

# **JavaScript**

## **Eventos e Objetos Nativos**

Ferramentas para Internet

Marx Gomes Van der Linden

( Material baseado no original de  
Marcelo José Siqueira Coutinho de Almeida )

# Eventos

- Um evento é um acontecimento ocorrido em algum momento durante
- O carregamento da página pelo navegador.  
ou
- A utilização da da página pelo usuário.
- A cada evento, pode-se associar um código Javascript.

# Exemplos de Eventos

- Objeto é clicado pelo mouse.
- O mouse passa por cima de um objeto.
- Texto de um formulário é modificado.
- Formulário é enviado.
- Página é carregada.
- Página é fechada.
- ...

# Eventos

- A maneira mais fácil de se associar um evento a um código Javascript é através das propriedades HTML.
- Todas começam com on...
- Não há distinção entre maiúsculas e minúsculas
- HTML × Javascript

# Eventos em HTML

Evento	Tags	Descrição
<b>onClick</b>	Quase todas	Botão do mouse (aperta e solta)
<b>onMouseDown</b>	Quase todas	Botão do mouse (aperta)
<b>onMouseMove</b>	Quase todas	Seta do mouse se move por cima
<b>onMouseOut</b>	Quase todas	Seta do mouse sai
<b>onKeyDown</b>	Elementos de formulário e <b>body</b>	Pressiona tecla
<b>onSelect</b>	<b>input</b> , <b>textarea</b>	Seleciona texto
<b>onLoad</b>	<b>body</b> , <b>img</b>	Carrega elemento

# Exemplo

```

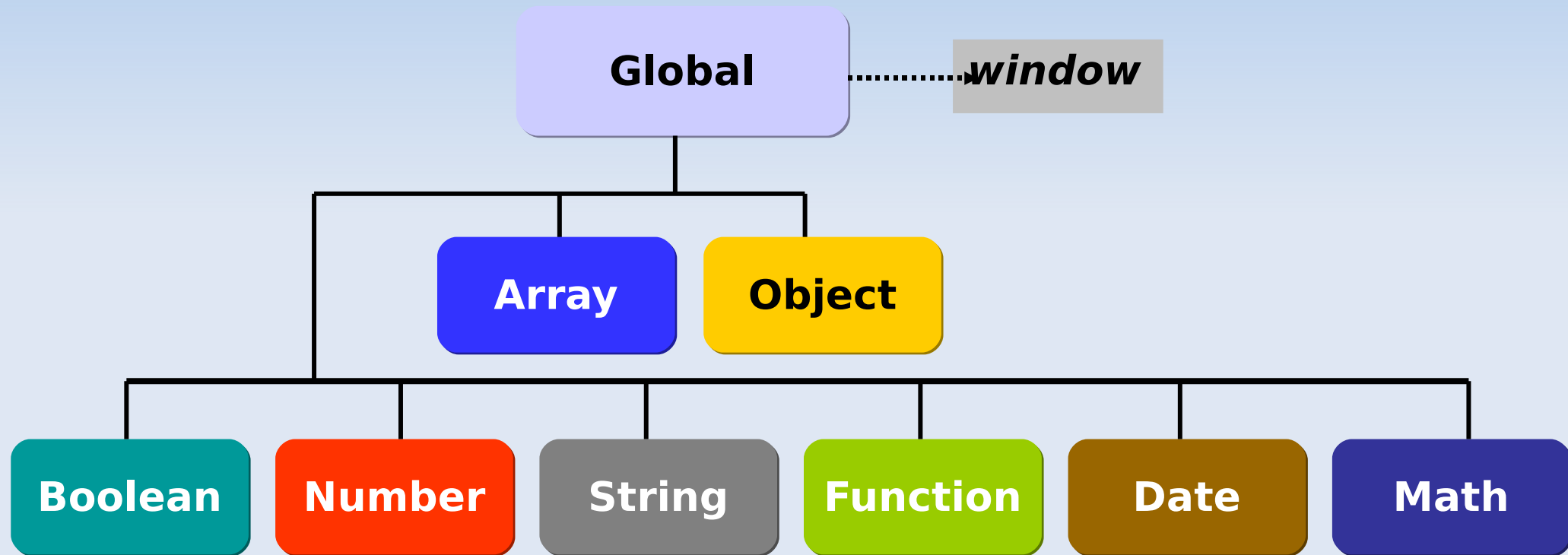
```



# Objetos Nativos Embutidos

- Fazem parte do núcleo da Linguagem
- Não são criados automaticamente
  - Exceção: **Global** e **Math**
- Nem todos os objetos nativos têm construtores

