

HTML TIPS & TRICKS

第 36 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

今月の特集「冬休みのホームページグレードアップ大作戦」は読んでいただけただかな? この連載では味わえないCGIなどのさまざまなテクニックが満載だ。TIPS & TRICKSだけでは飽き足りない欲張りな人はぜひ挑戦してほしい。TIPS & TRICKSも負けてはいられない。今月はFlashを操作する2つのTIPSを中心として、この連載でしか味わえない最先端の技をお届けしよう。



CD-ROM収録先 A Magnavi Ip0002 Htmltips
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(12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



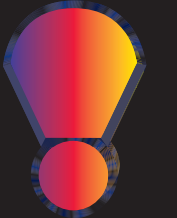
インターネットエクスプローラ5以上



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



1月号「HTMLパズルに挑戦しよう」の解答

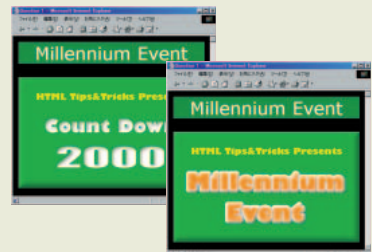
難易度が低い割には応募数が少なかったのは残念だ。時間を操作するテクニックはJavaScriptの基本なので、ぜひ使いこなせるようになっておこう。ここでは、解答のうちスクリプト部分だけを掲載しているが、付録CD-ROMにはHTMLのほかの部分も収録しているので参考にしてほしい。



ANSWER ① 2000年に画像を切り替える!

JavaScriptのDateオブジェクトを使って、現在時刻と2000年1月1日0時を比較し、その結果に応じて異なる画像を表示するだけなので、比較的簡単だ。

```
<SCRIPT LANGUAGE="JavaScript">
nowTime = new Date ();
Y2 = new Date (2000, 0, 1);
countDown = Y2.getTime () - nowTime.getTime ();
if (countDown > 0) document.write ("<IMG SRC='before.gif'>");
else document.write ("<IMG SRC='after.gif'>");
</SCRIPT>
```



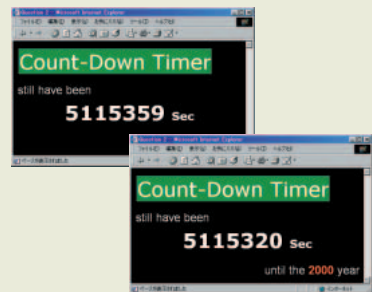
正解者: 正解者: 山口雅仁さん、村田正彦さん、よしともさん、Libretto 30さん、うおまさ@homeさん



ANSWER ② 2000年までカウントダウン!

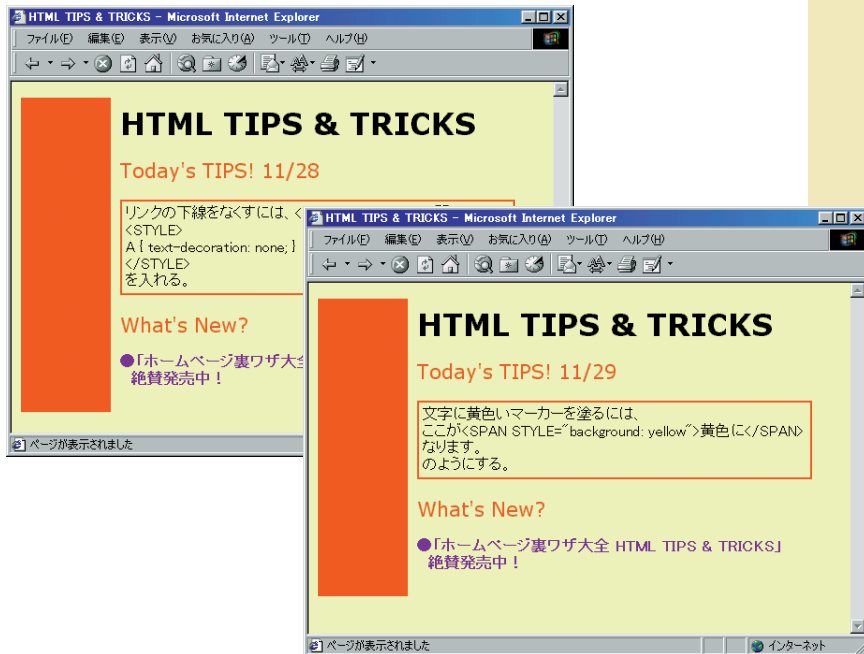
まず、1問目とほぼ同じ要領でカウントダウンまでの秒数を得ている。あとは空のレイヤーに文字列を表示する方法を知っていれば解ける。

```
<SCRIPT LANGUAGE="JavaScript">
function time () {
nowTime = new Date ();
Y2 = new Date (2000, 0, 1);
countDown = (Y2.getTime () - nowTime.getTime ()) / 1000;
countDown = Math.ceil (countDown);
if (document.layers) { document.timer.document.open ();
document.timer.document.write (countDown);
document.timer.document.close (); }
else if (document.all) timer.innerHTML = countDown;
}
</SCRIPT>
<BODY onLoad="setInterval ('time ()', 1000)">
<DIV ID="timer" STYLE="position: absolute;"></DIV>
```



