

HTML TIPS & TRICKS

第 28 回

誰よりも早く 最新のHTMLを使ってみよう

藤井 幸孝 大内 勇

インターネットエクスプローラ5が発表されてから1か月が過ぎた。オートコンプリートなどの機能ばかりが取り上げられがちなIE 5だが、このコーナーではHTMLやスタイルシート、JavaScriptを丁寧に解説していく予定だ。それでは今月もIE 5の新しいスタイルシートをはじめとしたTIPSとTRICKSをお届けしよう。パズルもIE 5に関するものなのでぜひ挑戦してほしい。



CD-ROM収録先 A Magnavi Ip9906 HtmTips
今月号のTIPSをすべてCD-ROMに収録!

このコーナーを楽しむために

最新のHTMLを使う際に、どうしても避けて通れないのがWWWブラウザの互換性の問題だ。そこでこのコーナーでは、TIPSごとにブラウザの対応状況をアイコンで表している(4月10日現在)。これを参考に使用するWWWブラウザを選んでほしい。



インターネットエクスプローラ3以上



インターネットエクスプローラ4以上



インターネットエクスプローラ5以上



ネットスケープナビゲーター3以上



ネットスケープナビゲーター4以上



5月号「HTMLパズルに挑戦しよう」の解答

先月号のパズルの解答を発表しよう。第1問、第2問ともにさまざまなテクニックを使ったスクリプトが寄せられた。第1問では動作するものを正解としたが、シンプルで汎用性の高い下の解答がおすすめだ。第2問ではopenメソッドを使用した解答は残念ながら不正解とさせていただいた。

ANSWER ① フレームをクリアせよ!

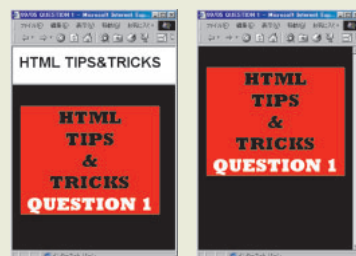
JavaScriptにおけるフレームの関係を知っていれば簡単に解ける。自分自身のフレームはselfで、最上位のウィンドウはtop。これがこのクイズの鍵だ。この関係を判定して、最上位のウィンドウに自分自身のフレームのURLを読み込ませる。

```
if ( self != top ) {  
    top.location = self.location;  
}
```

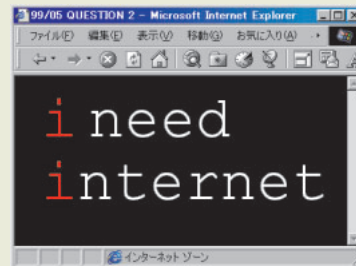
ANSWER ② ブラウザーのサイズを指定せよ!

resizeToメソッドでかつこの中に横の幅と縦の幅を指定すれば、ブラウザーのサイズが変更される。ページを読み込んだときにこのメソッドを実行させるには、BODYタグのonLoadイベントに指定する。

```
<BODY onLoad="resizeTo(400,300)">
```

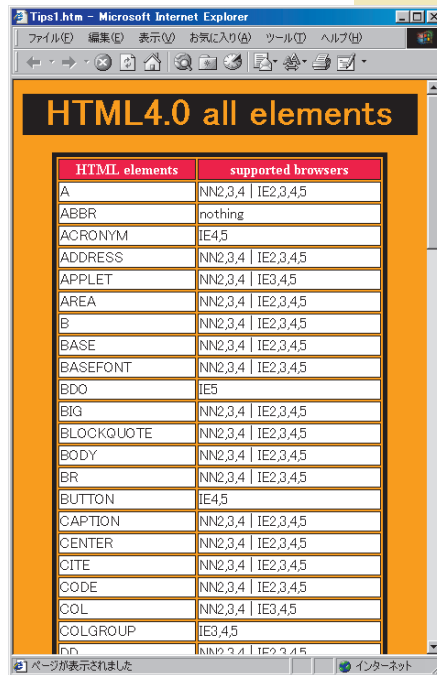
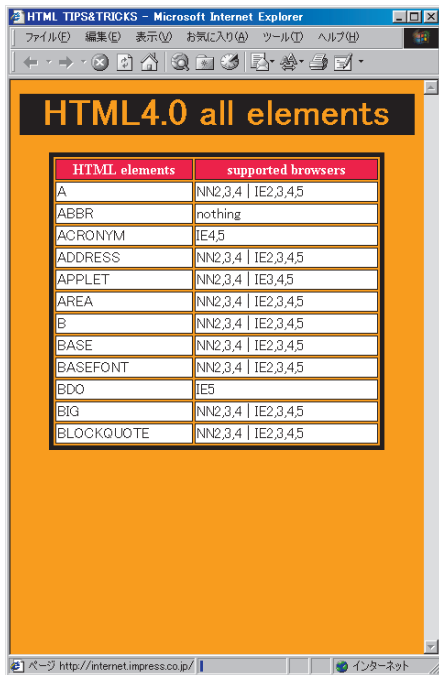


