento	x=5 x--	x=4

2-) Operadores de atribuição: Servem para guardar informações nas variáveis de memória.

Operador	Exemplo	Significado
$+=$	$x+=y$	$x=x+y$
$-=$	$x-=y$	$x=x-y$
$*=$	$x*=y$	$x=x*y$
$/=$	$x/=y$	$x=x/y$
$%=$	$x\%=y$	$x=x\%y$

3-) Operadores de comparação: Servem para comparar conteúdos de variáveis de memória.

Operador	Significado	Exemplo
$==$	É igual a	$15==15$ é verdadeiro
$!=$	É diferente de	$15!=15$ é falso
$>$	É maior que	$15>13$ é verdadeiro
\geq	É maior ou igual a	$15\geq13$ é verdadeiro
$<$	É menor que	$10<20$ é verdadeiro
\leq	É menor ou igual a	$13\leq35$ é verdadeiro

4-) Operadores lógicos: Servem para criar um expressão com mais de dois valores para serem comparados.

Operador	Significado	Exemplo x=10 e y=5
$\&\&$	E	$(x=10)\&\&(y<10)$ é verdadeiro
$ $	OU	$(x=10) (y=10)$ é verdadeiro

!	NÃO	x!=y é verdadeiro
---	-----	-------------------

Comandos: if e if ..else

A instrução if é uma das mais importantes em JavaScript , é através dela que podemos utilizar o conjunto dos operadores de comparação e determinar um novo curso para seu programa.

1-) Estrutura simples

```
if (condição){
    Bloco de comandos;
}
```

Exemplo: Considere a=10 e b=20

```
if ( a<=b){
    alert("O valor da variável a é menor que b");
}
```

2-) Estrutura composta

```
if (condição){
    Bloco de comandos verdade;
}
else{
    Bloco de comandos falso;
}
```

Exemplo : Considere a=30 e b=20

```
if ( a>=b){
    alert("O valor da variável a é menor que b");
}
else{
    alert("O valor da variável b é menor que a");
}
```

Comando case

É um comando de seleção onde durante a execução do script ele visa encontrar a igualdade.

Sintaxe:

```
switch (variável){
    case 'valor 1': { bloco de comandos;
        • break;
    case 'valor2': {bloco de comandos;
```

```
    ▪ break;
default:{ comando alternativo;
}
```

Exemplo

```
<html>
<head><title>Uso co comando switch em Javascript</title>
</head>
<script language="Javascript">
var mes=window.prompt("Digite um valor entre 1-12","");
switch(mes){
case'1':{alert('Este número digitado corresponde ao mês de JANEIRO');
break;}
case'2':{alert('Este número digitado corresponde ao mês de FEVEREIRO');
break;}
case'3':{alert('Este número digitado corresponde ao mês de MARÇO');
break;}
case'4':{alert('Este número digitado corresponde ao mês de ABRIL');
break;}
case'5':{alert('Este número digitado corresponde ao mês de MAIO');
break;}
case'6':{alert('Este número digitado corresponde ao mês de JUNHO');
break;}
case'7':{alert('Este número digitado corresponde ao mês de JULHO');
break;}
case'8':{alert('Este número digitado corresponde ao mês de AGOSTO');
break;}
case'9':{alert('Este número digitado corresponde ao mês de SETEMBRO');
break;}
case'10':{alert('Este número digitado corresponde ao mês de OUTUBRO');
break;}
case'11':{alert('Este número digitado corresponde ao mês de NOVEMBRO');
break;}
case'12':{alert('Este número digitado corresponde ao mês de DEZEMBRO');
break;}
default:
case'1':{alert('Este número digitado não corresponde a nenhum mês');}
}
</script>
</html>
```

Laços de Repetição em JavaScript

Temos 3 tipos de estrutura de repetição que servem para executar um script várias vezes.