# Objetos Nativos Embutidos





# Objetos Nativos Embutidos

- **Global**
  - Representa o contexto global de execução
  - Não é possível criá-lo
  - Existe antes que qualquer contexto de execução seja inicializado
  - Qualquer outro objeto é propriedade de **Global**
  - Define a propriedade **window**

# Objetos Nativos Embutidos

- **Object**
  - Tipo genérico de objeto usado para representar qualquer objeto criado com **new**

```
meuObjeto = new Object( )
```

# Objetos Nativos Embutidos

- **Object**

- Métodos

- toString()** : transforma qualquer objeto em uma representação string

- valueOf()** : converte qualquer objeto em seu valor primitivo

# Objetos Nativos Embutidos

- Exemplos:

```
// Date é um Object!
```

```
d = new Date();
```

```
document.write(d.toString());
```

```
// Number também é um Object
```

```
num = new Number(1000);
```

```
document.write(n.valueOf());
```

# Number

- Usado para representar números como objetos

```
meuNumero = new Number();  
meuNumero = new Number(1000);
```

# Number

- A principal utilidade é disponibilizar algumas constantes globais:

<code>Number.MAX_VALUE</code>	<code>1.79e+308</code>
<code>Number.MIN_VALUE</code>	<code>5e-324</code>
<code>Number.NaN</code>	<code>NaN</code>
<code>Number.POSITIVE_INFINITY</code>	<code>-Infinity</code>
<code>Number.NEGATIVE_INFINITY</code>	<code>Infinity</code>

# Boolean

- Usado para representar literais `true` e `false` como objetos

```
status = new Boolean(false);  
status = new Boolean(" "); //TRUE  
status = new Boolean("Mmmm"); //TRUE  
status = new Boolean( );  
status = new Boolean(null);  
status = new Boolean(5<4);
```

# String

- Usado para manipular cadeias de caracteres

```
palavra = new String("javascript");  
palavra = "javascript";
```



# String

- Propriedade

```
palavra = "essa não é uma frase longa";  
tamanho = palavra.length;  
tamanho = "essa não é uma frase longa".length;
```

# String

- Métodos

- `link("http://a.com")` : `<a href="http://a.com"> texto </a>`
- `small()` : `<small> texto </small>`
- `big()` : `<big> texto </big>`
- `strike()` : `<strike> texto </strike>`
- `sub()` : `<sub> texto </sub>`

# String

- Métodos

- **sup()** : `<sup> texto </sup>`
- **italics()** : `<i> texto </i>`
- **bold()** : `<b> texto </b>`
- **fontcolor("color")** : `<font color="cor"> texto </font>`
- **fontsize(tamanho)** : `<font size=tamanho> texto </font>`

# String

- Métodos
  - `charAt (n)`
  - `indexOf ("substring")`
  - `indexOf ("substring", inicio)`
  - `lastIndexOf ("substring")`
  - `lastIndexOf ("substring", fim)`
  - `split ("delimitador")`

# String

- Exemplo

```
data = "sexta-feira, 07 de Julho de 1777"  
dia = data.split(",");  
  
document.write(dia[0]); // sexta-feira  
document.write(dia[1]); // 07 de Julho de 1777
```

# Array

- Usado para representar coleções de qualquer tipo na forma de vetores ordenados e indexados.

```
elementos = new Array(4);
```

# Array

- Inicialização

```
elementos =  
    new Array("asa", "nasa", "casa", "usa");
```

```
elementos[0] = "asa";  
elementos[1] = "nasa";  
elementos[2] = "casa";  
elementos[3] = "usa";
```

# Date

- Usado para representar datas.

```
dataqualquer = new Date();
```

- Cria um objeto com a data atual.
- A data utilizada é a data do sistema do usuário



# Date

- Construtores

```
data = new (ano,mês,dia) ;
```

```
Exemplo: data = new (04,11,20) ;
```

```
data = new Date(  
ano,mês,dia,hora,minuto,segundo  
);
```

```
Exemplo: data =  
new Date(04,11,20,12,06,00) ;
```