正解者：丹羽光一、池永達美、鹿倉隆、谷口勝宣、うおまさ@home、杉本敬治、ばおー、齊藤貴志、田中博英、堀江(敬称略)



正解者：村田正彦、黒咲瑠架、池永達美、鹿倉隆、谷口勝宣、加島佳代子、杉本敬治、ばおー、岡野岳飛、齊藤貴志、田中博英、堀江(敬称略)

テーブルをリアルタイムに表示させる



ホームページの訪問者がブラウジングを快適と感じるのは、アクセスするとすぐに表示されるページだ。しかし、巨大なテーブルを使うと表示までに非常に時間がかかってしまうことはご存じだろう。これはブラウザがテーブルの内容をすべて読み込んでレイアウトを整えてから表示するためだ。だからといって、テーブルを使わずに一覧表などを作ることは不可能だ。今までのウェブ制作者はテーブルを使った場合に表示に時間がかかるのはしょうがないとあきらめていたが、IE5ではこの問題を解決してくれる新しい機能が登場した。ホームページで大きなテーブルを使っている人はぜひ試してみよう。



```
<TABLE STYLE="teble-layout:fixed">
<COL WIDTH="150"><COL WIDTH="200">
<TR><TH>HTML elements</TH><TH>supported browsers</TH></TR>
<TR><TD>A</TD><TD>NN2,3,4 | IE2,3,4,5</TD></TR>
.....
</TABLE>
```

Point

それではさっそく上のソースを見てほしい。
`<TABLE>` タグのスタイル指定で「table-layout」という見えないプロパティが指定されていることに気付くだろう。これがIE5から登場した新しいスタイルシートだ。W3Cの策定したCSS 2の仕様にも取り入れられている。このプロパティには下の2つの値を指定できる。

auto 文字数に依存してセル幅を自動調整
 fixed `<COL WIDTH>` に依存してセル幅を固定

「auto」を指定した場合は、普通のテーブルと同じでリアルタイムには表示されない。つまりすべてのセルの内容を読み込んでそれぞれの大きさを計算し終わってからテーブル全体が表示される。「fixed」を指定した場合は、次に説明する`<COL WIDTH = ".....">`によってセル幅が固定されるため、高速

なりリアルタイム表示が可能になる。つまり読み込まれた順に列が次々と表示される。

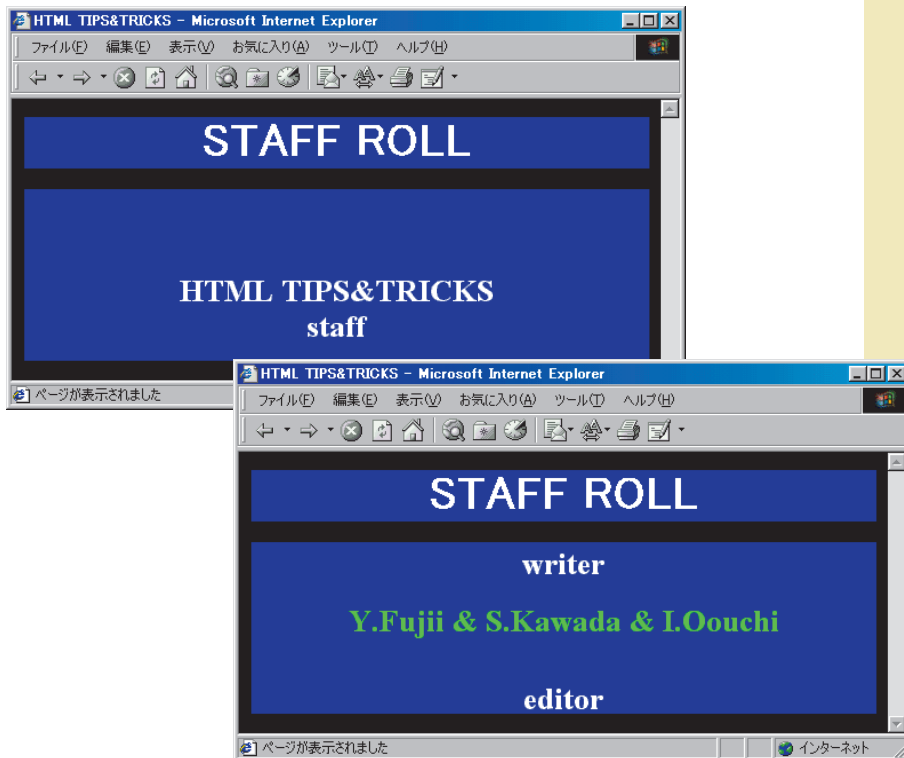
`<COL>` タグはHTML 4.0から採用されたテーブル関連のタグで、テーブル内の列に関する情報をあらかじめ指定するものだ。一般的には`<TABLE>`の開始タグの直後に`<COL>` タグを置く。今回のサンプルでは2列のテーブルを使ったので`<COL>` タグ指定が2つになっている。つまり、列の数だけ`<COL>` タグを指定する必要があるのだ。`<COL>` タグの使い方に慣れてない人はこのことを覚えておこう。

