


# HTML TIPS & TRICKS

第 44 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 / 高橋登史朗

今月からMozilla Millestone 17(M17)を本誌付録CD-ROMに収録する。ネットスケープ6のもとになっているオープンソースのブラウザだ。開発途上の英語版だが、興味のある人は試してみよう。なお、今月からネットスケープ6はPreview Release(PR)2を対象とするので注意してほしい。

Mozilla Millestone 17 CD-ROM収録先  Mozilla









CD-ROM収録先  Magnavi Ip0010 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(8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4以上
-  インターネットエクスプローラ5以上
-  インターネットエクスプローラ5.5以上
-  ネットスケープナビゲーター4以上
-  ネットスケープナビゲーター6以上
-  Mozilla M17以上



## 9月号「HTMLパズルに挑戦しよう」の解答

今月も多くの方に挑戦していただき、全員が正解だった。DOMとCSSは着実に浸透しているようだ。

お詫び: 9月号の「HTMLパズルに挑戦しよう」の表記に誤りがありました。第1問、第2問ともにIE 4には対応しておらず、IE 5.5に対応しています。お詫びして訂正いたします。



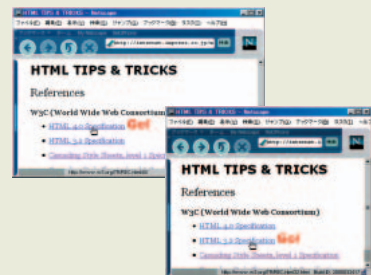
### ANSWER 1 文字を出現させる!

DOMでスタイルシートを操作する手順がわかっていれば簡単。あらかじめ「visibility: hidden」を指定しておいたタグをスクリプトで「visible」に変える。

```

<SCRIPT LANGUAGE="JavaScript">
function show (idstr) {
  obj = document.getElementById (idstr); obj.style.visibility = "visible";
}
function hide (idstr) {
  obj = document.getElementById(idstr); obj.style.visibility = "hidden";
}
</SCRIPT>
;
<A HREF="http://www.w3.org/TR/REC-html40/"
onmouseover="show ('go1') onmouseout="hide ('go1')>HTML 4.0 Specification</A>
<SPAN CLASS="go" ID="go1" STYLE="visibility: hidden;">Go!</SPAN>

```



正解者: 坂部和久、よしとも、富園慎一郎、TOYAMA Nao, Masahiko Murata, 山口雅仁、ENDE、なるる、てらす、杉本 知子、堀江、うおまさ@home (敬称略)



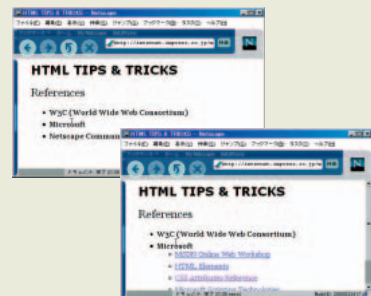
### ANSWER 2 リストを展開せよ!

2問目は、「表示」のときは「非表示」に、「非表示」のときは「表示」に、というように場合分けをする。「display」プロパティが「none」か「block」かで判断する。

```

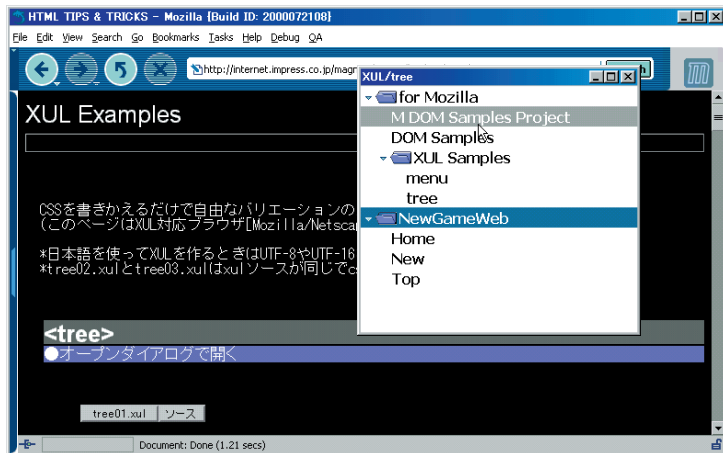
<SCRIPT LANGUAGE="JavaScript">
function showhide (idstr) {
  obj = document.getElementById (idstr);
  if (obj.style.display == "none") obj.style.display = "block";
  else obj.style.display = "none";
}
</SCRIPT>
;
<LI><SPAN onclick="showhide ('list1')>W3C (World Wide Web Consortium)</SPAN>
<UL ID="list1" STYLE="display: none;">
<LI> ..... </LI> .....

