

HTML TIPS & TRICKS

第 15 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

ついにマッキントッシュ対応のインターネットエクスプローラ4.0日本語版が出荷された。この製品によってマッキントッシュユーザーもスタイルシートやレイヤー、フローティングフレームの恩恵にあずかれるのだ。このコーナーではウィンドウズユーザーだけでなくマッキントッシュユーザーも楽しめるように紙面を充実させていくつもりだ。今月も高度なTIPSから誰にでもすぐできる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(3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



4月号「HTMLパズルに挑戦しよう」の解答

問題を難しく考えすぎると簡単なことを見落としてしまうことがある。今回はそんなパズルだったのではないだろうか。特に1問目は解答を見て驚く人がいるかもしれないほど簡単だったはずだ。ウェブページの制作はアイデア次第ということがわかる解答をさっそく発表しよう。

ANSWER ① 建物、地形の簡易画像を表示させる!

悩んだ人もいたかもしれないが、実は簡易画像とはフォントのことだ。ここで使われているフォントはE4.0と同時にインストールされる「webdings」というもので、画像として十分使える。

```
<FONT COLOR="red" FACE="webdings">B</FONT>
<FONT COLOR="blue" FACE="webdings">C</FONT>
<FONT COLOR="brown" FACE="webdings">D</FONT>
<FONT COLOR="gray" FACE="webdings">F</FONT>
<FONT COLOR="silver" FACE="webdings">G</FONT>
<FONT COLOR="yellow" FACE="webdings">H</FONT>
<FONT COLOR="green" FACE="webdings">S</FONT>
```

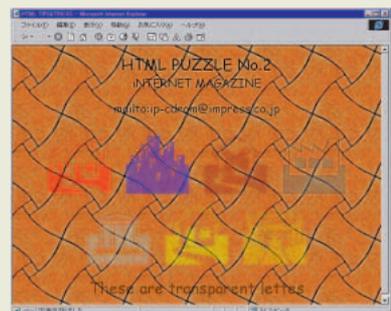


戸田順也さん、北出誠さん、たかすさん、ほか多数

ANSWER ② 簡易画像を半透明で表示させる!

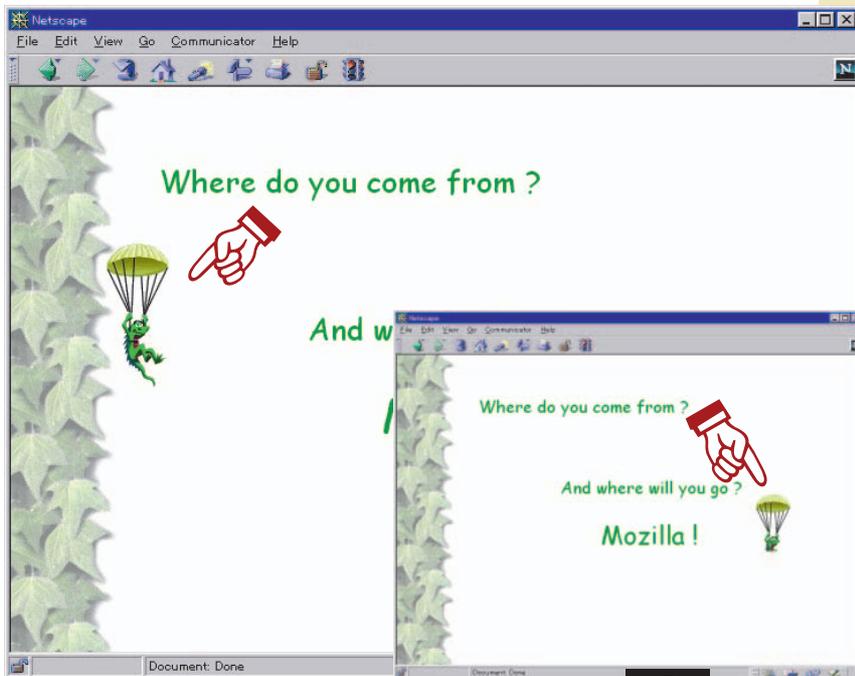
ヒントは2月号で紹介したTIPS「画像にさまざまな表示効果を付ける」に隠されていた。透明度を指定するには「Alpha」というフィルターを使えばいい。あとは「Opacity」で透明度を指定するだけだ。

```
<DIV ALIGN="center" style="filter:Alpha(opacity=40,enabled=1);">
<FONT COLOR="red" FACE="webdings">B</FONT>
<FONT COLOR="blue" FACE="webdings">C</FONT>
:
省略
:
<FONT COLOR="gold" FACE="webdings">S</FONT>
```



