

HTML TIPS & TRICKS

第 22 回

誰よりも早く 最新のHTMLを使ってみよう

藤井 幸孝 川田 哲 大内 勇

今回のTIPS & TRICKSは、2や3のようにやや長めのスクリプトを書いたり、4のようにActiveXコントロールを使うものとなったので、付属CD-ROMにソースのHTMLファイルを収録した。テキストエディターなどでHTMLファイルを開き、本誌を見ながら必要な部分を自分のHTMLにコピー＆ペーストして使ってみてほしい。それでは今月もさまざまなTIPSをお届けしよう。



CD-ROM収録先 A Magnavi Ip9812 Htmltips

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



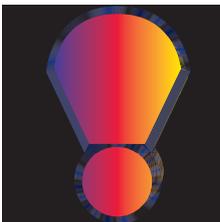
インターネットエクスプローラ4.0以上



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

<BLINK>と<MARQUEE>タグの動きを未対応のブラウザで動作させると言っても、あまりピンとこないものかもしれない。今までこのコーナーで扱ってきたダイナミックHTMLを使えばよいのだ。つまり、スタイルシートやレイヤーとJavaScriptを組み合わせて動きを作ればよい。それでは解答を発表しよう。



ANSWER ① エクスプローラでBLINKを実現せよ!

単に表示(visible)と非表示(hidden)を交互に入れ替えているだけだ。タイマーを設定すれば、一定時間ごとに入れ替えが起き、点滅しているように見える。

```
<SCRIPT LANGUAGE="JavaScript">
function ieblink() {
  if (blinkblock.style.visibility == "visible") blinkblock.style.visibility = "hidden";
  else blinkblock.style.visibility = "visible";
}
</SCRIPT>
<BODY onLoad="setInterval('ieblink()', 750)">
<DIV ID="blinkblock" STYLE="visibility:visible;">BLINK</DIV>
```



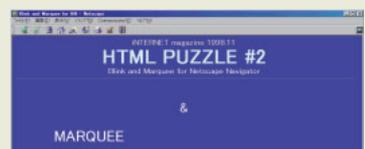
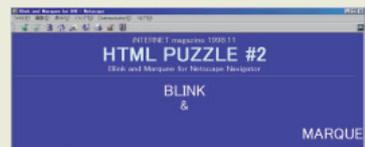
ANSWER ② ナビゲーターでMARQUEEを実現せよ!

レイヤーを右端(innerWidth)に置いてから、leftの値を減らしていくと、右から左へスクロールしているように見える。見えなくなったら、再び右端に置けば現れる。

```
<SCRIPT LANGUAGE="JavaScript">
function nnmarquee() {
  document.marqueeblock.left = innerWidth;
  document.marqueeblock.visibility = "show";
  setInterval('mving()', 100);
}
function mving() {
  document.marqueeblock.left -= 5;
  if (document.marqueeblock.left <= -250)
    document.marqueeblock.left = innerWidth;
}
</SCRIPT>
<BODY onLoad="nnmarquee();">
<LAYER LEFT=0 TOP=250 VISIBILITY="hide" NAME="marqueeblock">
MARQUEE</LAYER>
```



正解者：町谷円さん、Heronさん、良知敬介さん、田中博英さん



正解者：Heronさん、田中博英さん

フォームの部品を囲む



右のサンプルは、アンケート送信用に作成されたフォームだ。ラジオボタンやテキストボックスなどのフォームの部品が青い枠で囲まれているのがわかる。スタイルシートを使って枠を付けているだけだと思うかもしれない。しかし、サンプルをよく見ると、枠の左上に「あなたの情報」や「今月の評価」といった説明が付いていて、その文字は枠線の上に載せられたように表示されている。文字の位置をスタイルシートですらしているわけではない。HTMLで指定するだけで、自動的に説明付きの枠線が表示されるのだ。いったいどうやっているのだろうか？ フォームを使ったページを作成している人は必見だ。



```
<FIELDSET>
<LEGEND> あなたの情報 <BR> </LEGEND>
お名前: <INPUT TYPE="text" NAME="name" SIZE=40><BR>
メールアドレス: <INPUT TYPE="text" NAME="mail" SIZE=40><BR>
</FIELDSET>
```

