

HTML MANIAX

第2回

CSS(スタイルシート)で段組みを作る

前回紹介した「ブラウザ振り分けでCSSを切り分ける」テクニックを使えば、最新ブラウザでCSSを活用する準備は万端だ。今月からはよいよCSSの奥深いワザを探求していこう。まずはCSSで段組みを実現する方法を考える。段組みといえばこれまではテーブルを使うのが常識だった。CSSでレイアウトすれば、すっきりとしたメンテナンスしやすいページができてテーブルよりも有利だ。





文:佐藤和人

HTML MANIAXで紹介したテクニックは [Jump internet.impress.co.jp/maniax/](http://internet.impress.co.jp/maniax/) でソースを公開!






テーブルレイアウトよ、さらば!

テーブルレイアウト

 コンテンツ1	 コンテンツ3
 コンテンツ2	 コンテンツ4

```
<tr><td> コンテンツ1</td>
  <td> コンテンツ3</td></tr>
<tr><td> コンテンツ2</td>
  <td> コンテンツ4</td></tr>
```

CSSで配置する

 コンテンツ1	 コンテンツ3
 コンテンツ2	 コンテンツ4

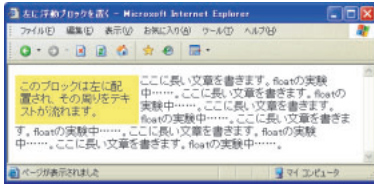
```
<div id="block1"> コンテンツ1</div>
<div id="block2"> コンテンツ2</div>
<div id="block3"> コンテンツ3</div>
<div id="block4"> コンテンツ4</div>
```

現在、世の中のウェブサイトの多くは、テーブルを使ったレイアウトでデザインされている。これにはさまざまな問題がある。何とんでもHTMLを記述するのが面倒だ。また、本来table要素は表を作成するためのもので、デザイン目的で使うとアクセシビリティが低下する。

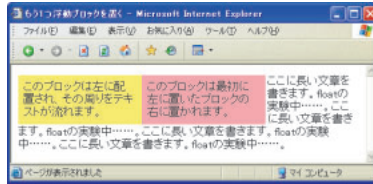
CSSでテーブルを100%置き換えるのは無理なので、筆者はテーブルレイアウトを必ずしも否定はしない。問題は、「コンテンツの並べ方がややこしくなる」ことだ。左上の図のようにテーブルを使うとする。コンテンツ1 2 3.....と読ませたくても、ソースにはコンテンツ1 3 2.....の順で書くしかない。これではソースを編集するときに効率が悪いし、音声読み上げソフトで利用すると、意味の通らないページになってしまう。CSSで配置を工夫すれば、コンテンツの順番が明快なHTMLを元にして柔軟なデザインができる。

[浮動ブロックを極める]

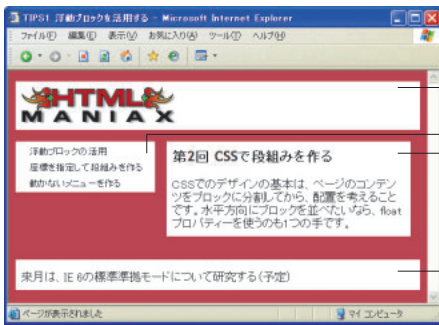
CSSで段組みを作るのに一番手取り早い方法は、「浮動ブロック」の利用だ。float プロパティを指定した浮動ブロックは、文字を回り込ませるのに使えるだけでなく、段組みのようなレイアウトも作れるのだ。



左に配置される浮動ブロックを作る。



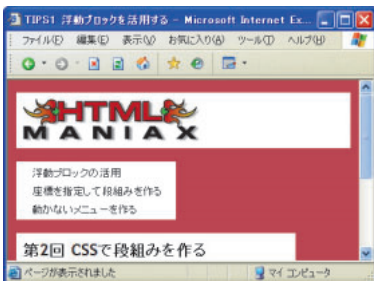
もう1つ浮動ブロックを作ると、2つのブロックが左右に並ぶ。



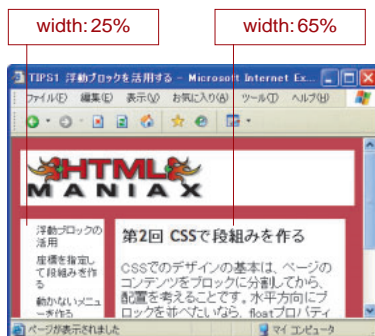
- A IDが「head」のdiv要素
- B IDが「menu」のdiv要素 (float: left)
- C IDが「content」のdiv要素 (float: left)
- D IDが「foot」のdiv要素 (clear: left)

layout1-1.css

```
div#head { margin-bottom: 1em; } ..... A
div#menu { float: left; width: 12em; ..... B
  margin: 0 1em 1em 0; }
div#content { float: left; width: 20em; ..... C
  margin-bottom: 1em; padding: 0.5em; }
div#foot { clear: left; padding: 0.5em; } ..... D
```



ブラウザの幅を狭めると上下に並ぶ。



%で幅を指定して伸び縮みさせる。

🔥 注意点

今月のサンプルでは、先月紹介したテクニックを使い、ネットスケープナビゲーター4にCSSが適用されないようにしている。



「float: left」× 2 で
あら不思議!