正解者: 山口雅仁さん、村田正彦さん、よしともさん、Libretto 30さん、うおまさ@homeさん

日替わりでメッセージを表示する



今月は初めてIE 5でサポートされている変わった機能を利用して、毎日アクセスするたびに違ったメッセージをページ上に表示させる方法を紹介しよう。このサンプルには「Today's TIPS」コーナーがあり、訪れたユーザーに毎日1つずつちょっとしたHTMLテクニックを教えてくれる。JavaScriptに慣れた人なら、日付を判定してdocument.writeで書き込めばいいと思うかもしれないが、それではHTMLファイルの中に366個の文字列を書かなければならない。それではページを編集するのがたいへんな作業になってしまう。IE 5なら、用意したメッセージを簡単にページの中に埋め込む方法があるのだ。



1

```
<STYLE TYPE="text/css">
#msg { behavior:url (#default#download); }
</STYLE>
```

2

```
<BODY onLoad="DownloadStart ()">
<P ID="msg"></P>
```

3

```
<SCRIPT LANGUAGE="JavaScript">
var today = new Date ();
var m = today.getMonth () + 1; if (m < 10) m = "0" + m;
var d = today.getDate (); if (d < 10) d = "0" + d;
function DownloadStart () {
    msg.startDownload (m + " " + d + ".txt", DownloadDone);
}
function DownloadDone (s) {
    msg.innerHTML = s;
}
</SCRIPT>
```

POINT

このサンプルで表示される日替わりメッセージは、1日ずつ別々のテキストファイルに書かれている。1月1日なら「0101.txt」、11月23日なら「1123.txt」のように、日付をファイル名に付けて保存しておけば管理が楽だ。こうして作った366個のファイルをサーバー上にアップロードして、日付に合わせてその内容が自動的にページに埋め込まれるようにしたい。

考えられるのは、フローティングフレーム(<IFRAME>タグ)を埋め込み、JavaScriptでSRC属性を変更することだ。そうすればIE 4でも表示できるようになるが、ページの動作が重くなるし、サンプルのように読み込んだ内容がページと一体化しているような効果は出せない。

そこでIE 5の新機能の出番になる。IE 5では、外部にあるテキストファイルやHTMLファイルを読み

込んで、その内容をJavaScriptで処理できる機能が付いた。ソース①を見てみよう。これは「msg」というID名のタグにDHTMLビヘイビアを指定するスタイルシートだ。1999年12月号の「訪問した回数を記録する」と仕組みは同じだ。今回は「url(#default#download)」と指定して、タグにファイルのダウンロード機能を付けている。

ソース②の「msg」というID属性のある<P>タグがダウンロード機能を追加したタグで、ここにメッセージが表示される。ページが読み込まれたときに表示されるように、<BODY>タグのonLoadイベントで関数「DownloadStart」を呼び出す。

ソース③のスクリプトを見てみよう。まずDateオブジェクトを使って日付を取り出す。これはこの連載で何度も紹介したテクニックなので、説明する必

要はないだろう。

関数DownloadStartでは、DHTMLビヘイビアによってmsgオブジェクト(つまりID名「msg」の<P>タグ)に追加されたメソッド「startDownload」を呼び出す。1番目の引数はダウンロードするファイル名だ。ここでは日付からファイル名を組み立てている。2番目の引数はダウンロード終了後に呼び出される関数名だ。ファイルのダウンロードが終わると、関数「DownloadDone」が呼び出される。このときの引数「s」がファイルの内容になる。これをmsgオブジェクトのinnerHTMLプロパティに代入するだけでメッセージが表示される。