サンプルでは`<COL>` タグのWIDTH属性で1列目の幅を150ピクセル、2列目の幅を200ピクセルと指定している。「table-layout:fixed」を指定すると、`<COL>` タグで指定した幅でテーブル内のすべ

てのセルを表示する。最初に指定されたセルの幅に従ってすべてのセルの幅を固定して表示するため、ブラウザは読み込んだデータの内容によってテーブルの幅を変える必要がなくなるわけだ。そのためテーブルのリアルタイム表示が可能になるのだ。「table-layout」をサポートしていないブラウザではリアルタイムに表示されないが、このプロパティがテーブルの内容に影響を及ぼすことはない。

今回紹介した「table-layout」は、IE5でしか使えないスタイルシートだが、ホームページで巨大なテーブルを作っている人にとってはたいへん便利な機能だ。ページの表示に時間がかかってしまうと悩んでいる人はさっそく使ってみよう。

スタッフロールを作る



```
<MARQUEE DIRECTION="UP" LOOP="1" SCROLLAMOUNT="2">
HTML TIPS&TRICKS<BR>
staff .....
</MARQUEE>
```



映画の最終シーンでは、画面の下から上に向かってスタッフの名前が流れる。これはスタッフロールと呼ばれる。今回紹介するTIPSは、このスタッフロールをホームページで作ってしまおうというものだ。文字を動かすので、スタイルシートとJavaScriptを使ったダイナミックHTMLかと思われるかもしれないが、今回はこのスタッフロールをマーカーキーを使って作ってみた。ソースファイルを見れば一目瞭然、非常に簡単になっている。単純なHTMLでこれだけ動きのあるページを作れるのだから、これはぜひ使ってみるべきだろう。それではさっそく、マーカーキーをどんなふうに使っているのか見てみよう。

POINT

見た目の派手さに反してソースが簡単なことに驚いた人もいるだろう。マーカーキーといえば右から左方向にテキストがスクロールするものが一般的だが、このサンプルでは下から上に向かってテキストをスクロールさせている。これは<MARQUEE>タグのあまり使われていない属性「DIRECTION」を指定しているからだ。「DIRECTION」とはマーカーキーのスクロール方向を指定する属性だ。IE 3までは「left」と「right」しか指定できなかったが、IE 4からは「up」と「down」も指定できるようになった。当然「up」は上方向へのスクロールで、「down」は下方向へのスクロールになる。サンプルでは「DIRECTION」属性とともに「LOOP」属性も指定している。これはスクロールの回数を表すものだ。今回は「LOOP="1"」として1回スクロールするだけで動作を停止させているが、何回も繰り返しスクロールさせたい場合は、「LOOP="

0"」とすればOKだ。

3つ目の属性は「SCROLLAMOUNT」で、これはスクロールの速度を指定するものだ。数値が小さいほど速度が遅く、大きいほど速くなる。以上で今回のTIPSは完成だ。スタッフロールに限らず、なんらかのメッセージをスクロールさせても同じ効果が出せるので、ぜひ使ってみてほしい。