Point

HTML4.0では、7月号のTIPS&TRICKS1で紹介したように、<LABEL>タグやアクセスキーなど、フォームにさまざまな改良が加えられた。フォーム内の部品をグループにまとめるための<FIELDSET>タグもその1つだ。<FIELDSET>を使えば、フォーム内の数多くの部品をいくつかのジャンルに分けることができるので、入力する人にとってわかりやすいインターフェイスになる。

<FIELDSET>には、<LEGEND>タグで説明用のキャプションを付けることができる。上のサンプルのように、<FIELDSET>タグのすぐ次に<LEGEND> ~ </LEGEND>で囲んだ説明を書き、それから<INPUT>などのフォームの部品を記述する。

<LEGEND> ~ </LEGEND> 内に
タグを入れているのは、ネットスケープナビゲーターでもうまく表示されるようにするためだ。

HTML 4.0では、<FIELDSET> はあくまでフォームの部品をグループ化するのが目的のタグであり、<LEGEND>でグループの説明を付けるとされているだけだ。サンプルのように枠で囲まれたり、説明が枠線の上に載ったりするのは、IE4で表示した場合だけかもしれないということに注意しよう。ネットスケープナビゲーターが将来<FIELDSET>をサポートしたときには、同じ表示になるとは限らない。HTML 4.0では、見栄えのためよりも音声読み上げソフトなどのために<FIELDSET>が利用されると考えられているようだ。

「見出し付きの囲み」を作るために、テキストだけを囲んだ<FIELDSET>を使ってしまいたくなるが、フォームの部品を囲む以外の目的で使うのは避けたいほうがいいだろう。

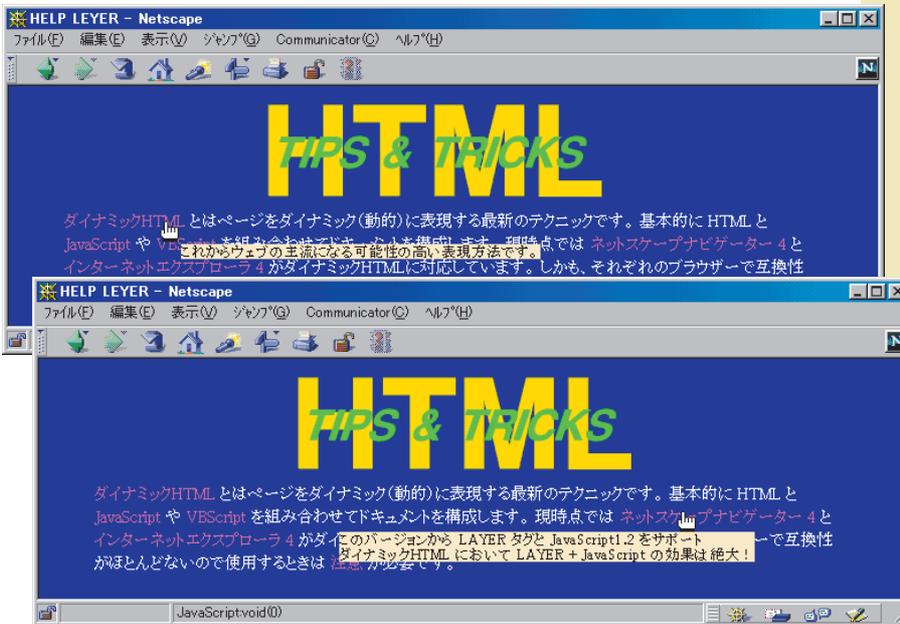
<FIELDSET>は「グループ化」という構造を表す

タグではあるが、サンプルでは次のようにスタイルシートを使って少々見栄えをよくしてみた。

```
FIELDSET { padding: 8px; border-color:
blue; background-color: #FFFFFFC0; }
LEGEND { font-weight: bold;
color: green; }
```

<FIELDSET>のスタイルを見てみよう。何も指定しないと、枠線とその中身がくっついた感じになるので「padding」で余白を空けた。さらに「border-color」で枠線に青い色を付け、「background-color」で背景を黄色くしている。<LEGEND>では、「font-weight」や「color」といったスタイルを指定して、説明の文字を太くしたり、色を付けたりしている。