このテクニックが使えるれば、いろいろな応用が考えられる。「今日のできごと」や「今日の誕生花」を表示するページを作るのも難しくない。

Flashを操作する その1



1

```
<OBJECT
  CLASSID="clsid:D27CDB6E-AE6D-11cf-96B8-444553540000"
  WIDTH="300" HEIGHT="300" ID="sapporo_map">
<PARAM NAME="MOVIE" VALUE="sapporo.swf">
<PARAM NAME="QUALITY" VALUE="autohigh">
<EMBED SRC="sapporo.swf" WIDTH="300" HEIGHT="300"
  NAME="sapporo_map" QUALITY="autohigh" SWLIVECONNECT="true">
</EMBED>
</OBJECT>

<FORM>
<INPUT TYPE="button" VALUE="Zoom-In" onClick="swfZoom(50)">
<INPUT TYPE="button" VALUE="Zoom-Out" onClick="swfZoom(200)">
</FORM>
```

2

```
<SCRIPT LANGUAGE="JavaScript">
if (navigator.appName == "Netscape") { swflash = document; }
if (navigator.appName == "Microsoft Internet Explorer") { swflash = window; }
function swfZoom (n) {
  swflash["sapporo_map"].Zoom(n);
}
</SCRIPT>
```

Point

Flashを使っている人にとってソース①の<OBJECT>タグ(ActiveXコントロールの指定)や<EMBED>タグ(プラグインの指定)は見慣れているだろう。手で書き写すのは面倒だろうから、付録CD-ROMに収録されているソースをコピーすればいい。なお、上記のソースではCODEBASE属性やPLUGINSOURCE属性は省略している。

<OBJECT>タグや<EMBED>タグに関しては特に説明はしないが、<EMBED>タグの中に普段は使うことのないSWLIVECONNECT属性を設定しているので、これについて説明しておく。SWLIVECONNECTは、ナビゲーターのJavaでFlashメディアを操作する際に設定する属性で、値は必ず「true」にする。ナビゲーターはJavaScriptでプラグインを操作するときにJavaを起動する。ナビゲーターの設定が「Javaを有効にする」

になっていなければ、このサンプルは動作しないので注意しよう。

ソース①の<FORM>タグはボタンのクリックで「swfZoom」関数を呼び出すための記述で、関数の引数には拡大や縮小の倍率を設定する。サンプルでは引数に50(拡大)と200(縮小)を設定している。拡大と縮小の値が逆のようだが、100を引数で割った値が拡大率になるので、これでよいのだ。

ソース②を見てみよう。ここでは最初にナビゲーターとIEを判別して、変数「swflash」の値を指定している。ナビゲーターなら「document」が、IEなら「window」が、変数swflashの値になる。Flashメディアの扱いがIEとナビゲーターでは違うために、このような手法をとっているのだ。Flash

メディアはナビゲーターではdocumentオブジェクトの、IEではwindowオブジェクトのプロパティとして扱われる。

このTIPSの核となる関数「swfZoom」では、swflashのかわりに「[]」の中に<OBJECT>タグや<EMBED>タグで指定したID属性やNAME属性の「sapporo_map」を入れて、ZoomメソッドでFlashメディアを拡大させたり縮小させたりする。「n」にはボタンのonClickイベントで渡された引数が入る。この引数の値に応じて拡大率や縮小率が決定される。

以上で今回のTIPSは完成だ。訪問者が右クリックでポップアップメニューを表示させる必要がないので、Flashの操作方法に慣れている人にとっても歓迎される機能と言えるだろう。



みなさんはFlashをご存じだろうか。名前は知らなくてもウェブを見ていけば、少なくとも一度はFlashで作成されたページを目にしているはずだ。「Flash」と言っても意味は2つあり、1つは再生するプレイヤーで、もう1つは作成するためのオーサリングツールだ。今回のTIPSでは、後者を使って作成したFlashメディアをJavaScriptを使って制御することに挑戦する。左のサンプルでは、「Zoom-In」ボタンと「Zoom-Out」ボタンを押してFlashメディアを拡大させたり縮小させたりしている。Flashメディアは右クリックメニューで操作するだけでなく、このような使い方もできるのだ。

