

# HTML TIPS & TRICKS

第 11 回

## 誰よりも早く 最新のHTMLを使ってみよう

藤井幸孝 白畑真

動きのあるウェブページや思いのままのレイアウトのウェブページを作る。そのために、ショックウェブやJava、アクロバットなどを使えば、それはそれで表現力のあるものが作れる。しかしナビゲーター4.0やようやく正式リリースされたエクスプローラ4.0を使えば、ちょっとしたJavaScriptやHTMLでも十分インパクトのあるものが作れる。今月は、そんな表現力豊かなお手軽TIPSとTRICKSをお届けする。

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(10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3.0以上



インターネットエクスプローラ4.0以上



ネットスケープナビゲーター3.0以上



ネットスケープナビゲーター4.0以上



### 11月号「HTMLパズルに挑戦しよう」の解答

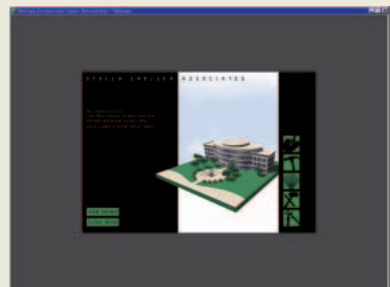
解答を送ってくれた人の解答例を見ると、いかにJavaScriptに精通しているかがよくわかる。目新しいウェブページを作るのは、もはやテクニックの問題だけなのかもしれない。というわけで、このコーナーではこれからも読者の知らないような新しいテクニックを紹介していくつもりだ。それでは「スクリーンを制する」の解答を発表しよう。



#### ANSWER ① キオスクウィンドウを出現させる!

比較的簡単だったのではないだろうか。ポイントはキオスクウィンドウの幅と高さを設定してやることだ。これさえ指定すれば、ツールバーなど余計なものを表示しないウィンドウが開く。ここで「outerWidth」、「outerHeight」とあるのはウィンドウの枠まで含めたウィンドウサイズを指定する場合に用いるプロパティだ。覚えておくと便利だぞ。

```
<SCRIPT LANGUAGE="JavaScript1.2">
function myKiosk() {
  window.open('kiosksrc.html',
    'kiosk_window','outerWidth=キオスクの幅,outerHeight=キオスクの高さ,
    top=0, left=0, alwaysRaised=yes');
}
</SCRIPT>
<A HREF="javascript://" onClick="myKiosk()">
```



正解者の方々：石山フミオさん、川田哲さん、米澤慎一郎さん、ほか



#### ANSWER ② フルスクリーンで表示させる!

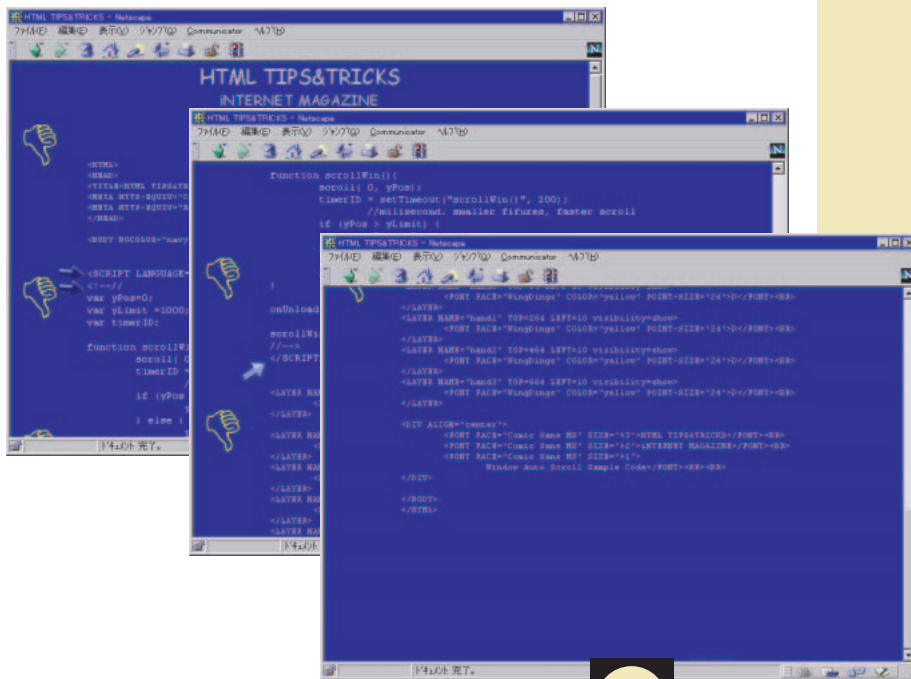
残念ながら2問目は知識を問うような問題だったためか正解者が少なかった。フルスクリーンにするにはウィンドウの幅と高さに、数値ではなく表示するディスプレイ解像度の値を自動的に判別するプロパティ「+screen.width+」と「+screen.height+」を与えればよい。これを使えば、画面の半分や4分の1のウィンドウも作れるぞ。

```
<SCRIPT LANGUAGE="JavaScript1.2">
function myFullscreen() {
  window.open('kiosksrc.html','kiosk_window',
    'width='+screen.width+',height='+screen.height+',top=0,left=0');
}
</SCRIPT>
<A HREF="javascript://" onClick="myFullscreen()">
```