```



正解者: 坂部和久、よしとも、富園慎一郎、TOYAMA Nao, Masahiko Murata, 山口雅仁、ENDE、なるる、てらす、杉本 知子、堀江、うおまさ@home (敬称略)

## XULのツリーで階層リスト



1

```
<tree>
<treecolgroup>
  <treecol flex="1"/>
</treecolgroup>
<treecolgroup flex="1">
  <treeitem container="true" open="true" id="moz">
    <treerow>
      <treecell class="treecell-indent" value="for Mozilla"/>
    </treerow>
    :
  </treeitem>
</treecolgroup>
</tree>
```

2

```
treecell-indent: hover { color: #ffffff; background-color: #aaaaaa; }
treeitem[container="true"][open="true"] > treerow { list-style-image: url ("holder0.gif"); }
treeitem[container="true"] > treerow { list-style-image: url ("holder1.gif"); }
```

Point

<tree>関連のタグはHTMLの<TABLE>関連のタグに似ている。対応関係は次のようになる。

<tree>関連	<TABLE>関連
tree	table
treecaption	caption
treecolgroup	colgroup
treecol	col
treehead	thead
treecolgroup	tbody
treeitem	tbody
treefoot	tfoot
treerow	tr
treecell	td

上記のソースでは、xml宣言などのお題目は前回同様なので省略している。CD-ROMからコピーし

てほしい。<tree>関連のタグの並べ方の簡単な例はこうだ（閉じタグは省略）

```
<tree>
<treecolgroup flex="1">
  <treeitem>
    <treerow>
      <treecell value="Test"/>
    </treerow>
  </treeitem>
</treecolgroup>
```

<treerow>が<TR>、<treecell>が<TD>と考えると理解しやすい。<treecell>を続けて並べれば列が増えて、<treerow>から</treerow>までを繰り返せば行が増える。「flex="1"」はほかのブロックとの位置関係を定める属性で必須だ。ソース①の6行目のように<treeitem>に「container="true"」を書いて<treecolgroup>以下を入れ子にしていけば、その<treeitem>内のリスト

が階層化されて、マウスクリックで閉じたり開いたりするようになる。「open="true"」を指定すると開いた状態で、「false」なら閉じて表示する。

ソース②はソース①のXULから指定したCSS。1行目はおなじみの「:hover」、2行目と3行目は「container」属性と「open」属性の状態によってデザインを変更するための分岐だ。CD-ROMのサンプル3つ（tree01.xul ~ tree03.xul）は実はすべて同一のXULコードだ。CSSを替えるだけで無数のデザインを実現できるというわけだ。