岸本政信さん、佐藤義和さん、青木正幸さん、高砂直純さん、ほか多数

ランダムにアニメーション画像を表示させる



紙面では伝えにくいのだが、右の画面の指でさした画像はWWWブラウザ上のあちこちにランダムに現れ、しかも上下に動いているというものだ。ダイナミックHTMLを使えば、高価なショックウェーブ作成ソフトなどを使わなくても複雑なアニメーション効果を得られる。しかもアニメーション画像を決められた場所ではなく画面のあちこちに出現させることができるのだ。ちょっと長いスクリプトが必要になるが、あまり難しいことは考えないで、使い方を覚えてしまえばすぐに実現できる。今あるページに追加するのも難しくはない。このTIPSを活かしてさらに面白いページを作ってみよう。



```
<body onLoad="moveLayer()">
<layer ID="parent" clip="0,0,100,100">
  <layer ID="child" top=100 left=0 clip="0,0,100,100">
    
  </layer>
</layer>

<script language="JavaScript">
var updown = false;
var parent = document.layers["parent"];
var child = parent.document.layers["child"];
var count = 0;

function moveLayer(){
  if (child.top <= 0){
    updown = true;
  }

  if (child.top > child.clip.height){
    updown = false;
    movelt();
  }
  if(updown) child.moveBy(0,2);
  else child.moveBy(0,-2);

  timerID = setTimeout("moveLayer()",10);
  if(count>5)clearTimeout(timerID);
}

function movelt(){
  var px = Math.random()*(innerWidth - child.clip.height);
  var py = Math.random()*(innerHeight - child.clip.width);
  parent.top = py;
  parent.left = px;
  count++;
}
</script>
```

Point

このTIPSでは入れ子にしたレイヤーを使っている。ここでは入れ子になっている外側のレイヤーを「親レイヤー」、中側のレイヤーを「子レイヤー」と呼ぶことにする。親レイヤーは子レイヤーを乗せてランダムに場所を移動し、子レイヤーは画像を表示しながら見え隠れする仕掛けになっている。レイヤーには「clip」と呼ばれる表示領域を指定する値がある。親レイヤー「parent」では「clip="0,0,100,100"」として、左上から100×100の正方形を表示領域とする。子レイヤー「child」は「top=100」、「left=100」、「clip="0,0,100,100"」とする。topとleftは子レイヤーの表示領域の始点（左上）を指定するもので、親レイヤーの左上を基準として指定している。この指定によって初期状態では子レイヤーの表示領域は親レイヤーの表示領域からはずれて

画面上に表示されない。

```
<layer ID="parent" clip="0,0,100,100">
  <layer ID="child" top=100 left=0
  clip="0,0,100,100">...</layer>
</layer>
```