参考までに、<MARQUEE>タグのその他の属性について少し触れておこう。

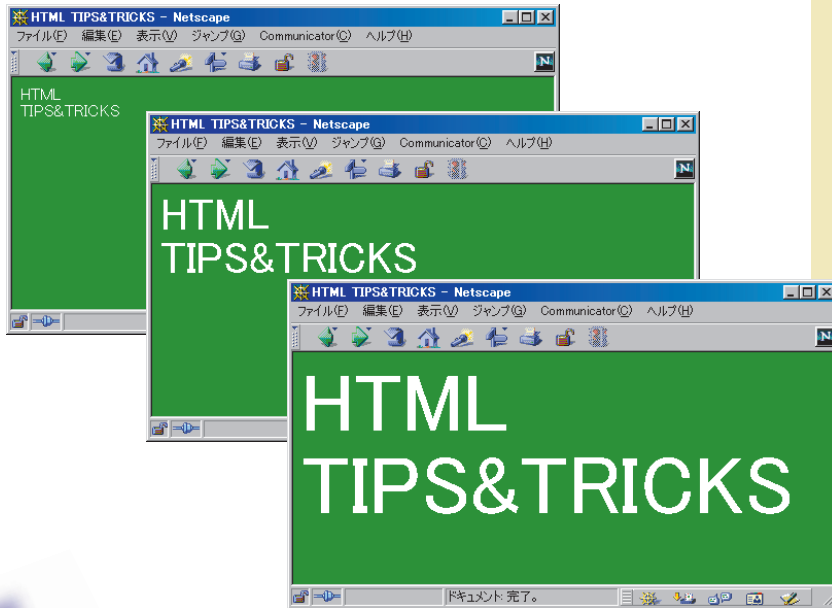
```
WIDTH="ピクセル数(または%)"  
マーカー枠の幅  
HEIGHT="ピクセル数(または%)"  
マーカー枠の高さ  
BEHAVIOR="alternate"  
左右(または上下)に振り子移動
```

```
BEHAVIOR="slide"
```

```
LOOP="1"と同時に指定すると、  
文字が端に達したときに停止する  
SCROLLEDELAY="ミリ秒"  
何ミリ秒おきに動かすかを指定
```

今回のサンプルに、これらの属性を組み合わせるとオリジナリティーの高いマーカーキーを作ってみよう。たとえば、「BEHAVIOR="slide"」を指定した<MARQUEE>タグを複数使って、スタイルシートを組み合わせると、ブラインドカーテンのような効果を実現することもできる。このブラインドカーテン・マーカーキーは紙面の都合上ここでは紹介できないが、本誌の付録CD-ROMに収録されているので、ソースを参考にしてほしい。

テキストをズームアップさせる



本誌の2月号では画像をズームアップさせるスクリプトを紹介した。そのスクリプトは画像しかズームアップできず、しかもIE 4以上のブラウザでしか動作しなかった。そこで今回はテキストをズームアップするダイナミックHTMLを紹介しよう。このコーナーの読者なら何度も悩まされたことと思うが、ダイナミックHTMLを使うと特定のブラウザでしか動作しなくなる。しかし今回紹介するTIPSは、2月号や3月号で紹介したように、互換性のないスクリプトの部分をブラウザ別に処理を分けているので、IEだけでなくナビゲーター4でも動作する。それではさっそくこのサンプルを見てみよう。



```

1 <SCRIPT LANGUAGE=" JavaScript">
  var NN = 0, IE = 0, i = 0;
  var Ver = navigator.appVersion.charAt(0);
  if (navigator.appName=="Netscape") NN = 1;
  if (navigator.appName=="Microsoft Internet Explorer")
    IE = 1;

  function msgZoom() {
    i++;
    if (i > 60) { clearTimeout(timer); return; }
    msg = '<DIV STYLE="font-size:' + i +
      'pt">HTML<BR>TIPS&TRICKS</DIV>';

    if (NN && Ver >= 4) {
      document.layers["lay"].document.write(msg);
      document.layers["lay"].document.close();
    }
    else if (IE && Ver >= 4) {
      document.all("lay").innerHTML = msg;
    }
    timer = setTimeout("msgZoom()",10);
  }
</SCRIPT>

2 <BODY onLoad="msgZoom()">
  <DIV ID="lay" STYLE="position:absolute"></DIV>