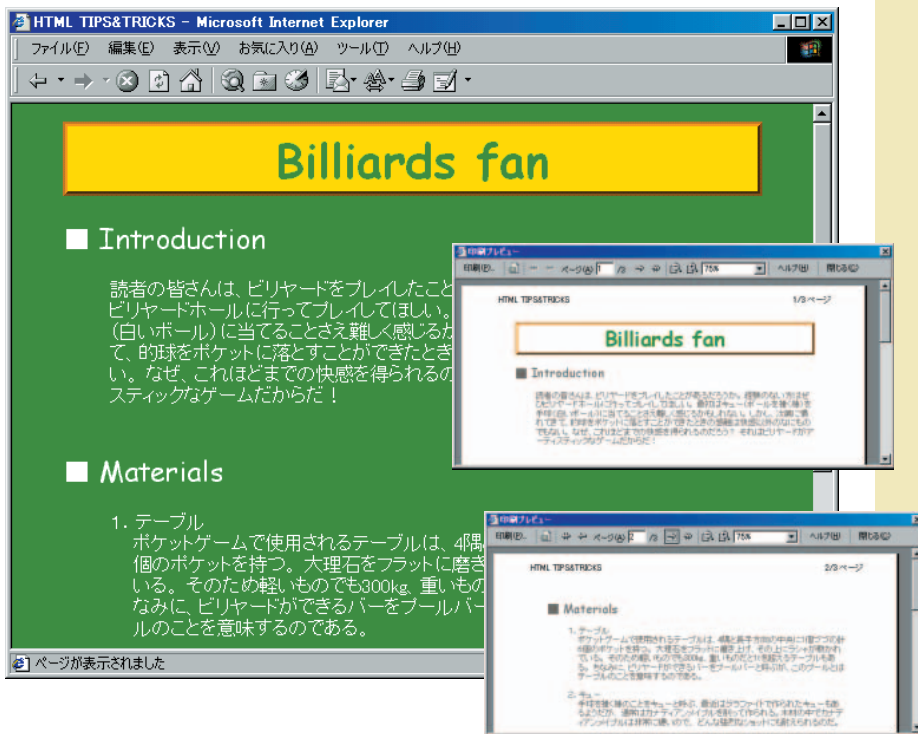
最新のMozillaに関する情報については、次のページがおすすめだ。

MozillaZine日本語版  
[www.mozilla.to/mozillazine/](http://www.mozilla.to/mozillazine/)



前回に引き続き、XULを使ってみよう。今回は<tree>タグだ。文字通りツリー構造の階層リストを作れるのが魅力。直感的でシンプルな使い勝手で、HTMLのテーブルのように行や列も指定できる。付録CD-ROMにはいくつかのバリエーションを収めてあるが、ここではおなじみの階層メニューを紹介しよう。クロスプラットフォームなネットスケープ6やその大元であるMozillaに実装されている言語だから、ウィンドウズでもマッキントッシュでもLinuxでも同じように動く。CD-ROM収録のMozillaでオープンソースブラウザの新機能を確かめよう。（高橋登史朗）

## 指定の場所で改ページ印刷する



ホームページを印刷したとき、意図しないところでページが切り替わってしまい、非常に読みにくい文書になったり、画像が半分に切れてしまったりした経験があるだろう。ここでは、プリントアウトするときに指定の位置で改ページさせるテクニックを紹介しよう。CD-ROMに収録されたサンプルをIE 5.5の「印刷プレビュー」で見よう。プレビューが複数のページにまたがっているが、1枚目を見ると文章の下が余白になっていることに気付いただろう。この位置で改ページするように指定しているのだ。ソースはとても簡単なので、さっそくチャレンジしてみよう。(大内 勇)



```
<DIV STYLE="page-break-after: always">
<P> Introduction</P> 読者のみなさんはビリヤードを.....
</DIV>
<DIV>
<P> Materials</P> 1 . テーブル.....
</DIV>
```

### POINT

Cascading Style Sheet Level2 (以下、CSS2)は、パソコンの画面に表示されるスタイルを定義しているだけではない。音声出力用にも、声の高さなどのスタイルが細かく指定されている。また、ページを印刷したときのためのスタイルも新しく追加されている。現在IEでサポートされているのがこのサンプルだ(ネットスケープ6、Mozilla M 17ではうまく動作しなかった)。