正解者の方々：石山フミオさん、ほか

# 画面を自動スクロールさせる



```
<SCRIPT LANGUAGE="JavaScript">
var yPos=0;
var yLimit=1200;
var timerID;

function scrollWin(){
  scroll( 0, yPos);
  timerID = setTimeout("scrollWin()", 200);

  if (yPos > yLimit) {
    yPos = 0;
  } else {
    yPos +=3;
  }
}

onUnload =clearTimeout(timerID);
scrollWin();
</SCRIPT>
```



縦に長いページ。そんなページを見るとき普通ならスクロールバーをマウスで動かしたりあるいはカーソルキーで動かしたりして見る。このように手動で画面をスクロールさせるのではなく、映画のエンディングタイトルのように自動的にページがスクロールしだすと新しい表現方法が生まれるのではないだろうか。そこで今回の目玉TIPSとして、JavaScriptを使ったページの自動スクロールを紹介する。開かれると同時に上へ上へと流れていくページ。これはかなり衝撃だ。それにこれならカーソル操作が不要なので、電車の中で最新のブラウザ搭載のPDAを片手で持ってWWWブラウジングもできるぞ。

## Point

意外にも短いスクリプトなので驚いたのではないだろうか。実はウィンドウの表示位置をコントロールするプログラムはJavaScriptに最初から用意されているのだ。「scroll()」というがそれで、ネットスケープナビゲーター2.02以降で使用可能だ。ただしこの関数は画面を文字通りスクロールさせるというものではなく、ウィンドウの内容をどの位置から表示するかを指定するものだ。scroll()は2つのパラメータを取る。それぞれ、画面表示開始位置のX座標とY座標だ。「yPos」はscroll()で使うY座標の値だ。これを変化させることで自動スクロールを実現する。初期値として「yPos=0」とするのは、最初の表示位置のY座標を0にすることを意味している。yLimitはスクロールの限界値で、ページ全体の長さより少し大きい値(ピクセル単位)を入れておくのが適当だろう。「timerID」はタイマーを使う場合に必要