ナビゲーターでヘルプを出現させる



2月号のTIPS & TRICKS 4「文字列に対してヘルプを出現させる」で、マウスカーソルを文字の上に載せると小さいヘルプを表示させる方法を紹介した。これは、各種タグの「TITLE」属性をサポートしているIE 4だけが対応しているもので、他のブラウザを使っている人には意味がないTIPSだった。今回紹介するのは、ナビゲーター4で同様のヘルプを表示させるTIPSだ。レイヤーを使うため、単に文字を表示させるだけでなく、文字色や背景色を変えるなどIE 4のTITLE属性ではできないさまざまな表現ができる。ウェブデザイナーにとっては朗報となるだろう。それではさっそく見てみよう。



1

```
var msg = new Array();
msg[0]="これからの.....";
msg[1]="ネットスケープ社が.....";
:
function show(x, y, str) {
  obj = document.layers["helpLay"];
  if (x < outerWidth / 2) obj.moveTo(x, y + 20);
  else obj.moveTo(x - 300, y + 20);
  obj.document.write(str);
  obj.document.close();
  obj.visibility = "show";
}
function hide() {
  document.layers["helpLay"].visibility = "hide";
}
```

2

```
<A HREF="javascript:void(0)"
onmouseover="show(event.pageX, event.pageY, msg[0])"
onmouseout="hide()">Dynamic HTML</A>
```

3

```
<LAYER NAME="helpLay" VISIBILITY="hide"
BGCOLOR="lemonchiffon"></LAYER>
```

POINT

このTIPSは、リンクの上にマウスカーソルを載せることでヘルプを出現させる動的HTMLだ。ヘルプはレイヤーでできているため、ネットスケープナビゲーター4だけで動作する。スクリプトとそれを呼び出す<A>タグ、それに<LAYER>タグの3つから成り立っている。

ソース①のスクリプトを見てみよう。まず、ヘルプ用のメッセージの配列を作成する。「msg[0] = "これからウェブの主流になる可能性の高い表現方法です。";」のように、「~」で囲んだ部分がレイヤーの表示内容になる。「~」の中にはテキストだけでなく、タグを入れて画像を使ったヘルプを作ることもできる。

次に、ヘルプを表示する関数「show」とソース②の<A>タグを見比べてみよう。関数「show」には

3つの引数があるが、これには呼び出し元である<A>タグ内の「onMouseOver=」で渡された値が入る。つまり、「x」にはカーソルの左右の座標「event.pageX」が、「y」にはカーソルの上下の座標「event.pageY」が、「str」にはヘルプのメッセージ「msg[0]」などが入る。「x」と「y」を使ってヘルプ用レイヤーの位置を移動する際には、if ~ elseで位置を調節する。この条件分岐がないとレイヤーの位置がブラウザの外になってしまう、読めなくなる場合があるので注意しよう。ヘルプのメッセージは、「document.write」を使ってレイヤーに文字を書き込んで変更する。書き込んだら「document.close」を呼び出すこと。<A>タグ内の「onMouseOut=」で指定しているのは、ヘルプのレイヤーを非表示状態にする関数「hide」だ。また、「HREF」には「javascript:void(0)」を指定してクリックを無効にしている。

最後に、<LAYER>タグでヘルプ用のレイヤーを作る(ソース③)。「NAME」で「helpLay」という名前を付け、「VISIBILITY」で非表示を指定し、「BGCOLOR」で背景色を付けている。次のように「show」関数を書き換えてレイヤーの「bgColor」を変更すれば、ヘルプの背景色を自由にすることもできる。

```
function show(x, y, color, str) {
  obj = document.layers["helpLay"];
  :
  obj.bgColor = color;
  :
```

■ 拡張ワイプで画像を入れ替える



何れともあれ、左のサンプルを見ていただく。最初に表示された1枚目の画像がだんだん上に縮まり、下から新たに2枚目の画像が現れるのがわかるだろう。そして、2枚目の画像がまた上に縮み、もとの1枚目の画像がまた下から現われ、これが繰り返される。アニメーションGIFを使っているわけではない。2枚の画像の位置と高さを変えてアニメーション表示させているのである。サンプルの画像は、一見すると回転しているように見えるが、これは「拡張ワイプ」と言われるものである。ここでは上下の拡張ワイプだが、少し変えれば左右の拡張ワイプも可能なので、マスターしているいろいろ応用してみよう。



