



JavaScript第7回目

4-4 windowオブジェクト

発表者 加藤 友宏
発表日 2006/11/24



ウィンドウを参照するためのプロパティ

- Webブラウザを起動 windowオブジェクトは1つ
- 複数ウィンドウや複数のフレームを作成 windowオブジェクトが複数
- Windowオブジェクトを参照するためのプロパティ

プロパティ	説明
window	現在のwindowオブジェクト自体を参照
self	現在のwindowオブジェクト自体を指し示す。(windowと同じ)
top	Webブラウザで最初に開いたwindowオブジェクトを指し示す。
parent	フレーム機能を使用している場合、親ウィンドウもしくは親フレームを指し示す。
opener	open()メソッドで新規のwindowオブジェクトを作成した場合、作成した側のwindowオブジェクトを指し示す。

読み出し専用のプロパティ

windowオブジェクトが1つのときは、

window, self, top, parentは自分自身を示す



ステータスバー関連のプロパティ

- ステータスバーは、リンク先の表示やシステムメッセージの表示に使われる。
- ステータスバー関連のプロパティ

プロパティ	説明
defaultStatus	ステータスバーにデフォルトで表示する文字列を設定する
status	ステータスバーに表示する文字列を設定する

statusとdefaultStatusの違い

statusプロパティは、ステートメントが実行された時点で表示
しかし、何らかのメッセージで上書きされたら何も表示されない。

defaultStatusプロパティは、何らかのメッセージで上書きされても
ステータスバーが空いたら再び表示する



ダイアログボックス関連のメソッド

- アラートダイアログボックスを表示する。

メソッド	alert()
構文	windowリファレンス.alert(message)
引数	message - ダイアログボックスに表示する文字列
説明	引数messageとして渡された文字列と「OK」ボタンを表示するアラートダイアログボックスを作成

- 確認のためのダイアログボックスを表示する。

メソッド	confirm()
構文	windowリファレンス.confirm(message)
引数	message - ダイアログボックスに表示する文字列
説明	引数messageとして渡された文字列と「YES」ボタンと「NO」ボタンを表示するアラートダイアログボックスを作成

```
answer = confirm("あなたは、18歳以上ですか?");
```

「YES」ボタン answerにtrue

「NO」ボタン answerにfalse



ダイアログボックス関連のメソッド

- ユーザに文字列を入力させる。

メソッド	prompt()
構文	windowリファレンス.prompt(message,default)
引数	message- ダイアログボックスに表示する文字列 default- デフォルトとして使用する文字列
説明	ユーザに何らかの値を入力させるためのダイアログボックスを表示。引数defaultはテキストボックスにデフォルトで表示する内容



新規のウィンドウをオープン/クローズする

■ 新規のウィンドウを開く

メソッド	open()
構文	windowリファレンス.open(URL, name, "option")
引数	URL - 新規ウィンドウに表示するHTMLドキュメントのURL name - ウィンドウ名 option - windowの特性を現す文字列
戻り値	新規のwindowオブジェクトへの参照
説明	open()メソッドは新規のウィンドウを開き、そのwindowオブジェクトへの参照を戻り値とする

・新規のウィンドウを開いてメッセージを表示する例

```
var newWin = window.open()
newWin.document.open();
newWin.document.write("これは新規のウィンドウ");
newWin.document.close();
```

新規のウィンドウをオープン/クローズする

- 第3引数のオプションでウィンドウの大きさやツールバーの有無などを、ひとかたまりの文字列として指定。
- それぞれのオプションは、「オプション名=値」で記述

オプション	値	説明
toolbar	yesまたはno	ツールバーを表示するかどうか
location	yesまたはno	ロケーションボックス(アドレスバー)を表示するかどうか
directories	yesまたはno	ディレクトリボタンを表示するかどうか
status	yesまたはno	ステータスバーを表示するかどうか
menubar	yesまたはno	メニューバーを表示するかどうか
scrollbars	yesまたはno	スクロールバーを表示するかどうか
resizable	yesまたはno	ウィンドウサイズを可変にするかどうか
width	ピクセル値	ウィンドウの幅
height	ピクセル値	ウィンドウの高さ