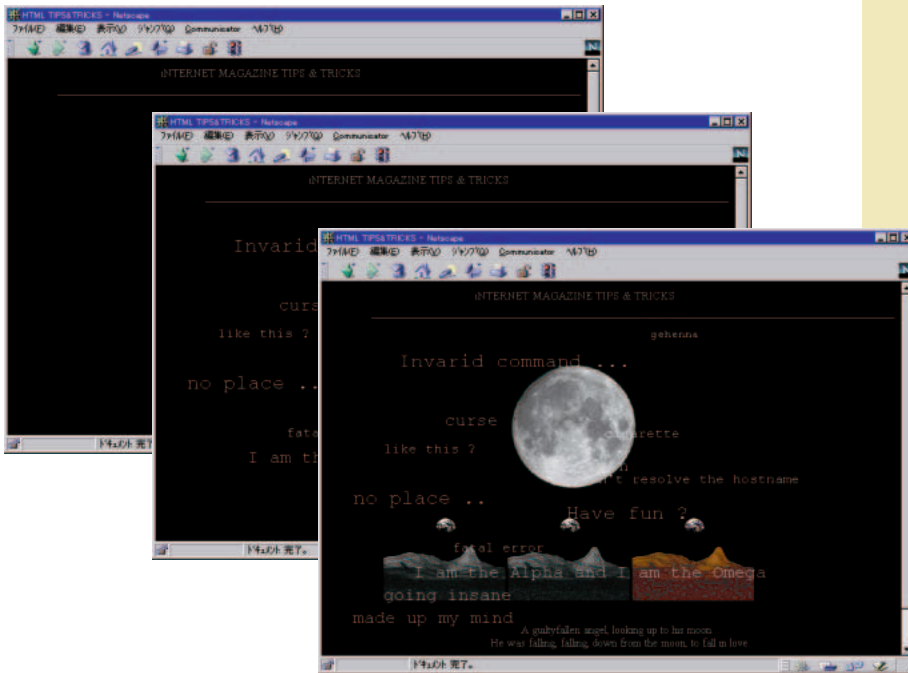
な変数なので定義しておく。「scrollWin()」はまず最初に、ページを、(0,yPos)つまり(0,0)から表示するようにscroll()を実行する。続いて「setTimeout()」を使い、scrollWin()自身を200ミリ秒後に再び実行するようにセットする。そして、yPosの値がスクロールの限界値であるyLimit以下の場合には、yPosに適当な値(ここでは3ピクセル)を加える。setTimeout()の効果で、200ミリ秒後には再度scrollWin()が実行される。ただしこのときyPosの値は3ピクセルずつ大きくなっている。これで、画面の表示は少し下から始まることになる。この繰り返しでウィンドウの中身が縦方向にスクロールして見えるという仕掛けだ。yPosの値がyLimitよりも大きくなると、yPosは0にリセットされ、画面は一番上に戻り、スクロールが最初からやり直される。このままだと永遠

に(ページが変わっても)スクロールが続いてしまうので、ページ移動の際にスクロールが止まるように、

```
onUnload = clearTimeout(timerID);
```

として、タイマーを解除しておく。なお、上のスクリプトは<BODY>内に<SCRIPT>として書いている。さてここまでできたら、スクリプト中の数字をいろいろ変えて、さまざまなスクロールを試してみよう。

## 文章を1行ずつ次々と表示させる



前述の「画面を自動スクロールさせる」で使った、ウェブページの状態を一定時間間隔で変化させるプログラム「setTimeout()」。これを使って、ショックウェブのように文字を1行ずつ表示させたり、画像を時間ごとに変化させたりするページを作ってみよう。ここで使用するの、ネットスケープナビゲーターのレイヤー機能だ。8月号で紹介した、アンカー上にマウスを持っていくと文字が浮き出てくるというレイヤーのTIPSを覚えているだろうか。レイヤーを使ってページに動きを与える方法と「setTimeout()」を組み合わせようというわけだ。まずは下の例を見ないで作ってみたいだろうか。そんなに難しくないはずだ。



```
<LAYER NAME="layer0" TOP=100 LEFT=128
visibility=hide>Invalid command ...</LAYER>
:
<LAYER NAME="layer18" TOP=240 LEFT=128
visibility=hide><IMG SRC=moongray.gif></LAYER>

<SCRIPT>
layerLimit = 18; timerID = 0; i = 0;
layer = new Array(19)
layer[0] = document.layers["layer0"];
:
layer[18] = document.layers["layer18"];

function tmdisplay() {
if (i <= layerLimit) {
clearTimeout(timerID);
timerID = setTimeout("tmdisplay()", 1000);
layer[i].visibility="show";
i = i+1;
} else {
i = 0;
}
}
tmdisplay();
</SCRIPT>
```