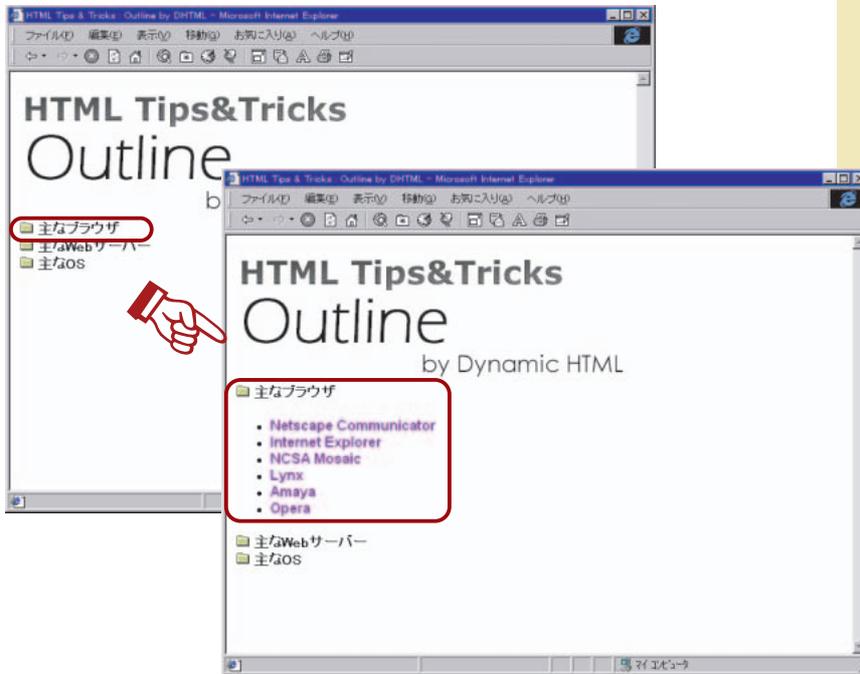
JavaScript部分の「moveLayer()」というプログラムは子レイヤーと親レイヤーの位置関係から子レイヤーを上または下方向に2ドットずつ移動させるといったものだ。このプログラムは「setTimeout」で10ミリ秒ごとに繰り返し実行され、子レイヤーが上に動けば親レイヤーの表示領域に入り、徐々に表示されていく。下に動けば徐々に消えていく。moveLayer()の中で子レイヤーが完全に隠れきったとき、すなわち「if (child.top > child.clip.height)」の条件を満たしたときに、親レイヤー

の位置を変えるプログラム「movelt()」を呼ぶようになっていく。この動作によって次に親レイヤーを表示させる位置を変えている。この表示位置はWWWブラウザの表示領域の大きさと子レイヤーの大きさをもとに「Math.random()」というプログラムを使ってランダムに決めている。

```
var px = Math.random()*(innerWidth -
child.clip.height);
```

なおこのサンプルでは、「count」を使って子レイヤーが5回表示されたら全体の動きを終了するようにしている。ページに応じて適当に数字を変えてみるといいだろう。

ダイナミックにアウトラインを表示させる



ウェブサイトにどんどんコンテンツを追加していくと次第に巨大化していく。コンテンツが増えるのはいいのだが、それに伴ってどうしても目的のコンテンツを見つけにくくなってしまふ。これを解決するために画面の左端などにウェブサイトのアウトラインを表示させる場合が多い。しかし、アウトラインを表示できるスペースは限られていて一度にすべてを表示できない。そこで限られたスペースを有効に使うために、ダイナミックHTMLを使ってアウトラインの文字をクリックするとサブカテゴリを表示し、もう一度クリックするとサブカテゴリを隠す方法を紹介しよう。

1

```
<SCRIPT LANGUAGE="JavaScript">
function checkExpand() {
  var sElem = event.srcElement;
  if ("!" != sElem.id) {
    var ch = sElem.id + "Child";
    var el = document.all[ch];
    if (null!=el) el.style.display = "none" == el.style.display ? "" : "none"
    event.returnValue=false;
  }
}
</SCRIPT>
```