# Date

- Construtores

```
data = new Date("Mês dd, aa hh:mm:ss");
```

```
Exemplo: data = new Date(  
"May 06,72 00:20:10"  
);
```

```
data = new Date(milissegundos);
```

```
Exemplo: data = new Date(1838646);
```

# Date

- Métodos
  - `getDate()`
  - `getDay()`
  - `getHours()`
  - `getMinutes()`
  - `getMonth()`
  - `getSeconds()`
  - `getFullYear()`

# Date

- Métodos
  - `setDate (dia)`
  - `setDay (dia)`
  - `setHours (hora)`
  - `setMinutes (minuto)`
  - `setMonth (mês)`
  - `setSeconds (segundo)`
  - `setYear (ano)`

# Janelas

- Podemos manipular a janela do browser de várias formas
  - Modificando o seu tamanho, aparência ou posição
  - Abrir e fechar janelas
  - Transferir informações entre janelas e frames
  - Criar janelas de diálogo

# Janelas

- Window
  - Representa a janela do browser onde roda o script
  - Não há um construtor com este nome
  - Permite o acesso a propriedades que referenciam eventuais sub-janelas, a janela pai (se existir) ou frames

# Janelas

- Propriedades
  - `defaultStatus` (r+w)
  - `status` (r+w)
  - `document`
  - `history`
  - `location`
  - `navigator`
  - `opener`

# Janelas

- Propriedades
  - `self`
  - `window`
  - `frames`
  - `length`
  - `parent`
  - `top`



# Janelas

```
<script>
```

```
a = new Array(  
"Socorram-me! Subi no ônibus em Marrocos");
```

```
</script>
```

```
<body>
```

```
<a href="http://www.wikipedia.com/"  
onmouseover="window.status=a[0];return true;"  
onmouseout="window.status='';return true;">  
Palíndromo </a>
```

```
</body>
```