1

```
img1 = "image1"; img2 = "image2";
step = 5;
function wipe () {
  if (document.all(img1).style.pixelHeight >= step) {
    document.all(img1).style.pixelHeight -= step;
    document.all(img2).style.pixelTop -= step;
    document.all(img2).style.pixelHeight += step;
  } else {
    document.all(img1).style.pixelHeight = 0;
    document.all(img2).style.pixelTop = 40;
    document.all(img2).style.pixelHeight = 155;
    document.all(img1).style.pixelTop = 40 + 155;
    temp = img1; img1 = img2; img2 = temp;
  }
}
```

2

```
<IMG ID="image1" SRC="image01.gif"
  STYLE="position:absolute; top:40; left:40;
  width:364; height:155;">

<IMG ID="image2" SRC="image02.gif"
  STYLE="position:absolute; top:195; left:40;
  width:364; height:0;">
```

POINT

まずは、画像の配置だ。幅と高さがまったく同じ2枚の画像を用意する。ソース②のように、1枚目の画像のタグにスタイルシートで「position: absolute;」を指定し、「top」、「left」、「width」、「height」の値をそれぞれ指定して、どこか好きな場所に置く。2枚目の画像は次のようにする。

```
top: 1枚目のtop + 1枚目のheight
left: 1枚目のleftと同じ値
width: 1枚目のwidthと同じ値
height: 0
```

このように設定することで、1枚目の画像のすぐ下に2枚目の画像が置かれるようになる。ただし、最初の段階では、高さは0なので2枚目の画像は表示されない。

次に、2枚の画像をどのように動かしていけばいいかを見てみよう(ソース①)。2つの変数「img1」と「img2」にそれぞれ1枚目の画像と2枚目の画像の名前(ID属性)を最初に入れておき、この2つの変数を関数の中で使っていくことにする。また、変数「step」に一度に動かす幅(ここでは5)を指定しておく。

関数「wipe」の中では、まず「img1」の画像を縮める(「style.pixelHeight」を小さくする)。それから「img2」の画像の位置を上にならなく(「style.pixelTop」を小さくし)、「img2」の画像を伸ばす(「style.pixelHeight」を大きくする)。これを繰り返し行うわけだ。

「step」の分だけ移動させることを繰り返しているため、このままでは「step」の値以下の半端な部分が残ってしまうことになる。そこで、「img1」の画

像の高さが「step」の値より小さくなったら、動かすことをやめて、2枚の画像の位置や高さを次のように調節する。

```
img1のpixelHeight : 0
img1のpixelTop : 2枚目の最初のY座標
img2のpixelHeight : 画像の高さ
img2のpixelTop : 1枚目の最初のY座標
```

こうした調節をしないと、動かしたときに位置がずれてしまったりうまく拡張ワイプができない。調節したら、変数「img1」と「img2」の値、つまり画像の名前を交換する。

最後に、<BODY>タグで「onLoad="setInterval('wipe()', 50);"」として、一定時間ごとに関数「wipe」が呼び出されるようにする。これで2枚の画像が次々と入れ替わるようになる。

データバインディング機能でデータベースを利用 その1



Tabular Data Binding Demo
using data source TDC and TABLE tag

City	00:00-06:00	06:00-12:00	12:00-18:00	18:00-24:00
Los Angeles	20%	20%	30%	40%
Chicago	10%	0%	0%	10%
New York	10%	10%	20%	10%
London	40%	50%	50%	20%
Paris	20%	20%	30%	10%
Berlin	20%	30%	10%	0%
Beijing	20%	10%	10%	10%
Tokyo	30%	40%	40%	10%

These figures are inserted from database(.csv) file
which is bound to this html file by TDC data source object.