2

```
<SPAN onclick="checkExpand()" ID="Browser">主なブラウザ</SPAN>
<DIV ID="BrowserChild" style="display:none"><BR>
<UL>
<LI><B><A HREF="http://home.netscape.com/">Netscape Communicator</A></B>
<LI><B><A HREF="http://www.microsoft.com/ie/">Internet Explorer</A></B>
<LI><B><A HREF="http://www.ncsa.uiuc.edu/SDG/Software/mosaic-w/">NCSA Mosaic</A></B>
</UL>
</DIV>
```

POINT

さっそく内容を見ていこう。なおここでは説明のために一部を簡略化してある。

まず①のタグから説明する。この部分は<HEAD> ~ </HEAD>に記述する。

「checkExpand()」はアウトラインのサブカテゴリ一部分が表示されていない場合には表示させ、表示している場合には隠すというプログラムだ。次に②のタグについて説明する。

```
<SPAN onClick="checkExpand()"
ID="Browser">主なブラウザ</SPAN>
```

この部分はスタイルシートなどでよく利用するタグだが、「onClick」によって ~ で囲まれた部分をクリックすると先ほど定義した「checkExpand()」を呼び出すことになる。さらにID属性によってこの囲まれた部

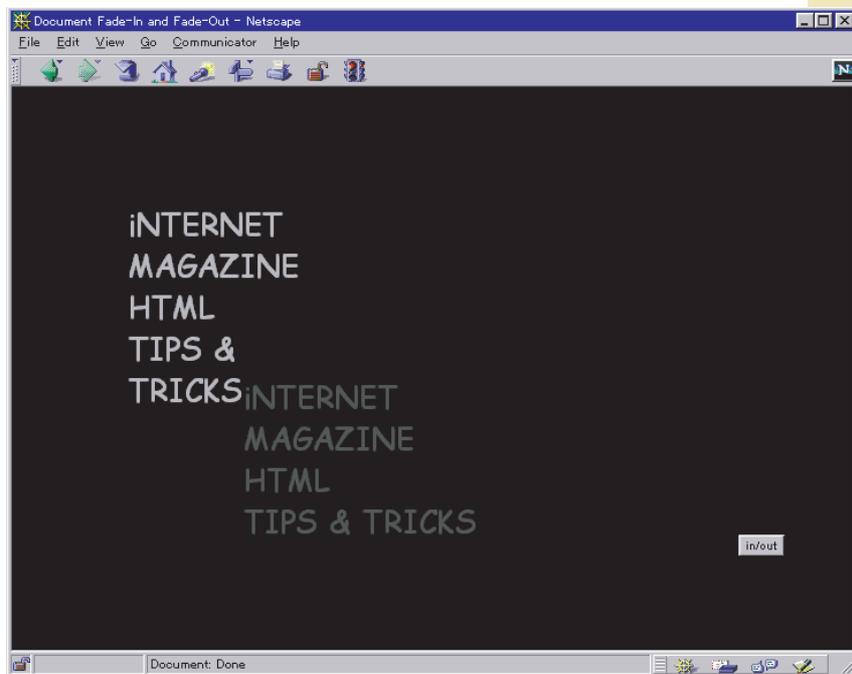
分に「Browser」という名前を付けている。

```
<DIV ID="BrowserChild" style="
display:none"><BR>
<UL>
<LI><B><A HREF="
"http://home.netscape.com/">
Netscape Communicator</A></B>
<LI><B><A HREF="
"http://www.microsoft.com/ie/">
Internet Explorer</A></B>
(省略)
</UL>
</DIV>
```

<DIV> ~ </DIV>で囲まれた部分がサブカテゴリとなる部分だ。ここでもID属性によって

「BrowserChild」という名前を付けている。注意しなければならないのは、この名前は必ず先ほどのタグに付けた名前に「Child」を足したものにしなければならないということだ。サンプルスクリプトでは文字をクリックして表示させる部分はタグを使ったリストだが、<DIV>タグを使って囲めるものなら画像でも文章でも使える。アウトラインのような単純なものだけでなくアイデア次第ではさまざまな場合に適用することも考えられる。たとえば文字をクリックすると、その文字の解説文や画像が出てくるものだ。自分なりのアイデアでこの特性を活かしたページを作ってみてほしい。