サンプルのソースを一目見て、<DIV>タグで2つに分かれていることに気付いただろう。2つの<DIV>タグの違いは、スタイルシートが設定されているか否かだ。1番目の<DIV>タグには「page-break-after: always」というスタイルを指定している。これがCSS2で定義されている、プリントアウトするときに改ページさせる指定だ。このスタイルを指定すると、印刷するときに要素

の後ろで改ページされるようになる。つまり、1番目の<DIV>要素が1ページ目にプリントされ、2番目の<DIV>要素は2ページ目にプリントされるようになるのだ。

「page-break-after」プロパティに指定できるキーワードは以下のとおりだ。

auto	ブラウザが改ページを判断
always	必ず改ページ
avoid	改ページしない

autoを指定するとブラウザに判断を任せてしまうため、どこで改ページされるかわからない。そのため改ページをはっきり指定したいときは、alwaysを使う。

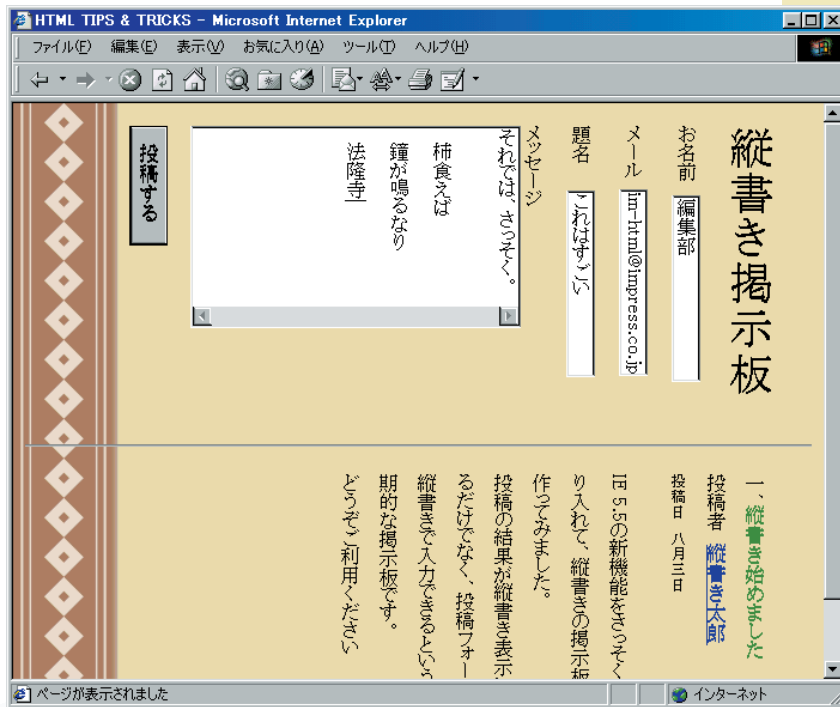
CSS2では、上記の3つ以外に「left」と「right」というキーワードも用意しているが、こ

の2つは現時点でどのブラウザでもサポートされていないので無視してもかまわない。本などを出版する際に、左ページや右ページを指定するためのキーワードなのだ。

サンプルではpage-break-afterプロパティを使って指定した要素の後ろで改ページさせているが、指定した要素の前で改ページさせたいときは「page-break-before」プロパティを設定するといふ。キーワードは、page-break-afterに指定できるものと同じなので、とまどうことはないだろう。

CSS2という新しめのスタイルシートだが、IE4でもサポートされているので、プリントアウトされることを意識してホームページを制作している人は、ぜひ使ってほしいテクニックだ。

## 縦書きフォームを作る



IE 5.5の一番の目玉と言えば、スタイルシートの新機能を使った「縦書き表示」だろう。この連載でも最初のベータ版が登場してすぐに使い方を紹介したのはご存じのとおり(3月号)。正式版が公開されたことで、さっそく試してみた人も多いのではないだろうか。縦書き表示はたいいていのタグに指定できる。3月号では<DIV>タグに指定しただけだったが、フォーム関連のタグにも使えるはずだ。テキストボックスやボタンを縦書きにして入力フォームを作れば、雰囲気のある掲示板だって作れる。新機能をいち早く取り入れて、誰も見たことのない世界初の縦書き掲示板を目指そう。(編集部)



```
<STYLE TYPE="text/css">
FORM { writing-mode: tb-rl; font-family: 'M S P 明朝'; }
INPUT, TEXTAREA { writing-mode: tb-rl; font-family: 'M S P 明朝'; }
</STYLE>
```