1-) Estrutura for

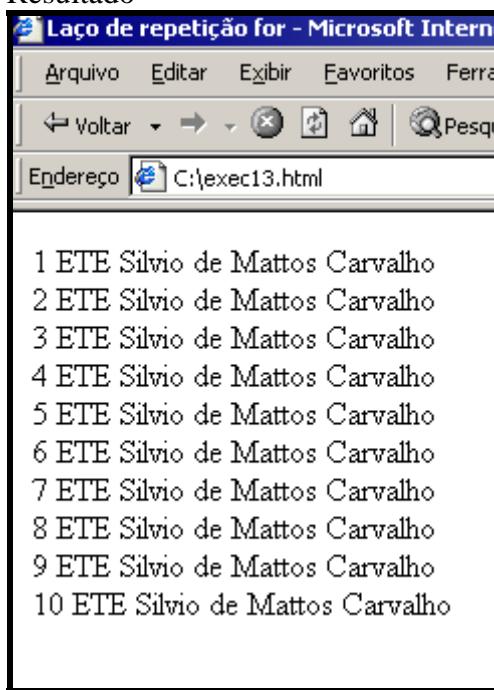
Supondo que devemos imprimir dez vezes o nome **ETE Silvio de Mattos Carvalho**. Os códigos em Java são:

```
<html><head><title>Laço de repetição for</title></head>
```

```
<script language="Javascript">  
  
for(i=1;i<11;i++){  
    document.write(i+" ETE Silvio de Mattos Carvalho"+<br>);}  
  
</script>  
  
</html>
```

A estrutura for é composta de três partes: (**inicialização da variável ; teste condicional e incremento da variável**) assim no exemplo acima a variável i é iniciada com o valor 1. Logo em seguida o teste i<11, isto quer dizer que o laço se repetirá até que o valor da variável i atingir 11. Por fim o incremento a variável i i++ (é igual a i=i+1).

Resultado



2-) Estrutura while

A estrutura while faz a mesma coisa que o for, porém a variável de controle do laço deve ser iniciada antes da instrução while. Veja a colocação no código abaixo:

```
<html><head><title>Laço de repetição fo</title></head>  
  
<script language="Javascript">  
var i=1;  
while(i<11){  
    document.write(i+" ETE Silvio de Mattos Carvalho"+<br>);  
    i++;}  
</script>  
  
</html>
```

3-) Estrutura do..while

A estrutura do.. while faz a mesma coisa que as outras duas anteriores, porém além da variável de controle do laço ser iniciada antes da instrução do..while, a repetição é executada pelo menos uma vez antes da verificação do teste condicional. Veja a colocação no código abaixo:

```
<html><head><title>Laço de repetição fo</title></head>

<script language="Javascript">
var i=1;
do{
document.write(i+" ETE Silvio de Mattos Carvalho"+<br>);
i++;
while(i<11);
</script>

<html>
```

Exercícios

- Desenvolver uma página que contenha código em Javascript que exiba os números de 1 a 15 sucessivamente. Salve com o nome de **conta.htm** e depois abra no seu navegador.
- Agora altere o código e faça a impressão na tela em ordem decrescente , ou seja, de 15 a 1. Salve este arquivo com o nome de **contadesc.htm**

Funções

Uma função é uma rotina com vários comando desenvolvida pelo programador que fica dispostos de uma forma lógica para chegar a um resultado desejado.

Função simples

O exemplo abaixo mostra uma função para digitar um nome e mostra-lo 10 vezes, a função é executada no evento onclick.

```
<html><head><title>Funções de Usuário em JavaScript</title></head>
<script language="Javascript">

function entra(){
var i;
var nome=window.prompt("Digite um Nome", "");
for (i=1;i<11;i++){
document.write(nome,"<br>");
}
</script>
```

```
<body>
<center><input type="button" name="bt1" value="Clique" onclick="entra()">
&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;
<input type="button" name="bt2" value="Fechar" onclick="window.close()">
</center>
</body>
</html>
```

Detalhes do código

onclick="entra()" faz a chamada da função no instante que o ponteiro do mouse for posicionado sobre o botão e efetuar o click.

A função **começa com a palavra FUNCTION e termina com }** ela abre uma caixa de diálogo e aguarda a digitação de um nome. Depois a informação é armazenada na variável **nome**. Logo em seguida um laço de repetição é iniciado, imprimindo na página dez vezes o nome digitado através do comando **document.write()**. Dentro deste comando existe a variável **nome** e depois o código em HTML que realiza a quebra de linha **
**

Após a execução do exemplo, clique no botão voltar do navegador e depois no botão fechar da página que é exibida e confirme seu fechamento.