文字をフェードイン、フェードアウトさせる



以前に画面の背景色をフェードインやフェードアウトさせるというトリックをパズルのコーナーで出題した。今回はその逆に、前景つまり文字をフェードインやフェードアウトさせるテクニックを紹介しよう。画面右下のボタンをクリックすると文字が浮き出たり消えていたりするものだ。このTIPSはこのコーナーではもうおなじみのダイナミックHTMLを使っている。レイヤー機能とJavaScriptをうまく組み合わせたものだ。一見複雑に見えるが、そんなに難しくはない。文字と表示位置の指定さえ変えればすぐにでも使えるスクリプトなので、ぜひ試してみてください。



```
<form>
<input type="button" value="on/off"
onClick="msgFade()">
</form>

<layer ID="layer1" left=100 top=100 width=200
visibility=show>
</layer>

<script language="JavaScript">
var fade = true;
var layer1 = document.layers["layer1"];

function msgFade(){
  if(fade){
    col_val = "fedcba9876543210";
    fade = false;
  }else{
    col_val = "0123456789abcdef";
    fade = true;
  }
  for(i=0; i<col_val.length; i++){
    col = col_val.charAt(i);
    col+=col+col+col+col+col;
    layer1.document.open();
    layer1.document.fgColor=col;
    layer1.document.write("ここに文章を書きます");
    layer1.document.close();
  }
}</script>
```

POINT

まず最初に文字を表示させるレイヤーを作る。レイヤーの初期値は空にしておく。レイヤーには「ID="layer1"」と名前を付け、必ず幅（width）を指定し、文字を表示させる位置を「left」と「top」で指定しておく。

続いてフェードインとフェードアウトの表示効果をもたらすJavaScriptに取り掛かる。まずフェードインとフェードアウトを判別するための値「fade」と先ほどのレイヤーを指す値「layer1」を作る。

```
var fade = true;
var layer1 = document.layers["layer1"];
```

次に今回のメインの仕掛けとなるプログラム「msgFade()」を作る。「col_val」は文字の色を決める変数だ。if文を使ってフェードインとフェ

ードアウトの場合の色を変えるように設定しておく。col_valの値はおまじない程度に考えておいてもいい。

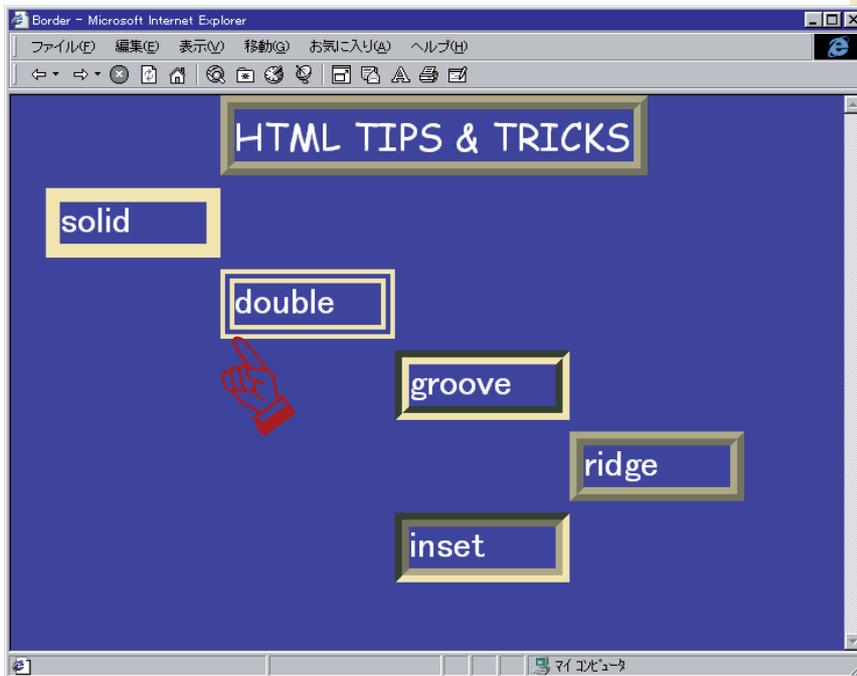
```
if(fade){
  col_val = "fedcba9876543210";
  fade = false;
}else{
  col_val = "0123456789abcdef";
  fade = true;
}
```