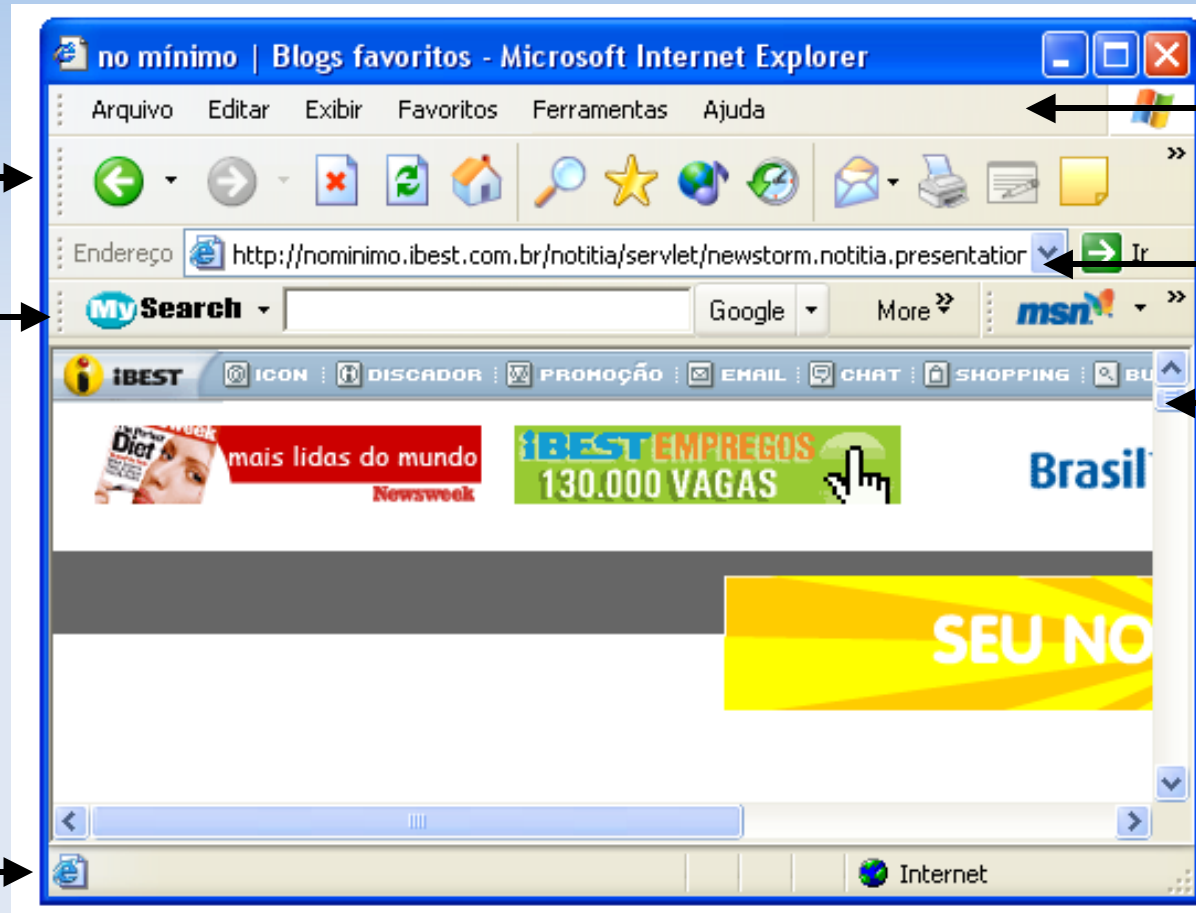
# Janelas

- Métodos
  - `open (URL)`
  - `open (URL, nome, características)`
  - `close ()`
  - `blur ()`
  - `focus ()`

# Janelas

- Características
  - `height=valor`
  - `width=valor`
  - `resizable`
  - `toolbar`
  - `menubar`
  - `location`
  - `status`
  - `scrollbars`

# Janelas



Barra de Menus

Localização

Barra de Rolagem

Height (largura)

# Janelas

- Exemplos

```
open("p1.html", "teste",  
"height=120,width=100,resizable");
```

```
ref = open("p1.html", "t2",  
"height=120,width=100,resizable,toolbar,directories"  
);
```

# Janelas

- Eventos
  - **onBlur**: janela deixa de ser ativa
  - **onError**: ocorre um erro na carga do documento
  - **onFocus**: janela se torna ativa
  - **onLoad**: documento é carregado
  - **onUnload**: documento é descarregado

# Janelas

- Exemplos

```
<body onLoad="window.open('banner.html');">
```

```
<body onLoad="open('teste.html')"  
onunload="alert('obrigado')">
```

```
<body onBlur="focus() ">
```

```
<body onUnload="open('teste.html')";
```

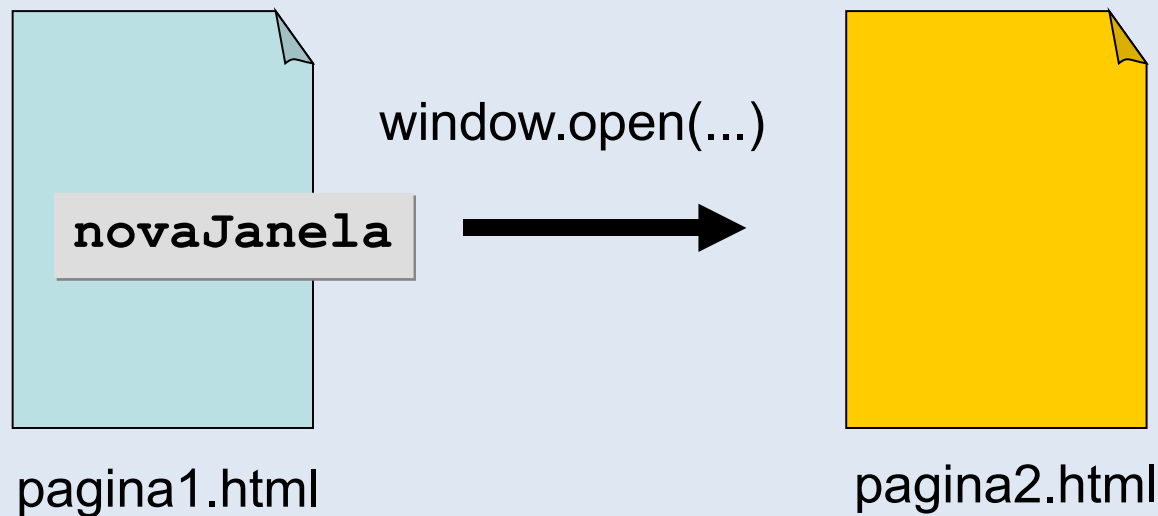
# Janelas

- Comunicação entre janelas
  - Para passar informações para uma janela recém criada é necessário obter uma referência para ela.
  - Isto só é possível se a janela for criada usando JavaScript.