Função com parâmetro

Quando trabalhamos com parâmetros, significa chamar a função e enviar um valor que possa ser recebido na função solicitada. Em alguns casos precisamos de uma informação para depois completar a execução de uma função. Veja o exemplo abaixo:

```
<html><head><title>Funções de Usuário em JavaScript com parâmentros</title>
<script language="Javascript">

function entra(recebe){
var i;
for (i=1;i<11;i++){
document.write(recebe,"<br>");}
}

</script>
</head>
<body>
<center><input type="button" name="bt1" value="Clique" onclick="entra('Curso de
Javascript')">
&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;
<input type="button" name="bt2" value="Fechar" onclick="window.close()">
</center>
</body>
</html>
```

Detalhes do código

onclick="entra('Curso de Javascript')" refere-se a chamada da função entra, porém com o parâmetro '**Curso de Javascript**' que será enviado para a função chamada.

Na função chamada **function entra(recebe)** , existe uma variável de nome **recebe** que guarda o valor passado como parâmetro. Logo em seguida um laço de repetição é iniciado, imprimindo na página dez vezes o conteúdo da variável **recebe** através do comando **document.write()**. Dentro deste comando existe a variável **recebe** e depois o código em HTML que realiza a quebra de linha **
**

Após a execução do exemplo, clique no botão voltar do navegador e depois no botão fechar da página que é exibida e confirme seu fechamento.

Cálculos em JavaScript

Em Javascript é possível realizar cálculos, basta para isto entender que é necessário o uso de uma função que converta um alfanumérico em um número inteiro, assim o cálculo será efetuado

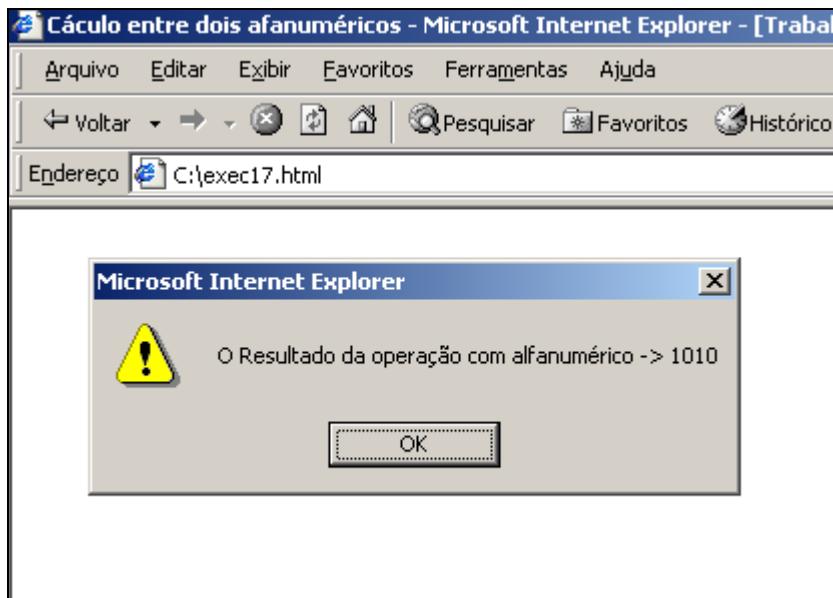
1-) Alfanumérico: Imagine uma operação para somar dois números.

"10"+"10"= 1010

observe as aspas, isto quer dizer que os números são alfanuméricos e portanto foram concatenados(**unidos**) um ao outro. Veja o código abaixo:

```
<html><head><title>Cálculo entre dois alfanuméricos</title>
</head>
<script language="Javascript">
var n1='10';
var n2='10';
total=n1+n2;
alert("O Resultado da operação com alfanumérico -> "+ total);
</script>
</html>
```