最後のfor文では、変数col_valの値を使って16進数で表現される色値「col」を作り、「layer1」の前景色「fgColor」に「col」を入力し、フェードインまたはフェードアウトさせる文字を「document.write」で表示させる。

for文の内容は16回繰り返され、フェードインのときは色の値であるcolが徐々に大きくなり、フェードアウトのときは小さくなるので、見た目には文字が浮き出たり徐々に消えていたりするようになるわけだ。

なお、このmsgFade()を呼び出すためのボタンは上記のようにFORMのボタンでもいいし、アンカーやレイヤーのonClickイベントを使ってもいい。

枠線のデザインを自由にアレンジする



画像や文字に枠線を付けると、それほどインパクトのないものがあたたかみのあるもののように見えることがあるから不思議だ。というのは大きさにしても、枠線によって多少見栄えがするようになるのも事実だ。しかし、これまで枠線に飾りを付ける場合にはわざわざ複雑な表組みを使うなどの面倒な方法をとるしかなかった。そこで今回は簡単に使える枠線をアレンジする方法を紹介する。このTIPSは枠線の四辺の色やスタイル、太さなどを自由に変えることができるものだ。今まで短調だったページも枠線の使い方次第では見栄えがするものに簡単に換えられる。レイアウトの凝ったページで活躍しそうなTIPSだ。



```
<DIV STYLE="border: thick solid moccasin;
position: absolute; top: 30; left: 30; width: 150; height: 50">
CSS / Border
</DIV>
```

TIPS

Point

これまで枠線を作ろうとした場合にはわざわざ<TABLE>タグを使って表を作成する必要があった。しかし、スタイルシートを利用することによって簡単にさまざまな枠飾りが利用できるようになった。しかも今回のTIPSはIE4.0でもネットスケープナビゲーター4.0でも両方使えるので、ほとんどのユーザーが表示できるといっても過言ではない。それではさっそく内容を見ていこう。

```
<DIV STYLE="border: 幅 スタイル色 .....>
```

まずは枠線の幅の指定方法を解説する。「幅」の部分がつくる値は、「thin」、「medium」、「thick」でそれぞれ「太い」、「中間」、「細い」を表している。ほかにも10pxや1.5cm、12ptなどの単位でも指定することができる。次に枠線のスタイルの指定方法を解説する。

「スタイルの部分」がつくる値は、次のようになっている。

```
solid : 実線
double : 二重線
groove : 溝のような線
ridge : 浮き出た線
inset : インセット
outset : アウトセット
```

ほかに「dotted（点線）」と「dashed（破線）」も定義されているが、現段階ではIE4.0、ネットスケープナビゲーター4.0共に実装されていないようだ。

最後に枠線の色の指定方法を解説する。「色」の部分がつくる値は「#CCFF00」のような16進数で指定するRGBカラーか「moccasin」

のような定義された色の名前だ。今までに説明した例では枠線の四辺が同じ指定になってしまうが、四辺を別々に設定することもできる。

```
border-top: 幅 スタイル色
border-bottom: 幅 スタイル色
border-left: 幅 スタイル色
border-right: 幅 スタイル色
```

「border-top」、「border-bottom」、「border-left」、「border-right」はそれぞれ枠線の上部、下部、左部、右部の指定を個別にしている。