Flashを操作する その2



1

```
<BODY onLoad="setTimeout('swfZoom(50)', 500);">
```

2

```
<A HREF="javascript:void(0)" onClick="swfPan(-50,-50)">
<IMG SRC="tips3a.gif" BORDER="0" ALT="左上に移動"></A>
```

3

```
<SCRIPT LANGUAGE="JavaScript">
if (navigator.appName == "Netscape") { swflash = document; }
if (navigator.appName == "Microsoft Internet Explorer") { swflash = window; }
function swfPan(x,y) {
  swflash ["sapporo_map"].Pan (x,y,0);
}
</SCRIPT>
```

Point

このTIPSも前ページと同じく、FlashメディアをJavaScriptで操作する。違うのは、前ページが拡大と縮小の操作だったのに対して、今度は表示領域の移動する点だ。ソース自体は前ページとほとんど同じなので、ここでは<OBJECT>タグと<ENBED>タグを省略してある。

まずはソース①だ。<BODY>タグにonLoadイベントを設定して、Flashメディアの表示を拡大させている。なぜあらかじめ拡大するのかというと、ソース③に「Pan」という命令があるが、これは移動させる領域がない場合は機能しないからだ。初期状態ではすべての領域が表示されて、移動させる領域がないので、矢印ボタンが機能しなくなってしまう。なお、onLoadでZoomを実行するとIEでは最大化表示になってしまうので、setTimeoutを使ってタイミングをずらしている。

ソース②の<A>タグを見てみよう。ここでは画像にリンクを設定している。リンク先のHREF属性はJavaScriptを使って無効にしておき、onClickイベントでソース③の関数「swfPan」を呼び出すようにしている。関数swfPanのかっこ内には2つの引数を設定し、1番目が横方向の移動量(ピクセル数)になり、2番目が縦方向の移動量になる。上記のソースでは左上に移動するためのボタンしか示していないが、8方向の移動を行っているので、実際には<A>タグは8個書かれている。それぞれの引数には左上に移動させたい場合は「-」(マイナス)を付けた数値を設定し、右下に移動させたい場合はそのままの数値を設定する。

ソース③の最初の2行は前ページと同じなので説明は省略。次に関数swfPanだがこれも「swflash ["sapporo_map"].」の部分は前ページを参照し

てもらおうとして、最後の「Pan」だけを説明する。カッコ内の「x」と「y」にはソース②の<A>タグで渡された引数が入る。これで移動するための矢印ボタンが押されるたびに、表示領域を移動させるメソッドPanが呼び出されるのだ。

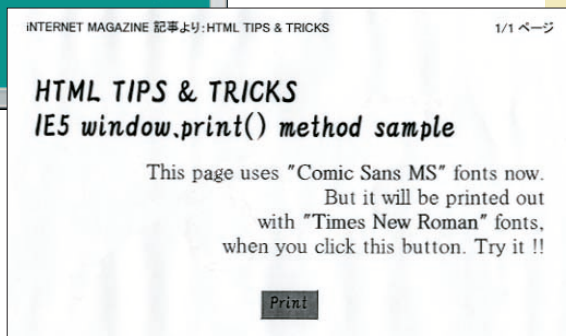
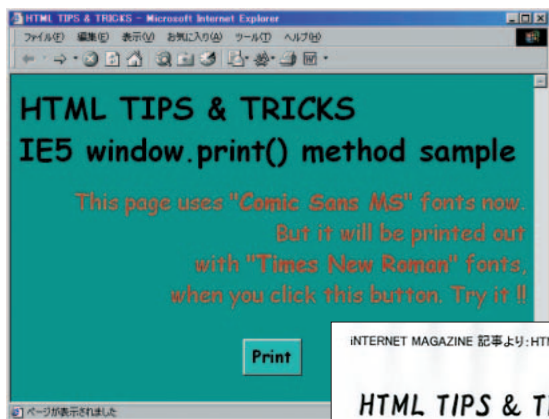
Flashメディアを拡大させておけば、マウスカーソルを上に乗せるだけで自動的に手のマークに変わり、ドラッグして自在に移動できるのだが、地図サイトのようにボタンで明確に移動させたい場合もあるだろう。そのようなときに使ってほしい技がこれだ。ぜひチャレンジしてみよう。