# Janelas

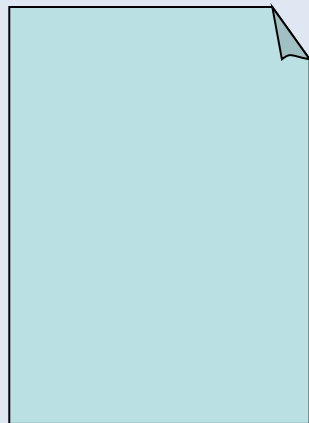
```
novaJanela = window.open("pagina2.html");  
novaJanela.document.write("Olá, Mundo");  
novaJanela.focus();  
novaJanela.close();
```



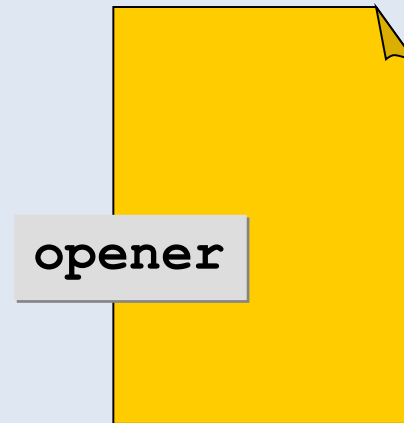
# Janelas

```
<form>  
<input type="text" name="nome">  
<input type="text" name="mensagem">  
</form>
```

```
opener.focus ( );  
opener.document.forms[0].elements[1].value=  
"Oi!!!!";  
opener.close ( );
```



pagina1.html



pagina2.html

# Browser

- Navigator
  - Representa as propriedades do browser
  - Permite o desenvolvimento de páginas personalizadas
  - No Netscape Navigator é uma propriedade global, mas no Internet Explorer é uma propriedade de window

# Browser

- Propriedades
  - **userAgent**
    - Informação contida no cabeçalho userAgent.
    - Mozilla/4.0 (compatible; MSIE 4.0; Windows 98)
  - **appName**
    - Contém nome interno do browser (String)
    - Mozilla/4.0
  - **appVersion**
    - Versão da Aplicação (String)
    - 4.0 (compatible; MSIE 4.0; Windows 98)

# Browser

- Propriedades
  - **appName**
    - Contém o nome oficial do browser (String)
    - Netscape, Microsoft Internet Explorer
  - **mimeTypes**
    - Tipos suportados pelo browser direta ou indiretamente (Array)\*
  - **plugins**
    - Plugins instalados (Array)\*