このTIPSとスタイルシートを組み合わせれば枠線を自由に配置することができる。

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

・反転表示を制する

今回のパズルはちょっとした復習のようなものになっている。今までいろいろな場面で紹介してきた「レイヤーの配置と動作」がテーマだからだ。レイヤーはこれからウェブページ制作者を目指す人にとって必須の機能であることは間違いない。レイヤーを使うポイントはシンプルズベスト。できるだけ短いスクリプトで注文どおりの動きを表現しよう。そこで今月はレイヤーを駆使した文章の反転表示に挑戦していただく。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名にオリジナル腕時計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“反転表示を制する”にチャレンジ！

「HTMLパズルに挑戦しよう」

宛先

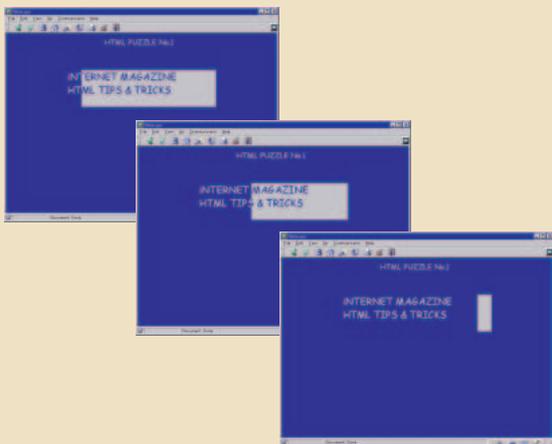
正解が分かった人も、分からなかった人も、ご意見、ご感想など何でもOK、次の宛先にメールしよう。用件の欄には必ず

HTML TIPS & TRICKS

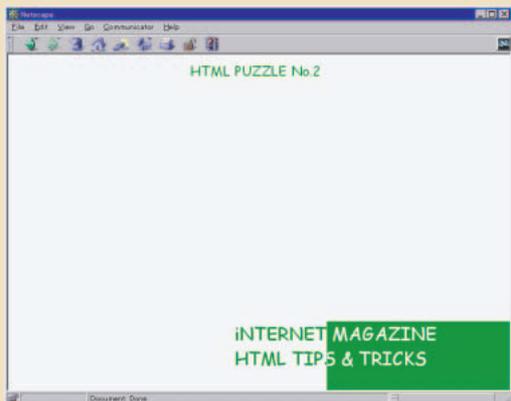
の行を忘れずに。あなたの挑戦を待つ！

impress.co.jp

QUESTION 1



QUESTION 2



文章をダイナミックに反転表示させる
紙面ではわかりづらいが、右の画面は「INTERNET MA
GAZINE HTML TIPS & TRICKS」部分にあるスタイルの階層の部分が左か
ら右まで消えていく。その消えていく部分の背景色と文
字の色が1文字ずつ変わっている。第1問はこの文章がダイナミックに反
転表示されることだ。今回使うレイヤーは2つだけ。レイヤー
の条件を付ける。しかもスクリーンは2つだけ。レイヤー
のページを動かす必要はない。多少高度なテクニックが必要とするが、このコー
ナーでは挑戦してほしい。Javaスクリプト

文字の色が1文字ずつ変わっているのだから、それとも.....

反転表示部分をウィンドウの右下に表示させる！

続いて1問目で使ったダイナミックに反転表示している部分をWWWブラウザの画面の右下に隙間なくぴったりとくっつけて表示させてみよう。WWWブラウザがどんなサイズで使われていても、常に右下に表示されるようにしてほしい。表示される文字が画面から切れたりしてはならない。反転している部分すべてがWWWブラウザ内に収まるようにぴったりと配置してほしい。1問目が解ければすぐにでも解けるだろう。1問目が難しいという人は、WWWブラウザの画面の右下に隙間なく画像が表示されるものでも正解しよう。ただしレイヤー機能を使ったものに限らせていただく。

ブラウザ画面の大きさを得るには.....



[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