```

Point

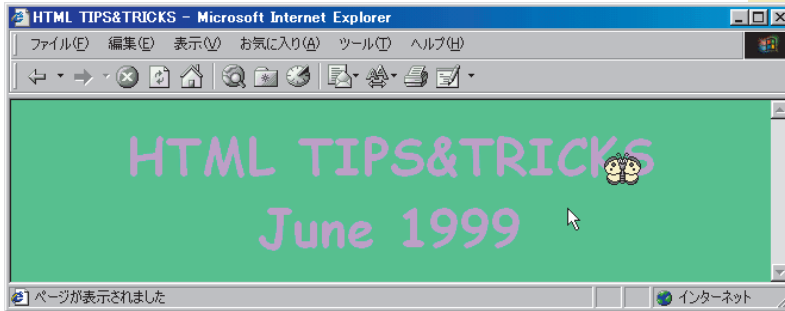
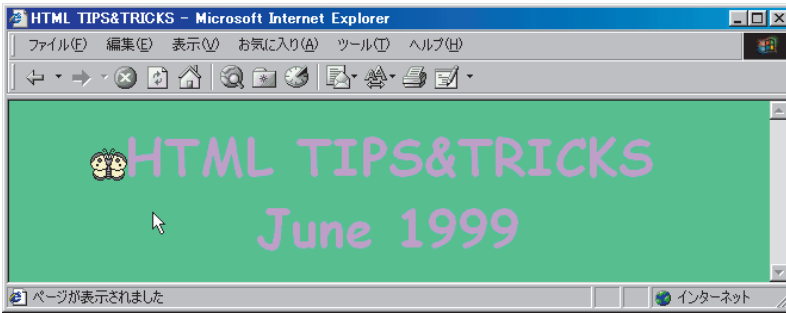
ソース①から説明しよう。まず変数「NN」、「IE」、「i」を宣言して初期値をすべて0にしている。変数はそれぞれ、「NN」はナビゲーター、「IE」はインターネットエクスプローラを表し、「i」はフォントサイズを表す。次にブラウザのバージョンを取得し、その値を変数「Ver」に入れている。「charAt(0)」ではバージョンを表す文字列の先頭の1文字だけを取得している。つまりバージョンが「4.01」の場合でも、小数点以下は無視される。このスクリプトはマイナーバージョン(小数点以下の数値)に関係なく動作するので小数点以下は無視してかまわない。その次の2つのif文ではnavigatorオブジェクトから得られたブラウザ名によって変数「NN」と「IE」の値を変える。ブラウザがナビゲーターなら「NN」の値を1にし、インターネットエクスプローラなら「IE」の値を1にしている。

実際にズームアップ処理を行う関数「msgZoom」を見てみよう。「i++」で「msgZoom」が呼び出されるたびにフォントサイズを1増やしている。Aのif文ではフォントサイズの最大値を60ptとして、「clearTimeout」で処理をストップさせる。Bはズームアップさせるテキストの指定で、ここではスタイルシートを使ってフォントサイズを変数「i」の値にしている。これで「msgZoom」が呼び出されるたびにフォントサイズが大きくなるのだ。Cのif文とDのelse if文はブラウザ別の処理だ。ナビゲーターの場合は「innerHTML」が使えないので、レイヤーの中にテキストを出力するようにしている。通常は「write」だけで出力するのだが、今回のようにレイヤーの中にテキストを出力する場合は、「close」を呼ぶことを忘れないようにしよう。インターネットエクスプローラでは「innerHTML」を使えるので、シンプルなスクリプトで済

む。ちなみに、「innerHTML」とはタグに含まれるすべてのHTMLを指定するもので、開始タグ、終了タグ、その中に含まれるテキストを意味する。最後はタイマー処理で、「setTimeout」のかわりに指定した時間(1000分の1秒単位)ごとに関数「msgZoom」を呼び出すようにする。

ソース②の説明をしよう。まず、<BODY>タグの「onLoad」イベントでページが読み込まれたときに関数「msgZoom」を呼び出す。次の<DIV>タグは中身が空だが、これはズームアップさせる位置を示しており、「ID」属性がそのオブジェクト名になる。同時にスタイルシートで「position」を指定する。ここで「position」を指定しないと、ズームアップさせる位置をブラウザが認識できなくなってしまうので必ず指定しよう。

マウスカーソルを追いかけるアイコン



右のサンプルを見てほしい。ページの上でマウスカーソルを動かすと、蝶のアイコンがマウスに合わせて動き出す。マウスを右に動かすと蝶がカーソルの左に、左に動かすとカーソルの右に少し離れてついていき、まるでマウスカーソルを追いかけているように見える。いったいどうやっているのだろうか。少々計算は必要だが、スクリプトとしてはそれほど複雑なものではない。サンプルのソースをコピーして画像を入れ替えば、好きなキャラクターを自分のホームページで動かすことも簡単だ。実用性はあまりないが、こうしたしかけを取り入れれば、ホームページの人气がぐっと高まることは間違いなしだ。