### POINT

ここでも<LAYER>と<SCRIPT>は<BODY>内に書く。まずタイマーで表示させたいテキストや画像を<LAYER>タグでくっつけておく。各レイヤーには名前を付けるが、「layer0、layer1、...、layer18」のように、0から始まる数字連番の付いた名前にしておこう。重要なのは数字の順番だ。表示させたい順に番号を振っておいてほしい。各レイヤーの位置は好きなところにする。<LAYER>タグの中の「TOP」でブラウザ表示画面の上からの位置を、「LEFT」でブラウザ表示画面の左からの位置をそれぞれ指定すればよい。また、「visibility」の値を「hide」にして、各レイヤーが最初は見えないようにしておこう。次にスクリプトだ。「layerLimit」は表示させるレイヤーの最後の番号すなわちここでは18となる。「timerID」はタイマーに必須の変数だ。「i」

はどのレイヤーを表示するかのカウンターになる。

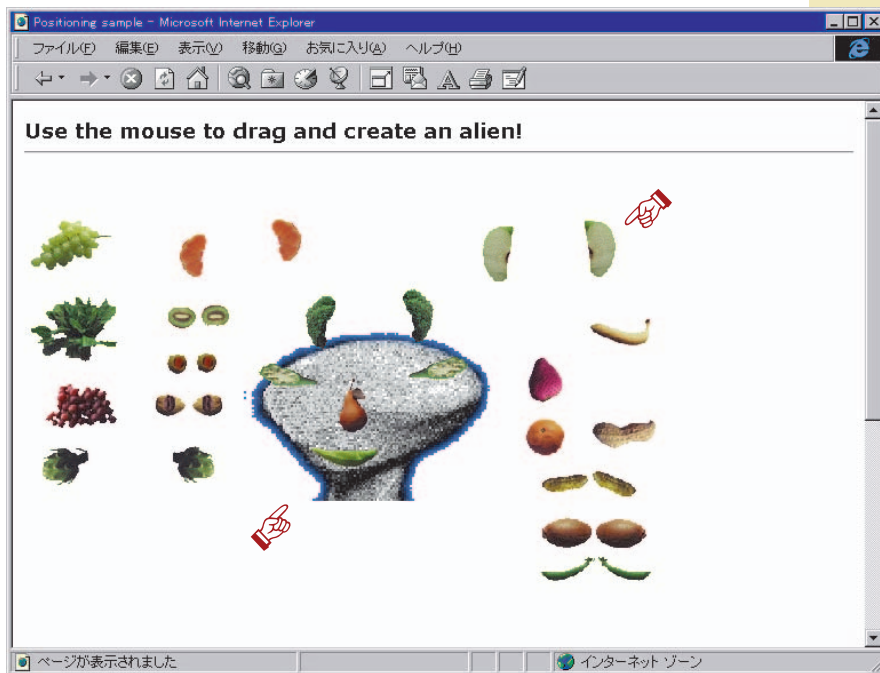
```
layerLimit = 18; timerID = 0; i = 0;
layer = new Array(19)
layer[0] = document.layers["layer0"];
: (省略)
layer[18] = document.layers["layer18"];
```

「layer」は、各レイヤーに対応する配列だ。JavaScriptの「Array()」を使って、レイヤーの個数分の配列を作る。この例ではレイヤーは19個(layer0からlayer18)あるのでArray(19)として配列を19個作った。配列の準備ができたなら、document.layers["layer0"]を変数layer[0]で参照できるように、対応関係を作る。同様にすべてのレイヤーの対応関係を作る。ここまでできたらメインのプログラム「tmdisplay()」