なお、今回紹介したFlashを操作する2つのサンプルは、いずれもマッキントッシュ版のIE 4.5では動作しないことに注意してほしい。



前ページではFlashメディアを拡大させたり縮小させたりするTIPSを紹介したが、単純に拡大するだけではその拡大した部分がどこにあたるのかがわかりにくい。そこで今度は拡大したあとに表示領域を移動するJavaScriptを紹介しよう。左のサンプルでは、地図の8方向に矢印がある。この矢印それぞれがFlashメディアの表示領域を移動させるボタンになっているのだ。サンプルのようにFlashを使って地図を作る際には、移動させるテクニックも知っておけば便利だ。拡大・縮小と移動のテクニックを併せて使って、自分の住んでいる街の地図をホームページで公開してみよう。

印刷機能を制御する



1

```
<SCRIPT LANGUAGE="JavaScript">
var originalTitle ;
function beforeprint () {
  originalTitle = document.title;
  document.title = "INTERNET MAGAZINE 記事より : "
    + document.title;
  document.all.t1.style.fontFamily = "Times New Roman";
}
function afterprint () {
  document.title = originalTitle;
  document.all.t1.style.fontFamily = "Comic Sans MS";
}
</SCRIPT>
```

2

```
<BODY onbeforeprint="beforeprint()"
  onafterprint="afterprint()">
```

3

```
<FORM>
  <input type="button" value="印刷" onClick="window.print ();">
</FORM>
```

Point

初めにページに「Print」ボタンを付けてみよう。いつもフォームで使っているようなボタンをクリックすると、ブラウザのメニューから「印刷」を選んだ場合と同じように印刷できるようにしてみる。ソース②を見ればわかるとおり、これはとても簡単だ。IE5で追加された新しいJavaScriptのメソッド「window.print()」をボタンのonClickイベントに組み込めばいいだけだ。これだけで、Visual Basicで作られた普通のアプリケーションと同様に、わざわざメニューをたどらなくてもクリック1つで印刷できるようにする。

さてこれだけでは面白くないので、もう1つちょっとした仕掛けを組み込んでみよう。ソース②の<BODY>タグを見てほしい。ここに「onbeforeprint」と「onafterprint」という2つのイベントが指定されているのわかるだろうか。これはwindow.print()

と同じくIE5で新たに追加されたイベントで、印刷の実行前と実行後に何らかの仕掛けを動かすことができるようにするものだ。たとえば印刷前にタイトルをわかりやすく変更し、印刷が終わったらサイト全体のポリシーに合わせたものに変更する。または、印刷前にフォントや背景色を変えて、印刷終了後には元に戻す、というようなサービスが可能になる。

組み込みたい仕掛けは、ソース①の「beforeprint」と「afterprint」のようにJavaScriptの関数としてあらかじめ定義しておく。ソース②の<BODY>タグのように、onbeforeprint イベントと onafterprint イベントが発生したとき、つまり印刷前と印刷後にこの2つの関数がそれぞれ呼び出されるようにしている。

2つの関数の内容はこの連載の読者ならひと目で

レストラン紹介サイトの地図やクーポン、気になる新聞記事など、ウェブページを印刷する機会は意外と多い。そんなとき、普通はブラウザのメニューから「ファイル」「印刷」を選んでプリントアウトする。しかしこれではメニューバーを隠したウィンドウの場合や、まだブラウザに不慣れなユーザーの場合には不便なこともあるだろう。そこでIE5では新しくJavaScriptを使って印刷をサポートする機能が付いた。左のページにある「Print」ボタンをクリックすると、メニューを選んだときとまったく同じように「印刷」ウィンドウが開き、印刷することができる。今回はこの機能をマスターしよう。