1

```
<SCRIPT>
var x2 = 0, y2 = 0;
var a = 48, b = 10;
function MoveImg () {
  x1 = event.clientX; y1 = event.clientY;
  len = Math.sqrt (Math.pow (x2 - x1, 2) +
    Math.pow (y2 - y1, 2));
  if (len > b) {
    icon.style.pixelLeft =
      Math.round (a/len*(x2-x1)+x1)-15;
```

```
icon.style.pixelTop =
  Math.round (a/len*(y2-y1)+y1)-13;
  x2 = x1; y2 = y1;
}
</SCRIPT>
```

2

```
<BODY onMouseMove="MoveImg ();">
<IMG ID="icon" SRC="TIPS4.GIF"
  STYLE="position: absolute; left: 0; top: 0;">
```

POINT

まずはアイコン用の画像を用意して、ソース②のタグで画像ファイルを指定する。スクリプトで操作できるように、タグにはID属性を付けて「icon」という名前にする。またスタイルシートの「position」に「absolute」を指定して、スクリプトで位置を移動できるようにする。「left」と「top」にはそれぞれ適当初期位置を指定しておく。ソース②の<BODY>タグでは「onMouseMove」イベントに関数「MoveImg」を指定して、ページ上でマウスを動かしたときに関数が呼ばれるようにする。

それではソース①のスクリプトを見てみよう。最初に宣言している変数「x2」と「y2」は、アイコンの以前の位置のX座標とY座標を保存するものだ。変数「a」と「b」の役目はあとで説明するが、この2つの変数の値は自分の好みに合うように変えて調整

してほしい。

関数「MoveImg」はページ上でマウスを動かすたびに呼ばれる。関数が呼ばれたときのマウスの座標は、IEでは「event.clientX」にX座標が、「event.clientY」にY座標が入っている。スクリプトを書きやすいようにそれぞれを変数「x1」と「y1」に入れておく。関数の終わりで「x1」と「y1」をそれぞれ「x2」と「y2」に保存すれば、「x1」と「y1」はマウスの現在位置になり、「x2」と「y2」はマウスの以前の位置になるわけだ。マウスの現在位置と以前の位置の間に直線を引き、直線上でマウスの現在位置から決まった長さ（変数「a = 48」）だけ離れた位置にアイコンを移動させれば、マウスを動かすたびにアイコンが追いかけるような効果を出せる。それには少々数学を使わなければならないが、数学が苦手な人は関数「MoveImg」をまる写しにすればいい。

ここで使っているJavaScriptの数学メソッドの機能だけを説明すると、「sqrt」は平方根を、「pow」は累乗を計算し、「round」は小数を一番近い整数にするものだ。

マウスを動かすたびに必ずアイコンを移動させるとちらつきが多くなるので、サンプルでは決まった距離よりも長くマウスが動いたときだけアイコンを動かすことにした。関数「MoveImg」の中の変数「len」は、以前の位置と現在の位置の距離を計算した結果だ。「len」が「b」の値より大きいときだけつまりサンプルでは10より長いときだけアイコンを移動させる。

あとはこの連載で何度か紹介したように、「style.pixelLeft」と「style.pixelTop」にそれぞれX座標とY座標を入れてアイコンの位置を移動させれば、アイコンがマウスを追いかけるスクリプトの完成だ。

メディアプレイヤーを制御する その2



前回紹介したように、IE 5に付属している新しいメディアプレイヤーをActiveXコントロールとしてウェブページに埋め込めば、さまざまな形式の音声や動画を再生できる。メディアプレイヤーには表示可能ないくつものパネルが用意されており、パラメータの指定でパネルの表示をオンにすれば、訪れた人が再生や停止、音量などを操作できるようになる。しかし、メディアプレイヤーにもともと付いている操作パネルは味気ないし、自分のページのデザインとは合わないと思う人もいるだろう。そういう人のために、画像ファイルとスクリプトを使ってオリジナルのコントローラーを作る方法を紹介しよう。