だ。この中で使われている「setTimeout()」については前述のTIPS「画面を自動スクロールさせる」を参考してほしい。setTimeout()で1000ミリ秒ごとにtmdisplay()を実行している。これによってlayer[i]の「visibility」を次々に「show」にし、各レイヤーを表示させる。これをレイヤーの数だけ繰り返す。setTimeout()の2つめのパラメータである数字を大きくすると表示間隔を長くできる。

このTIPSを使えば、文章がメインであるページを1行ずつ表示させていくこともできるし、画像とテキストをウィンドウ内にちりばめていくこともできる。読みやすいページ、映像を使ったページ。どう表現するかは使い次第だ。

## スタイルシートで配置を指定する



URL <http://www.microsoft.com/msdn/sdk/inetsdk/samples/dhtml/position/drag.htm>



左の画面を見てほしい。ここで表示されている画像は1つの画像ファイルで構成されているのではない。この画面に散らばる果物や野菜の画像は、1つ1つが別のものである。IE3.0までは、このような複雑な画像の配置をするために表組みを駆使しなければならなかった。しかし、ここでは<TABLE>タグは一切使っていない。ではどうやって実現しているのだろう。IE4.0のスタイルシート機能を使っているのだ。スタイルシートを使うと、このように自由な画像配置ができる。自由なページレイアウトはウェブページを作成する人にとって最も待ち望んでいた機能だろう。早速このTIPSを紹介しよう。



```
<IMG ID="image1"
  STYLE="position:absolute;
  TOP:20pt; LEFT:10pt; WIDTH:63pt;
  HEIGHT:38pt; Z-INDEX:1;"
  SRC="画像ファイル.gif">
```

### POINT

IE3.0ではスタイルシートを使用して文字の重ね合わせができた。IE4.0ではこれに加え、画像をウェブページの上端と左端からの距離を指定して配置できるようになった。またネットスケープナビゲーターのレイヤー機能のように、画像の上に画像を重ねることも可能だ。上記のIMGタグを順を追って説明しよう。

「ID="image1"」は、IMGタグで指定する画像に「image1」という名前を指定している。これはHTMLやJavaScriptなどから画像を参照するのに必要だ。

次に「STYLE」以下のオプションについて説明する。

「position」は画像などの配置の位置関係を指定するもので、「absolute」、「relative」、「static」のいずれかの値を取る。absoluteは、ウェブページでの絶対的な位置を指定する場合に使い、

relativeはある基準となる画像などからの相対的な位置を指定する場合に使う。staticは画像をどこに配置をするかを指定しない場合に使う。なお、何も指定しなかった場合はrelativeと同じになる。この例ではabsoluteを指定しているので、ウェブページでの絶対的な位置を指定していることになる。

「TOP」の値はウェブページの上端から配置する画像までの絶対距離だ。ここでは20ptすなわち上端から20ポイントの距離となる。

「LEFT」の値はウェブページの左端から配置する画像までの絶対距離だ。ここでは10ptすなわち左端から10ポイントの距離となる。

「WIDTH」と「HEIGHT」は表示させる画像の横の長さおよび縦の長さの値を指定する。指定しなかった場合にはブラウザが自動的に決定する。

「Z-INDEX」は同一ウェブページにある画像の重

ね合わせの順番の値を指定するものだ。この例の「1」という値は、画像が一番うしろに配置されていることを意味している。たとえば3つの画像が重なっているとすると、Z-INDEX:3が一番前に、Z-INDEX:1が一番うしろに表示されることになる。画像が重なっていれば当然一番うしろのものが隠れる。