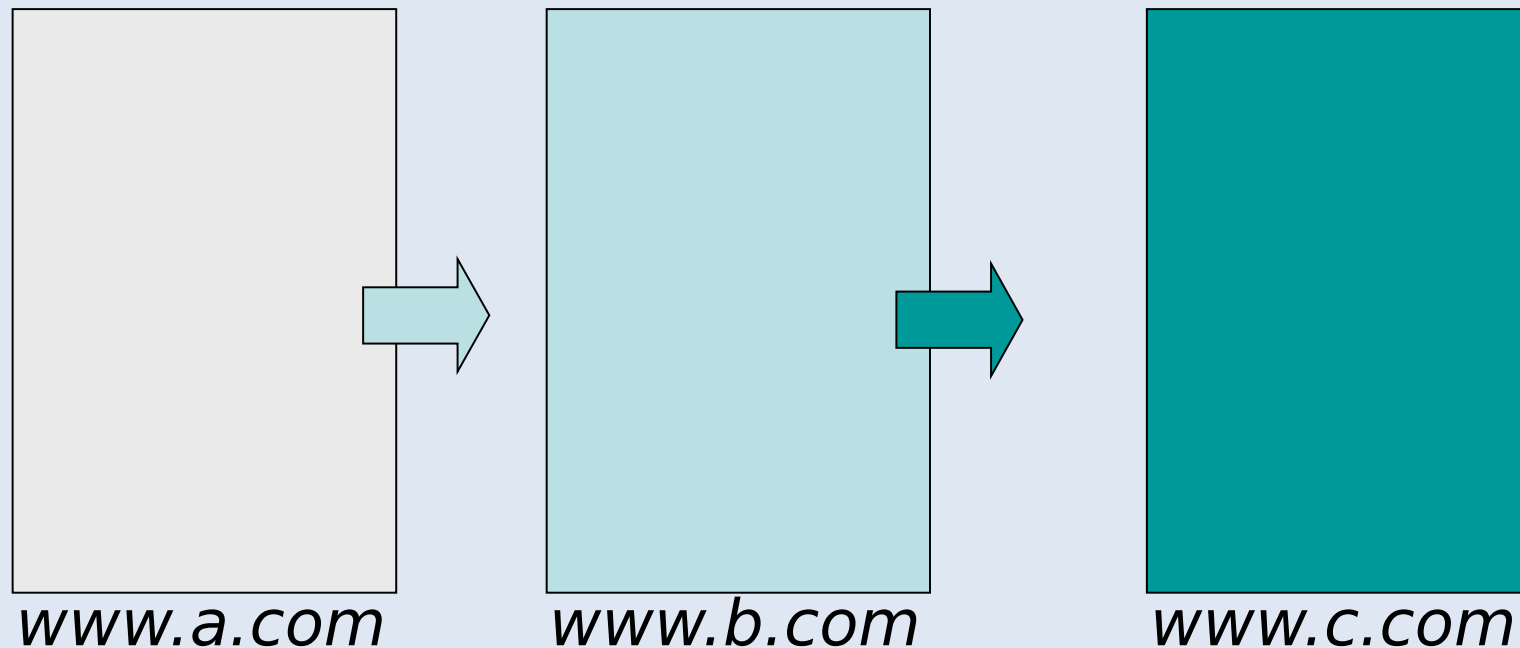
# Browser

```
for(i in navigator){  
    document.write("<b>" + i + ":" +  
                    navigator[i] + "<br>");  
}
```

```
if (navigator.appName == "Netscape") {  
    location.href =  
        "http://www.netscape.com";  
}  
else if (navigator.appName ==  
        "Microsoft Internet Explorer"  
) {  
    location.href =  
        "http://www.microsoft.com";  
}
```

# Navegação

- History
  - Representa os lugares visitados durante uma determinada sessão



# Navegação

- Propriedades
  - **length**
    - Acessível pelo programador(Number)
  - **current** (String)
  - **next** (String)
  - **previous** (String)



# Navegação

- Métodos
  - `go (+n)`
  - `go (-n)`
  - `back ()`
  - `forward ()`

# Navegação

- Location
  - Representa a URL cujo documento está sendo exibido

```
location.href =  
"http://www.abracadabra.com"
```

# Navegação

- Propriedades
  - **href**: A URL completa  
“http://www.abracadabra.com:8080/  
sub/dir/compra.asp?name=cookie#parte2”
  - **protocol**: http
  - **host**: www.abracadabra.com
  - **port**: 8080

# Navegação

- Propriedades
  - **port**: 8080
  - **hostname**: www.abracadabra.com:8080
  - **pathname**: /sub/dir/compra.asp
  - **hash**: #parte2
  - **search**: ?name=cookie

# Navegação

- Métodos
  - **reload()**
    - Recarrega a página caso ainda não tenha sido modificada
  - **reload(true)**
    - Recarrega a página incondicionalmente
  - **replace("URL")**
    - Carrega a página especificada pela URL

# A página HTML

- Document
  - Representa o documento HTML sendo acessado no momento.
  - É uma propriedade de window

```
window.document.write("Olá, mundo cruel!");  
document.write("Olá, mundo cruel!");
```

# A página HTML

- Propriedades (read/write)
  - `bgColor`
  - `fgColor`
  - `linkColor`
  - `alinkColor`
  - `vlinkColor`

```
<form>
<input type="radio" name="x" value="1"
onclick="document.bgColor='blue';
document.fgColor='white'">
<strong> AZUL </strong><br>

<input type="radio" name="x" value="2"
onclick="document.bgColor='yellow';
document.fgColor='green'">
<strong> AMARELO </strong><br>

<input type="radio" name="x" value="3"
onclick="document.bgColor='black';
document.fgColor='white'">
<strong> PRETO </strong><br>
</form>
```



# A página HTML

- Propriedades (read)
  - `title`
  - `links`
  - `applets`
  - `anchors`
  - `plugins`
  - `images`

# A página HTML

- Propriedades (read)
  - `location` (ou URL)
  - `lastModified`
  - `cookie` (r+w)