1

```
<A HREF="javascript:play()"><IMG BORDER=0 SRC="play.gif"
NAME="pbutton"></A>
<OBJECT ID="WMPlay"
CLASSID="CLSID:22D6F312-B0F6-11D0-94AB-0080C74C7E95">
<PARAM NAME="FileName" VALUE="media.mpg">
<PARAM NAME="ShowControls" VALUE="False">
<PARAM NAME="ShowTracker" VALUE="False">
<PARAM NAME="ShowDisplay" VALUE="False">
<PARAM NAME="ShowGotoBar" VALUE="False">
<PARAM NAME="ShowStatusBar" VALUE="False">
</OBJECT>
<A HREF="javascript:stop()"><IMG BORDER=0 SRC="stop.gif"
NAME="sbutton"></A>
```

2

```
<SCRIPT LANGUAGE="JavaScript">
function play () {
    WMPlay.CurrentPosition = 0;
    WMPlay.play();
}
function stop () {
    WMPlay.stop();
}
</SCRIPT>
```

POINT

サンプルとして、枠のない動画の左右にボタン画像を配置して、ボタンをクリックすると動画をスタートさせたりストップさせたりできるスクリプトを紹介しよう。

まずソース①の<OBJECT>タグだが、CLASSID属性で指定しているメディアプレイヤーのクラスIDやファイル名の指定は前回と同じだ。違う点は、操作パネルや情報パネルを消すために「Show……」のパラメータが「False」になっていること、ID属性で「WMPlayer」という名前が振られていることだ。ID属性はJavaScriptからアクセスするときのオブジェクト名として使われる。次に<OBJECT>タグの前後の2つのタグを見てみよう。<A>タグで囲んで画像をクリックすると、それぞれ「play」と「stop」というJavaScriptの関数が呼ばれ、再生と停止を行うようになってい

る。この2つの関数はHTMLファイルの末尾で定義することになる。

ソース②の「play」と「stop」という関数の中身を見てみよう。それぞれ「WMPlay」オブジェクトのプロパティを変えたりメソッドを呼んだりすることで、メディアプレイヤーを操作している。「play」関数では、「WMPlay.play()」メソッドを呼んで動画を再生させる。その前に「CurrentPosition」プロパティに0を設定して動画の再生位置を先頭に戻しているのは、再生位置が最後にある場合だと再生させてもすぐに停止してしまうためだ。「stop」関数は、「WMPlay.stop()」メソッドを呼んで動画を停止させるだけの簡単なものだ。

これだけのスクリプトで、左の画像をクリックする

と動画がスタートし、右の画像をクリックするとストップする機能を持ったあなたのオリジナルプレイヤーができていく。もっとカッコよくしたいなら、マウスを画像ボタンに載せたときやクリックしたときに画像を変化させてみるのもいいだろう。

たとえば、マウスを再生ボタンの上に載せたときに画像を入れ替えて変化させるスクリプトは以下のよう<A>タグの中に書けばいい。

```
<A HREF="javascript:play()"
onMouseOver="pbutton.src='play2.gif'"
onMouseOut="pbutton.src='play.gif'">
```

マウスを載せたとき(onMouseOver)と離れたとき(onMouseOut)のイベントで、画像の「src」プロパティに別のファイル名を指定すればいい。

HTMLパズルに挑戦しよう

隠されたトリックを解き明かせ！



今月のテーマ

・IE 5の新機能を制する

IE 5には数々の新機能が用意されているが、その中の1つが「ダイナミックプロパティ」だ。文字や画像の位置やサイズをダイナミックに変化させたい場合に、スクリプトとスタイルシートを組み合わせ、簡単な記述で大きな効果を出せるものだ。新しい機能をいち早く取り入れて、あっといわせるページを作ってみたいという人は、IE 5の新機能ダイナミックプロパティに挑戦してみよう。トリックがわかったらすぐに解答を送ってほしい。正解者には抽選で1名にオリジナル折りたたみ傘をプレゼントさせていただく。なお、正解は来月のこのコーナーで発表する。それでは頭をやわらかくして、今月のテーマ“IE 5の新機能を制する”にチャレンジ！

「HTMLパズルに挑戦しよう」

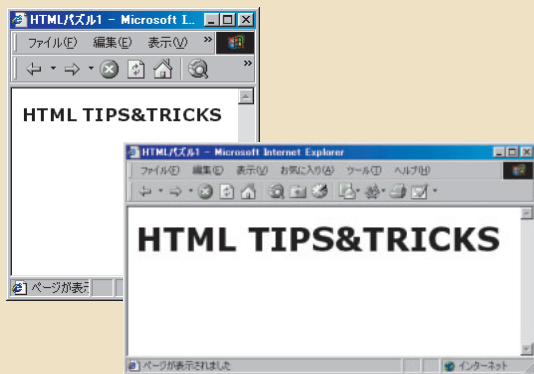
宛先

正解がわかった人も、わからなかった人も、ご意見、ご感想など何でもOK、次の宛先にメールしよう。用件の欄には必ず
HTML TIPS & TRICKS
の1行を忘れずに。あなたの挑戦を待つ！