わかるような簡単なものだ。beforeprintでは、変数「originalTitle」に文書のタイトル(document.title)を保存してから、タイトルの前に「INTERNET MAGAZINE 記事より:」という文字列を追加している。また、ページの一部のフォントを「Times New Roman」に変えている。afterprintでは、変更した文書のタイトルを元に戻し、フォントを「Comic Sans MS」に戻している。これで、印刷されたときに便利のように一時的にページのスタイルを変更し、印刷が終わったら元のページに戻すという仕掛けができる。

このサンプルは新機能をチェックしてみるだけの簡単なものだが、ブラウザで見たときと印刷したときとで表示を細かく調整するスクリプトを書けば、実用的なページもできるだろう。

HTMLパズルに挑戦しよう

隠されたトリックを解き明かせ！



大好評発売中！



今月のテーマ

・新年のあいさつを制する

最近では電子メールで年賀状を出したり、元旦にホームページを更新して新年のあいさつを表示させたりしている人も多いのではないだろうか。そこで今回は、スクリプトを使ってページ上で自動的に「あけましておめでとうございます」などとあいさつするパズルに挑戦していただく。これがわかれば、年賀状を出しそこねた人も安心だ。大晦日までに問題を解いて2000年の元旦を迎えよう。トリックがわかったらすぐに解答を送ってきてほしい。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“新年のあいさつを制する”にチャレンジ！

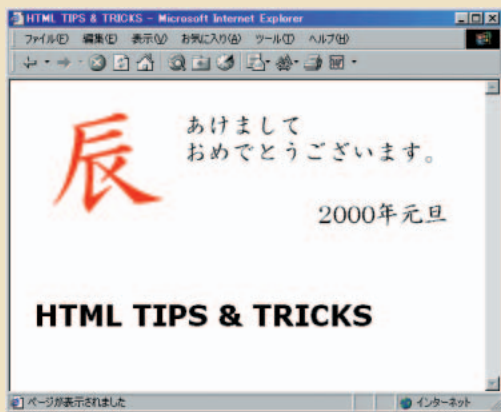
「HTMLパズルに挑戦しよう」宛先

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！

✉ im-html@impress.co.jp

なお、締め切りは1月10日とさせていただきます。

QUESTION 1 干支を計算せよ！

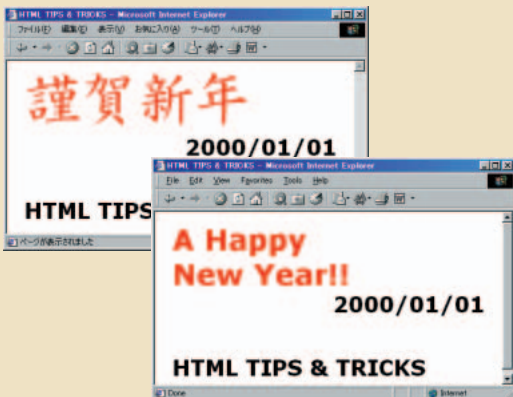


筆不精な人にとっては、毎年新年賀状を書くのは何とも面倒なことだ。そこで1回作れば毎年使えるようになる年賀状をホームページに表示させてみよう。左のサンプルは、まったく同じスクリプトを使って毎年元旦にアクセスしたときだけに新年のあいさつを表示させるページだ。その年が何年かを表示させるだけでなく、干支を自動的に計算して2000年なら「辰」、2001年なら「巳」という文字を表示させる機能も付いている。解答はIE3やナビゲーター3でも表示できるものとする。正解とは関係はないが、マイクロソフトオフィスなどに付属している和文フォントを持っているなら、雰囲気のある解答を作してほしい。



最初の子年は西暦4年……

QUESTION 2 英語と日本語であいさつせよ！



左のサンプルは第1問と同じく、毎年元旦だけに新年のあいさつを表示させるページだ。ただし、日本語のブラウザでアクセスしたときと英語のブラウザでアクセスしたときで表示が変わるようにしたい。これがわかれば新年のあいさつだけでなく、訪れた人の言語によって自動的に各国語のページにジャンプするなどの応用が利く。正解はシンプルなものだがIEとナビゲーターでやり方が違うことに注意。IE5では、「ツール」メニュー「インターネットオプション」「言語」で「変更」ボタンを押して英語表示を追加すればテストができる。なお、解答は英語でなくても日本語以外の外国語であればOKだ。



navigatorオブジェクトのプロパティを調べよう……



[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