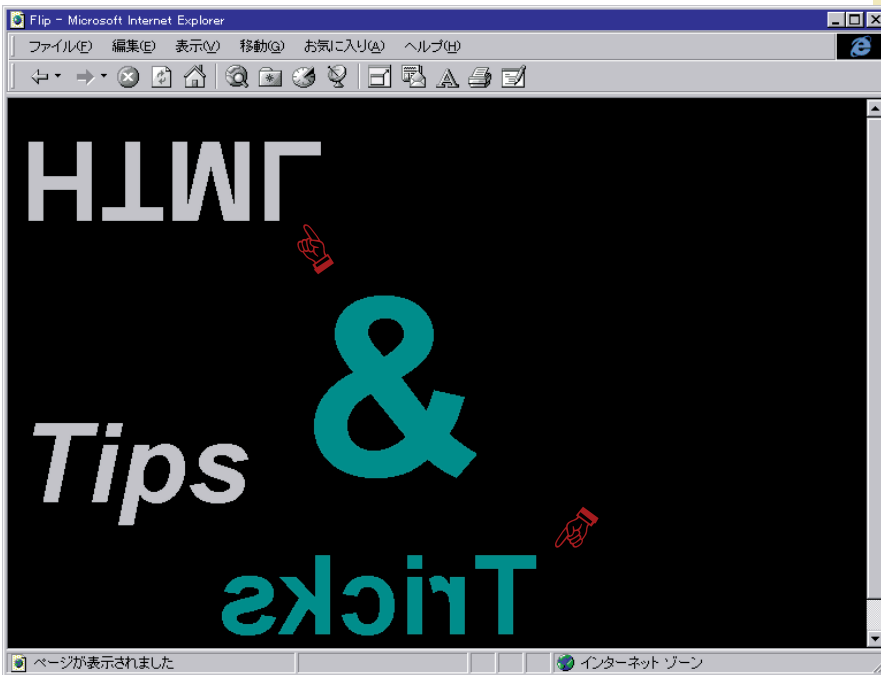
以上のようにスタイルシートを使えば画像の配置は自由自在だ。

最後にこのTIPSのを使った面白い例を紹介しよう。実は上の画面の個々の画像は、マウスでドラッグして画面上を自由に動かすことができるのだ。仕掛けはこのTIPSとJavaScriptとの組み合わせなのだが、うまく作ればこんなインタラクティブなしくみも可能だ。ぜひ挑戦してみたい。

## 文字を上下左右に反転させる



4.0



ウェブページを見ていて大胆な表示効果でないにもかかわらず、ちょっとしたことで目を引くようなものがある。ここで紹介するのはそんなTIPSだ。まずは左の画面を見てほしい。「HTML」と「Tricks」の文字列がそれぞれ上下と左右に反転しているのが分かるだろう。当然このページには1つたりとも画像は使用していないし、プラグインなども利用していない。ただのHTMLだけで構成されているのだ。どんどん複雑化していくHTMLだが、これは簡単に作成できる。しかも、上下左右に反転した見慣れない文字は、ページを見る側にインパクトを与えるはずだ。それでは、今月最後のクールなTIPS、文字の反転方法を紹介しよう。

URL <http://www.alles.or.jp/~shira/hlMiTsU/flip.html>



```
<SPAN STYLE="width: 480; height: 64; font-size: 96px; font-weight: bold; font-family: Arial; color: #C0C0C0; Filter: FlipV">HTML</SPAN>
```

```
<SPAN STYLE="width: 450; height: 32; font-size: 72pt; font-weight: bold; font-family: Arial; color: #008080; Filter: FlipH">Tricks</SPAN>
```

Point

これまでは反転させた文字を表示するためには画像を利用するしか方法がなかったが、今回紹介するTIPSを使えば、HTMLタグだけを使って簡単に文字を反転させることができるようになる。では、さっそく内容を見ていこう。

```
<SPAN STYLE="width: 480; height: 64; font-size: 96px; font-weight: bold; font-family: Arial; color: #C0C0C0; Filter: FlipV">HTML</SPAN>
```

一見、ごく普通のスタイルシート定義に見える<SPAN> ~ </SPAN>はそれ自体が意味を持つものではないが、部分的なスタイルシートの指定に使われるタグだ。注目してほしいのは、<SPAN>タグの中にある「Filter」というオプションである。

これはその名のとおり、<SPAN> ~ </SPAN>タグで囲まれた文字列にフィルターをかけるものだ。このフィルター機能を使って文字を上下に反転するには、「Filter: FlipV」と指定してやればよい。同様に、文字列「Tricks」の部分には「Filter: FlipH」というフィルターをかけて、文字列の左右を反転させている。