幅300ピクセル、高さ200ピクセルのツールバー付きウィンドウ

```
var newWin = window.open("", "newWin",  
    "width=300, height=200, toolbar=1");
```



新規のウィンドウをオープン/クローズする

- ウィンドウを閉じる

メソッド	close()
構文	windowリファレンス.close()
説明	windowリファレンスで指定したウィンドウを閉じる。



タイマー関連のメソッド

- 指定した時間度に何らかの処理をさせたい、一定時間ごとに同じ処理をさせたいときに使用
- タイマーをセット `setTimeout()`メソッド (一回だけ実行)
- 停止する `clearTimeout()`メソッド
- 一定時間間隔で繰り返し処理を実行 `setInterval()`メソッド

メソッド	<code>setTimeout()</code>
構文	<code>windowリファレンス.setTimeout("function", delay)</code>
引数	function - 指定した時間後に行う処理 delay - functionが実行されるまでの時間(ミリ秒単位)
戻り値	タイマーを識別するID(タイマーID)
説明	タイマーをスタートさせ、delayで指定した時間後にfunctionで指定した処理を行う。



タイマー関連のメソッド

- タイマーを停止するメソッド

メソッド	clearTimeout()
構文	windowリファレンス.clearTimeout(timeoutID)
引数	timeoutID - setTimeout()メソッドの戻り値で取得したタイマーID
説明	timeoutIDで指定したIDのタイマーを停止する

- 一定時間間隔で繰り返し処理を実行するメソッド

メソッド	setInterval()
構文	windowリファレンス.setInterval("function", delay)
引数	function - 指定した時間後に行う処理 delay - functionが実行されるまでの時間(ミリ秒単位)
戻り値	タイマーを識別するID(タイマーID)
説明	タイマーをスタートさせ、delayで指定した間隔でfunctionで指定した処理を行う。タイマーを止めるにはclearTimeout()メソッドで...



その他のメソッド

- フォーカスを失わせるメソッド

メソッド	blur()
構文	windowリファレンス.blur()
説明	ウィンドウのフォーカスを失わせ、フォーカスを失ったウィンドウは背面に移動する

- フォーカスをウィンドウに移動するメソッド

メソッド	focus()
構文	windowリファレンス.focus()
説明	blur()メソッドとは逆に、フォーカスをウィンドウに移動する



その他のメソッド

- 指定した座標までスクロールするメソッド

メソッド	scroll()
構文	windowリファレンス.scroll(x-coordinate, y-coordinate)
引数	x-coordinate -- x座標 (ピクセル単位) y-coordinate -- y座標 (ピクセル単位)
説明	指定した座標までウィンドウをスクロールする。座標は、ウィンドウの左隅を原点(0,0)としてピクセル単位で指定



Windowオブジェクトのイベントハンドラ

- Windowオブジェクトの主なイベントハンドラ

イベントハンドラ	呼び出されるタイミング
onload	HTMLドキュメントがロードされたとき
onunload	HTMLドキュメントがアンロードされたとき
onblur	ウィンドウがフォーカスを失ったとき
onfocus	ウィンドウがフォーカスを得たとき
onerror	JavaScript実行中に何らかのエラーが生じたとき

HTMLタグ内に記述するときは、
<body>タグの属性として記述する

- イベントハンドラをプロパティとして記述する方法

```
window.onload = new Function("alert('ようこそ');");
```



練習問題

- スロットマシンを作ってください。
 - コインの配当は、各自で適当に...
 - タイマー関連のメソッドを使用して、数字に動きを付けてください。
 - スタートとストップのボタンを作成し、コインがなくなったら、ゲームオーバーで。