✉ ip-cdrom@impress.co.jp

なお、締め切りは5月10日とさせていただきます。

QUESTION 1 文字のサイズを自動的に変える！

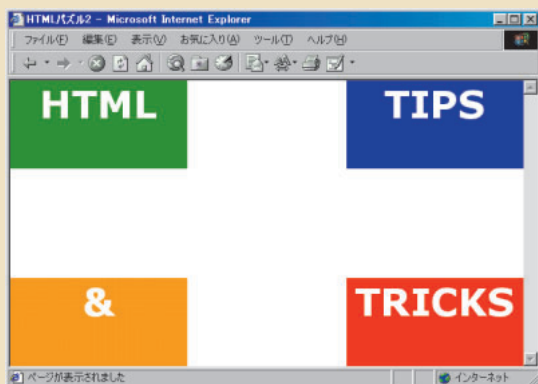


1問目は、ダイナミックプロパティの基本的な使い方だ。左のサンプルでは、ユーザーがブラウザのウィンドウサイズを変更すると、文字のサイズが自動的に変わる。ブラウザのサイズを大きくしたり小さくしたりすると、文字のサイズがブラウザの横幅に合わせて大きくなったり小さくなったりする。このテクニックがわかれば、訪れた人のディスプレイの解像度に合わせて自動的にデザインが変更されるようなページが作成できる。IE 4の場合のようにonResizeイベントが発生したときにスタイルシートのプロパティを変化させるスクリプトを書くのではなく、IE 5の新機能を使ってコンパクトにまとめてほしい。



<http://msdn.microsoft.com/workshop/author/dhtml/overview/recalc.asp>

QUESTION 2 四隅にブロックを置け！



2問目はダイナミックプロパティの応用問題だ。左のサンプルでは、ブラウザの四隅に色付きのブロックが置かれている。ブラウザのウィンドウサイズをどのように変更しても、各ブロックは必ずページの四隅に配置される。こんなデザインを使えば、柔軟で斬新な構成のページが作成できる。もちろんテーブルを使っているのではない。IE 4までのダイナミックHTMLに加えて、IE 5の新機能を使いこなす必要がある。1問目と同じように、onResizeイベントとオブジェクトの位置を変更するスクリプトを組み合わせるのではなく、ダイナミックプロパティを使ってなるべく短い記述で済ませてほしい。



ウィンドウの幅や高さから
ブロックの幅や高さを引けば



[インターネットマガジン バックナンバーアーカイブ] ご利用上の注意

このPDFファイルは、株式会社インプレスR&D(株式会社インプレスから分割)が1994年～2006年まで発行した月刊誌『インターネットマガジン』の誌面をPDF化し、「インターネットマガジン バックナンバーアーカイブ」として以下のウェブサイト「All-in-One INTERNET magazine 2.0」で公開しているものです。

<http://i.impressRD.jp/bn>

このファイルをご利用いただくにあたり、下記の注意事項を必ずお読みください。

- 記載されている内容(技術解説、URL、団体・企業名、商品名、価格、プレゼント募集、アンケートなど)は発行当時のものです。
- 収録されている内容は著作権法上の保護を受けています。著作権はそれぞれの記事の著作者(執筆者、写真の撮影者、イラストの作成者、編集部など)が保持しています。
- 著作者から許諾が得られなかった著作物は収録されていない場合があります。
- このファイルやその内容を改変したり、商用を目的として再利用することはできません。あくまで個人や企業の非商用利用での閲覧、複製、送信に限られます。
- 収録されている内容を何らかの媒体に引用としてご利用する際は、出典として媒体名および月号、該当ページ番号、発行元(株式会社インプレス R&D)、コピーライトなどの情報をご明記ください。
- オリジナルの雑誌の発行時点では、株式会社インプレス R&D(当時は株式会社インプレス)と著作権者は内容が正確なものであるように最大限に努めましたが、すべての情報が完全に正確であることは保証できません。このファイルの内容に起因する直接のおよび間接的な損害に対して、一切の責任を負いません。お客様個人の責任においてご利用ください。

このファイルに関するお問い合わせ先

株式会社インプレスR&D

All-in-One INTERNET magazine 編集部

im-info@impress.co.jp