Resultado do script

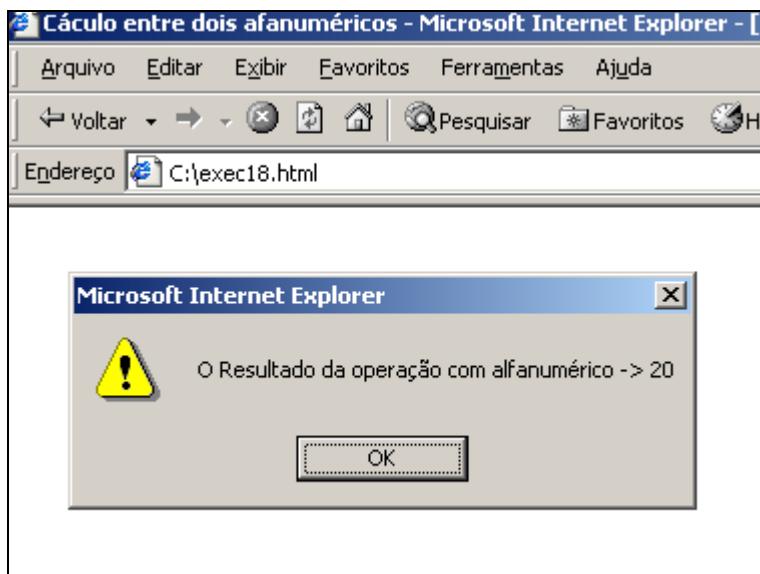


2-) Numérico: Agora para realizar uma operação é preciso utilizar uma variável que guarde este valor entre duas variáveis

total=10+10 o resultado da variável é 20. Veja o código abaixo:

```
<html><head><title>Cálculo entre dois alfanuméricos</title>
</head>
<script language="Javascript">
var n1=10;
var n2=10;
total=n1+n2;
alert("O Resultado da operação com alfanumérico -> "+ total);
</script>
</html>
```

Resultado do script.



Uso de uma função de conversão

Vamos supor que desejamos digitar os dados em duas caixas de textos distintas e depois clicar em um botão que faça o cálculo. Será necessário o uso de uma função matemática que converta os alfanuméricos em inteiro.

Faremos um script para criar a página mostrada abaixo:



Listagem do código completo

```
<htm><head><title>Cálculos entre objeto de Texto</title>
</head>
<script language="Javascript">
function operacao(){
var a=eval(document.formcalc.t1.value);
var b=eval(document.formcalc.t2.value);
var c=a+b;
alert("O Resultado da operação é -> "+c);
}
</script>
<body>
<form name="formcalc">
<h3><center>Cálculo em JavaScript</center></h3>
Digite o primeiro número:<input type="text" name="t1" size="4" maxlenht="4">
<br><br>
Digite o segundo número:<input type="text" name="t2" size="4" maxlenht="4">
<br><br><hr><br><br>
<input type="button" name="bt1" value="Calcular" onclick="operacao()">
&nbsp;&nbsp;
<input type="reset" name="bt2" value="Cancelar"
onclick="document.formcalc.t1.focus()">
```

```
&nbsp&nbsp
<input type="button" name="bt3" value="Fechar"
onclick="window.close()"></center>
</form></body></html>
```

Detalhes do Código

1-) Código em Javascript

```
<script language="Javascript">
function operacao(){
var a=eval(document.formcalc.t1.value);
var b=eval(document.formcalc.t2.value);
var c=a+b;
alert("O Resultado da operação é -> "+c);
}
</script>
```

Definição da função com o nome de **operacao** e logo em seguida as duas linhas para capturar os valores digitados nas caixas de textos converte-los através da função **eval()** ..

Em seguida é realizado o cálculo para adição guardando o resultado na variável **c** e exibido na caixa de saída com o comando **alert()**

2-) Chamada da função calculo() em Javascript

onclick="operacao()"

Esta instrução é simples, pois foi usado o evento de mouse e feita a menção ao nome de uma função já citada no código em Javascript. Observe o detalhe das letras minúscula pois como já sabe se colocar acentos ou digitar qualquer letra em maiúsculo a função não será localizada.

3-) Limpando o dados digitados

```
<input type="reset" name="bt2" value="Cancelar"
onclick="document.formcalc.t1.focus()">
```

Esta linha o **reset** representa recomeço e mais adiante a instrução localiza o formulário(**formcalc**) e depois a primeira caixa de texto chamada aqui de (**t1**). Por último o comando (**focus()**) se encarrega de fazer o cursor ficar piscando nesta caixa caso o internauta queira digitar outro número.

4-) Fechando a Janela

```
<input type="button" name="bt3" value="Fechar" onclick="window.close()">
```

A instrução (**onclick="window.close()"**) executa o fechamento da janela após o clique do mouse.