CSSをかじった人なら、float プロパティをご存じだろう。float: leftを指定すると段落や画像が左に配置され、テキストがそのまわりを流れるようになる。float: rightなら右に配置される。このように左や右に配置された要素を「浮動ブロック」と呼ぶ。さてfloat: leftを指定した要素の次に、またfloat: leftを指定した要素を並べるとどうなるだろう。2つの浮動ブロックが段組みのように左右に並ぶのだ。

ソース (layout1-1.css)は、浮動ブロックのこの性質を利用したCSSだ。Aは普通にページの上に配置されるdiv要素。Bはfloat: leftを指定してページの左に配置するdiv要素で、サイトのメニューなどに使う。Cのdiv要素にもfloat: leftを指定すれば、段組みのようにページの右にコンテンツ領域を置ける。Dのdiv要素には、clear: leftを指定してテキストの回り込みを解除する。ここは連絡先や著作権表示などページのフッターとして使える。



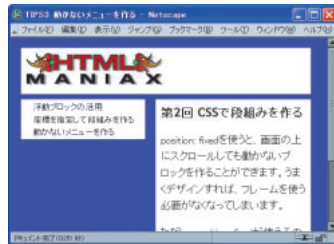
上下に並んでしまっても
あわてるな!

float プロパティを使うときの注意点は、必ずwidthプロパティを指定して浮動ブロックの幅を明示することだ。ソースでは単位がemの値で幅を指定している。この場合は、ブラウザの横幅を狭めると、2つの浮動ブロックが上下に並んでしまう。浮動ブロックの性質からしてあたりまえなのだが、ちょっと格好が悪い。これを避けるには、左の下端の画面のように、widthの単位を%にすればいい。こうすれば、ブラウザの幅に合わせて浮動ブロックが伸び縮みするようになる。

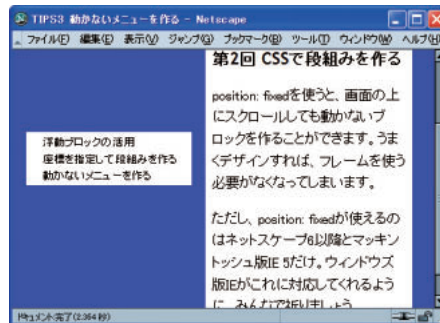
また、2つの浮動ブロックをさらにdiv要素で囲み、そのdiv要素のwidthプロパティを40emなどに固定するという手もある。

[動かないメニューを作る]

CSSで段組みを作る方法を紹介したついでに、ちょっと変わったワザを見てみよう。ページをスクロールしてもメニューを同じ場所に表示させ続けるテクニックだ。CSSを活用したページデザインの未来が見えてくるだろう。



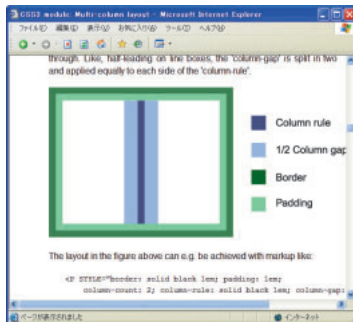
一見左ページと同じテクニックを使っているようだが……



なんと！スクロールしてもメニューの位置がそのままだ。

layout3-1.css

```
div#head { position: absolute; left: 1em; top: 1em;
width: 28em; height: 4.5em; }
div#menu { position: fixed; left: 1em; top: 6em;
width: 12em; }
div#content { position: absolute; left: 14em; top: 6em;
width: 14em; padding: 0.5em; }
```



現在草案段階の「CSS3 module: Multi-column layout」
www.w3.org/TR/css3-multicol/

将来実現するかもしれないCSS

```
p#column { border: solid black 1em; padding: 1em;
column-count: 2; column-rule: solid black 1em;
column-gap: 4em; }
```

フレームいらずの固定配置

CSSのpositionプロパティーは、absolute、relativeのほか固定という値を取れる。左のソース（layout3-1.css）の「div#menu」のようにposition: fixedを指定すると、ページをスクロールしても動かないブロックになる。サイトのメニューのように、常に表示させておきたい部分に指定すると便利だ。これまではこうした機能を付けるにはフレームを使うしかなかったが、CSSを活用すればテーブルだけでなくフレームを使ったレイアウトもいなくなるのだ。

ただし、残念なことに、これが使えるのはネットスケープ6以降とマッキントッシュ版のIE 5だけだ。ウィンドウズ版のIEでは効果がない。IEの次バージョンがposition: fixedに対応してくれることに期待しよう。

CSSで本当の段組みができる？

今まで紹介してきたワザは、本当の段組みではなく、言わば「段組みもどき」だ。現在のCSSでは、ページ中のコンテンツを左右方向に並べるのはなかなか難しい。

現在W3Cでは、次世代のCSSであるCSS 3が検討されている。CSS 3の草案は、「Color」や「Text」などのモジュール別に発表されているが、左のURLの「CSS3 module: Multi-column layout」は、そのものずばり、CSSで段組みを表現する仕様の草案だ。ソース（将来実現するかもしれないCSS）はこの草案から引用したもので、column-countで2段組みのレイアウトを作成し、column-ruleで2つの段の間に線を引き、column-gapで段の間隔を4文字分としている。もちろん、現在どんなブラウザも対応していない。将来この機能がブラウザに採用されるようになれば、テーブルレイアウトを使う必要性はまったくなくなるだろう。



[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