このように縦方向に文字を反転させる場合ならFlipVを、横方向に文字を反転させる場合にはFlipHを、Filterオプションで指定することによって文字を反転させることができる。

このTIPSを使うと、簡単にインパクトのあるページが作成できる。ただし、使いすぎると内容が見にくくなってしまふのでほどほどにしたほうが良いだろう。またインターネットエクスプローラ4.0以外のWWWブラウザを使用しているユーザーは、このTIPSによる表示効果を見られない。

このようなことを考慮してJavaScriptを使い、インターネットエクスプローラ4.0でアクセスしてきたユーザーのみこのTIPSを利用するようにすれば、ユーザーの環境を考えて細部までこだわった「軽い」ページを作成することができる。TIPSを組み合わせて使えば面白いページになること間違いなしだ。

# 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

### ・レイヤーを制する・

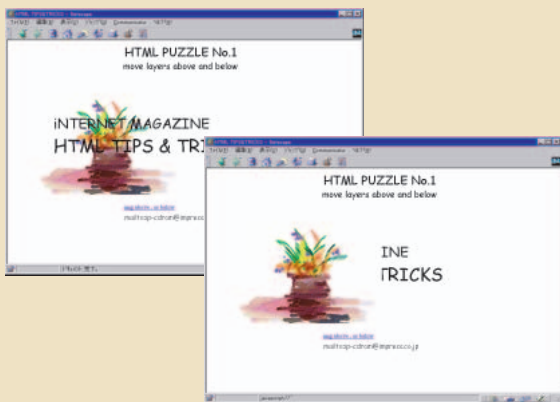
文字や画像を、ウェブページ上の思いどおりの場所に配置する。そんな自由なレイアウトを実現するのがネットスケープナビゲーター4.0に実装されているレイヤー機能だ。ウェブページ作成者はこの機能を使って自由な発想でページを作ることができる。しかし、この自由さゆえに作成者の技術やセンスが問われるのだ。そこで、今月はこの「レイヤー」に挑戦していただく。トリックが分かったらすぐに解答を送ってきてほしい。正解者にはダイレクトプラスを発行させていただく。なお、正解は来月のこのコーナーで発表する。それでは頭をやわらかくして、今月のテーマ“レイヤーを制する”にチャレンジ！

### 「HTMLパズルに挑戦しよう」 宛先

正解が分かった人も、分からなかった人も、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

### QUESTION 1 レイヤーを前面に表示させる！



レイヤーを使えば、画像と画像、文字と文字を重ねて表示することができる。このとき隠れてしまったうしろの文字や画像はどうなっているのだろうか。重ねること自体がページ全体のデザインではあるのだが、全部表示する手段はないのだろうか。そこで今月の第1問。ボタンを押すと重なった後ろ側のレイヤーが前に出てくる、というページを作ってもらいたい。それが簡単だというならば、一歩進んでアンカーの上にマウスを載せると裏の画像が表に出て、マウスを外すと再び後ろに隠れるというものに挑戦してほしい。このほうが便利そうだ。解答はどちらでも正解にしよう。



レイヤーの指定方法を思い出して.....

### QUESTION 2 レイヤーを使ってウェブページを表示させる！



IE3.0以上では、ページ内にページを表示させる機能としてフローティングフレームがある。これはウィンドウの真ん中に他のウェブページを表示することを可能にするものだ。しかし、この機能はまだネットスケープナビゲーターでは対応していない。WWWの特性を活かせる機能だけに残念だが、実をいうとレイヤーを利用することで似たようなことが実現できるのだ。ボタンやアンカーを使えば、フローティングフレームのようにレイヤーの中身を入れ替えることも可能だ。ページ内にレイヤーを1つ作って、お気に入りのページを表示する。そんな解答を送ってきてほしい。



アンカーのレイヤーとJavaScriptをうまく組み合わせれば.....



## [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](mailto:im-info@impress.co.jp)