### POINT

IE 5.5の縦書き表示機能をおさらいしよう。タグにスタイルシートを設定し、「writing-mode」プロパティの値を「tb-rl」にする。これだけで指定したタグの中のテキストが縦書きになり、「。」や「、」などの表示も自動的に縦書きにふさわしいものになる。縦書きの中で数字など横書きにしたい部分があれば、次のようにwriting-modeの値を「rl-tb」とすればいい。

```
<SPAN STYLE="writing-mode: rl-tb">
99</SPAN>年
```

この縦書き表示には、いくつか注意する点がある。まず、<BODY>タグには指定できない。そのため、ページ全体をまるごと縦書きにはできない。<DIV>タグなどを使ってページの中をいくつかの

ブロックに分けて指定する必要がある。また、writing-modeとともに、スタイルシートでwidthやheightを指定してブロックの縦や横の幅を固定したほうがきれいに表示される。

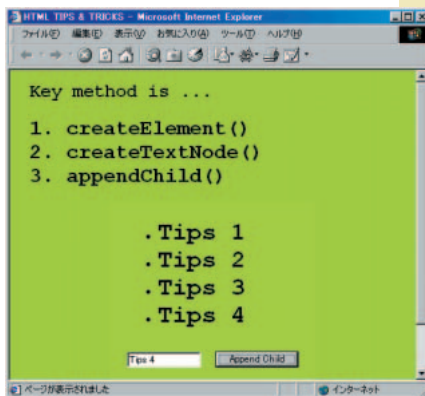
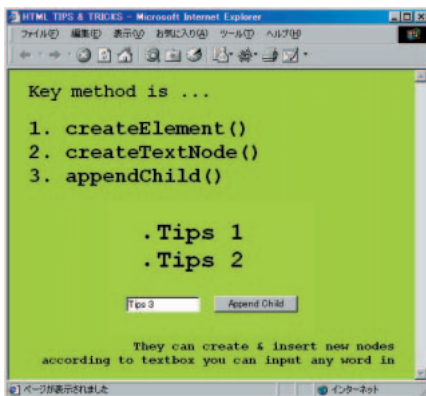
ここまでは3月号で紹介したとおりだが、今回のようにフォームを縦書きにするときは、もう1つ注意点がある。IE 5.5の縦書き表示は、指定したタグの中のタグ、たとえば<DIV> ~ </DIV>の中の<A> ~ </A>にも引き継がれる(継承される)が、タグによっては例外がある。フォームの部品である<INPUT>タグや<TEXTAREA>タグには継承されないのだ。テキストボックスやボタンを縦書きにしたいときは、上記のソースのように<INPUT>タグや<TEXTAREA>タグにもwriting-modeを別に指定しなければならない。なお、IE 5.5ではフォームの部品でも<SELECT>タグだけ

は縦書きにならないようだ。

これで縦書きフォームができあがる。テキストボックスに入力すると、ワープロのようにマウスカーソルが横向きになり、入力中の文字まできちんと縦書きになるのが面白い。サンプルのようにフォントに明朝体を指定すれば、さらに凝ったものになる。掲示板を作るなら、投稿結果も縦書きにして全体の雰囲気を統一するといい。投稿全体を縦書きにするよりも、投稿ごとに縦書き表示を指定した<DIV>で囲って、段組みが縦に並ぶように見せたほうが、使い勝手はいいだろう。

なお、この掲示板はあくまでサンプル用のダミーHTMLだ。実際に投稿できるわけではない。CGIの腕に覚えがある人は、自分で掲示板プログラムを作って公開してみたいかがだろうか。