1

```
<OBJECT ID="tdcWeather" WIDTH=0 HEIGHT=0
CLASSID="CLSID:333C7BC4-460F-11D0-BC04-0080C7055A83">
<PARAM NAME="FieldDelim" VALUE=",">
<PARAM NAME="DataURL" VALUE="database.csv">
<PARAM NAME="TextQualifier" VALUE=""">
<PARAM NAME="UseHeader" VALUE="true">
</OBJECT>
```

2

```
<TABLE datasrc=#tdcWeather>
<TR>
<TD><SPAN datafld="City"></SPAN></TD>
<TD><SPAN datafld="am_early"></SPAN></TD>
<TD><SPAN datafld="am_late"></SPAN></TD>
<TD><SPAN datafld="pm_early"></SPAN></TD>
<TD><SPAN datafld="pm_late"></SPAN></TD>
</TR>
</TABLE>
```

3

```
'City','am_early','am_late','pm_early','pm_late'
'Los Angeles',20,20,30,40
'Chicago',10,0,0,10
'New York',10,10,20,10
'London',40,50,50,20
'Paris',20,20,30,10
'Berlin',20,30,10,0
'Beijing',20,10,10,10
'Tokyo',30,40,40,10
```

ウェブとデータベースを連結させてダイナミックなページを見せるしくみは、古くはCGIで、最近ではIIS(インターネットインフォメーションサーバー)+ASP(Active Server Pages)で使うADO(ActiveX Data Object)やオラクルのウェブアプリケーションサーバーなどを便利に利用して作成できる。だが、これらのテクニックはサーバー上のアプリケーションの機能に頼っているため、プロバイダーのホームページサービスでは残念ながら利用できなかった。ところが、IE 4にはサーバーにまったく頼らずにデータベースを利用できる機能がある。ここではこの「データバインディング機能」を紹介しよう。

POINT

データバインディング機能は、データソースオブジェクト(DSO)と呼ばれるActiveXやJavaのオブジェクトを使ってデータベースファイルにアクセスし、その中のデータをHTMLに埋め込んで表示させるものだ。DSOはクライアント側で動くので、どんなウェブサーバーを使用しているサイトでも機能する。サンプルでは最も基本的なDSOである、TDC(Tabular Data Control)を使用して、各地の降水確率を表示するHTMLを書いてみた。このようなページは、フォーマットは変化しなくても中身の値だけは毎日変わるので、データベースの値を動的に組み込むことができれば、作成が非常に楽になる。ふだんはデータベースだけを更新すればいいわけだ。

TDCとはIE 4.0以降に付属するコントロールで、CSVファイル(テキスト形式の簡易データベース)にアクセスする機能を持っている。最初に、読み

込み対象のCSVファイルを作成しておこう(ソース①)。一番上の行には各フィールドの名前を書いておく。2行目からが実際のデータだ。都市名と4つの数値、合わせて5つのフィールドを作る。文字列は「'」でくくり、各フィールドはカンマで区切る。行は改行することで自動的に区切られる。このファイルはあとでサーバー上に置いておくことになる。続いて、このCSVファイルを読み込むHTMLファイルを作成する。最も重要なのは、TDCオブジェクトの定義だ。<OBJECT>タグでActiveXコントロールを追加する(ソース②)。ID属性は「tdcWeather」とする。<PARAM>タグの「Filed Delim」は区切り文字の「,」のこと、「DataURL」は読み込むファイル名、「TextQualifier」は文字列をくくった文字の「'」のことだ。「UseHeader」を「True」にするのは、CSVの1行目がフィールド名定義行だということを知らせるためだ。わからないときは上のとおりに書いておこう。

最後に、このCSVデータベースから読み出した値を、テーブルにして表示させよう(ソース③)。<TABLE datasrc=#tdcWeather>として、さきほど作ったTDCオブジェクトのIDを指定する。そして<TD>タグ内でとして、CSVファイルで使ったフィールド名を指定し、このセルにどのフィールドを表示するのかが決まろう。面白いのは、<TABLE>タグ内には<TR>は1組だけでよいこと。あとは自動的にすべての行が読み込まれ、行数だけ<TR>が生成される。日付が変わって降水確率が変わったら、CSVファイルだけを変更すればよい。