# ダイナミックにノードを追加する



9月号では、DOMテクノロジーを使ってHTMLオブジェクトを操作し、ダイナミックにテキストを表示するTIPSを紹介した。ただし、あらかじめテキストの「入れ物」にあたる空のノードを作成しておいてから、テキストを変更したものだ。今回はさらに実用的になるように、ノードをダイナミックに作成してテキストを追加する方法を紹介しよう。DOMの理屈が難しいと感じる人は、「ウィンドウズのエクスプローラで、新しいフォルダを作って、「新しいフォルダ」という名前を自分の好きな名前に変える」という操作を想像すれば、イメージがつかみやすいだろう。（藤井幸孝）



1

```
<SCRIPT LANGUAGE="JavaScript">
function appendNode (text) {
  var oList = document.getElementById ("samplelist");
  var oNewNode = document.createElement ("LI");
  oList.appendChild (oNewNode);
  var oNewTextNode = document.createTextNode (text);
  oNewNode.appendChild (oNewTextNode);
}
</SCRIPT>
```

2

```
<UL ID="samplelist">
<LI>Tips 1
<LI>Tips 2
</UL>
<INPUT ID="newtext" TYPE="text" VALUE="5" SIZE="8">
<INPUT TYPE="button" VALUE="Append Child Node"
onclick="appendNode (document.getElementById('newtext').value)">
```

POINT

先月説明したように、DOMはHTML文書の中タグやテキストをツリー構造で把握するモデルだ。<UL>と<LI>を使ったリストは、<UL>が「親ノード」、<LI>が「子ノード」となっているので、ツリー構造としてわかりやすい。ここでは、子ノードである<LI>を追加してみよう。ソース②のように<UL>リストを作り、いくつかの<LI>を並べておこう。<UL>タグには、名前（ID属性）を付ける。今回は「samplelist」とした。この名前はソース①の関数で使うことになる。

ソース①の関数「appendNode」は、リストに項目を1つ追加し、追加した部分に引数「text」で指定した文字列を表示する。まず準備として、「document.getElementById」を使って「samplelist」オブジェクト（<UL>のオブジェクト）を「oList」という簡単な変数で扱えるよう

にしておこう。次に子ノードとなる新しいオブジェクト「oNewNode」を作る。ここで使っているのが、「document.createElement」メソッドだ。引数には、HTMLのタグ名を入れる。今回は<LI>タグを作成するわけだから、「createElement ("LI")」となる。

続いて、この新しい<LI>オブジェクト（oNewNode）を<UL>オブジェクト（oList）の子ノードにしよう。オブジェクトを作っただけでは、ツリー構造の関係がわからないからだ。そのためには「appendChild」メソッドを使う。引数には子ノードになるオブジェクトが入る。このメソッドは親となるオブジェクトのもので、「oList.appendChild(oNewNode)」となるわけだ。

これで、もとは項目が2つだった<UL>のリストに3つめの<LI>要素を追加できた。これだけでは<LI>の中身は空っぽなので、文字列を書き込む

作業が必要だ。先月号を覚えているだろうか。DOMでは、タグの中のテキストもオブジェクト（=テキストノード）として扱われる。テキストノードを追加するには、「createTextNode」というメソッドを使う。ここでは関数appendNodeの引数「text」から「oNewTextNode」というテキストノードを作成しよう。作成したテキストノードは、タグの場合と同じく親ノードの子供として追加する。ここでは親は、先ほど作成した<LI>タグつまりoNewNodeオブジェクトなので、「oNewNode.appendChild(oNewTextNode)」となるわけだ。appendChildの使い方は、子ノードがタグでもテキストでも同じになる。

最後に、ソース②のように入力された文字列を使って関数appendNodeを呼ぶフォームを作成すればおしまいだ。

# HTMLパズルに挑戦しよう

## 隠されたトリックを解き明かせ!



大好評発売中!



今月のテーマ

### 複雑な配置を制する

これからネットスケープ6が普及することで期待できるのが、スタイルシートを使ったページ要素の自由な配置だ。ちょっと複雑なスタイルシートを指定すると表示が崩れるナビゲーター4とは違い、ネットスケープ6とIE 4以上では位置を指定するプロパティを好きなように組み合わせても安全だ。そこで今月は、配置に関するスタイルシートの機能を駆使して、HTMLだけを使ったこれまでのレイアウトを置き換える問題に挑戦してみよう。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“複雑な配置を制する”にチャレンジ!

### 「HTMLパズルに挑戦しよう」

宛先

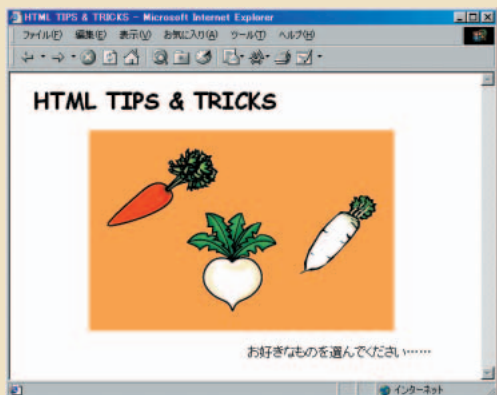
正解がわかった人も、わからなかった人も、ご意見、ご感想など何でもOK、次の宛先にメールしよう。用件の欄には必ず  
HTML TIPS & TRICKS  
の1行を忘れずに。あなたの挑戦を待つ!

✉ [im-html@impress.co.jp](mailto:im-html@impress.co.jp)

なお、締め切りは9月10日とさせていただきます。

## QUESTION 1

イメージマップを取り替える!



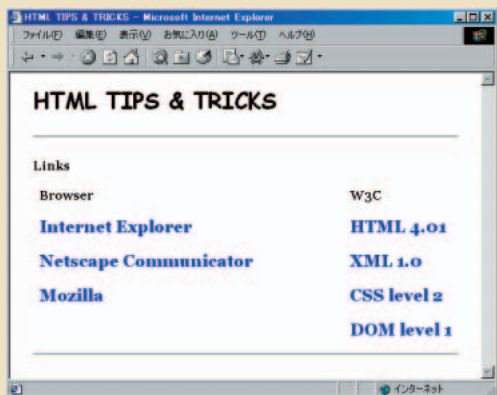
左のサンプルでは、リンク用の画像を並べたブロックが見出しの下に置かれている。この3つの画像は、スタイルシートを使って配置したものだ。これを再現してほしい。もちろんこのブロック全体が1つの画像になっているわけではない。ポイントは、見出しとブロックの間に文字などの要素をいくつ挿入しても、ブロックと一緒に画像も下に移動し、画像の間の位置関係は変わらないということだ。つまり、単純に「position: absolute」を使ってページの左上からの座標を指定しているだけではない。これがわかれば、画像を並べてちょっとしたメニューを作るとき、いちいちイメージマップ用の画像を用意する必要はなくなるぞ。



relativeとabsoluteを組み合わせる……

## QUESTION 2

左右に並べろ!



2問目のサンプルは、一見ありがちなレイアウトに見える。リンク集が2段の段組みのようになっていて、ブラウザのウィンドウサイズを横に広げても、左の段はページの左端に、右の段は右端に並び、<TABLE>タグを使えば簡単だ。2つの列を作って<TD>タグにalign属性を指定すればいい。ところが、このサンプルはスタイルシートだけで配置しているのだ。第1問と同じように、見出しとこのリンク集の間に文字などを追加しても、下に位置がずれるので、ページ全体の中での座標を固定しているわけではない。基本的にやり方は第1問と同じで、あるプロパティを活用するだけだ。さあ、トライしてみよう。



「右の位置」を指定する……



## [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)