データバインド機能では、ADOのように「Recordset」オブジェクトを使って、必要なレコードの必要なフィールドだけを取り出したり、カーソル移動をしたりできる。今回はテーブルによる自動表示を扱ったので、今回はもう少し高機能な処理に挑戦してみよう。

HTMLパズルに挑戦しよう

隠されたトリックを解き明かせ！



今月のテーマ

文字レイアウトを制する

HTML文書の構造を組み立てていくのがHTMLタグの役割なら、文書のデザイン部分を担っているのは、この連載でもずいぶん取り上げてきたスタイルシートだ。色やフォントのためだけでなく、文字をレイアウトするためにスタイルシートを使えば、ふつうのページを見栄えのするデザインに変えることができる。そこで今月は、文字のレイアウトに関するパズル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“文字レイアウトを制する”にチャレンジ！

「HTMLパズルに挑戦しよう」

宛先

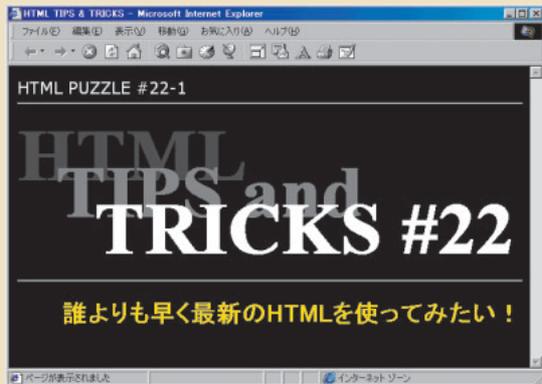
正解がわかった人も、わからなかった人も、ご意見、ご感想など何でもOK、次の宛先にメールしよう。用件の欄には必ず
HTML TIPS & TRICKS
の1行を忘れずに。あなたの挑戦を待つ！

✉ ip-cdrom@impress.co.jp

なお、締め切りは11月10日とさせていただきます。



QUESTION 1 文字を重ねろ！



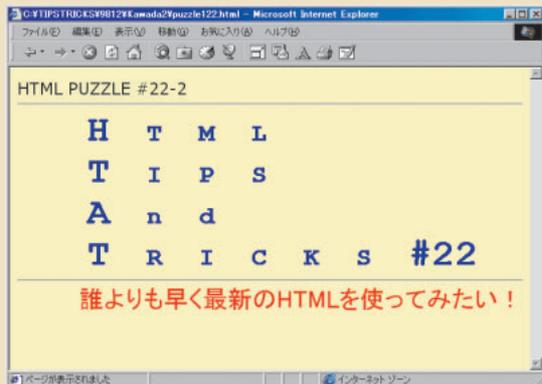
左のサンプルを見てほしい。タイトルの文字にグラデーションがかかり、見栄えのするデザインになっている。といってもグラデーションのかけ方が問題なのではない。よく見ると、文字と文字が重なっているのわかるだろう。サンプルのように大きめのフォントを用意して、色違いの文字を重ねて書くだけでもずいぶん印象が違ってくる。第1問では、文字と文字を重ね合わせる方法を考えてほしい。文字をいくつか用意して、「margin-top」を使って配置する方法もある。しかし、あるスタイルシートのプロパティを1つ加えるだけで、もっとスマートにできる方法があるのだ。いったい何を加えればよいだろうか。さあ、やってみよう。



文字の高さより何かを小さくすれば……



QUESTION 2 間隔を開けて文字を配置しろ！



第2問も、まずはサンプルを見てほしい。タイトルの部分に文字が等間隔で配置されており、その間隔が比較的大きく取ってあるのがわかるだろう。こんなふうに広い間隔を空けて文字を置くと、これもなかなか見栄えのするデザインになる。一定の間隔を開けて文字を配置するにはどうすればいいか、それが第2問だ。文字と文字の間に全角スペースをいくつか埋めていくという手もあるが、あまりよい方法とは言えない。第1問と同様に、あるスタイルシートの属性を1つ加えるだけでできるスマートな方法があるので、それを考えてほしい。第1問、第2問ともに、解答は単純なものなので、スタイルシートを詳しく調べて挑戦しよう。



「文字 - 間隔」というプロパティは……



[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