

: Hypertext Preprocessor

無料サービス、ホスティング対応情報も満載！

今日からはじめる PHP入門

近頃、プロのウェブ製作者の間で話題になっているのが、PHPというスクリプト言語だ。PHPはこれまでのCGIやPerlよりも簡単であるにもかかわらず、データベースと連携した大規模なサイト構築にも対応できる豊富な機能を備えている。いままでCGIを使ってきた人はもちろんのこと、これからCGIを覚えようとしている人も、今日からはPHPをマスターしてワンランク上のウェブ作成方法を身につけよう。

宮本和明 + 編集部

簡単便利なウェブスクリプト

これからの スクリプトは PHPで決まりだ!

That's PHP

① PHPはHTMLファイルに 組み込める

PHPではスクリプトをHTMLファイルの中に直接記述できるので、CGIに比べて管理が簡単になる。

② スクリプト部分が HTMLに置き換わる

実際にユーザーがアクセスすると、PHPのスクリプト部分はHTMLに置き換わるので、ユーザーにスクリプトが見えない。

PHPで管理が簡単に

PHPは、これまでのCGIに代わるスクリプト言語として、ウェブ製作者の間で注目を集めている。PHPがこれまでのCGIと大きく異なるのは、スクリプトがHTMLの中にそのまま記述できる点にある。

PHPの実際の記述例は、右の①のようになる。ファイル中の「<?php」から「?>」の間がPHPのスクリプトで、この部分はサーバー側で実行されたプログラムの結果に置き換わる。それ以外の部分は普通のHTMLファイルと変わらないし、これがなぜ便利なのかは、CGIと比べてみるとよくわかる。

多くのCGIで使われているPerlで①と同じような動作をするスクリプトを記述すると②のようになる。<head>や<body>といったHTMLタグも、すべてPerlのスクリプトとして記述しなければならないため、PHPに比べて複雑になってしまうのがわかるだろう。

こうした問題は、SSIを使えばある程度は解消できる。SSIでは③のようにHTML中に「<!--\$exec ~>」として呼び出すスクリプトを指定して、スクリプトそのものは④のように別のファイルとして記述する方式だ。この方式であればHTMLとスクリプトは分離できるので、それぞれのファイルはわかりやすくなる。ただし、ファイルが別々になったぶんだけ、ページ数が増えるとそれだけ管理は面倒になる。

PHPは、CGIよりも記述が簡単で、SSIよりも管理が楽になる。これがPHPを使う最大のメリットだと言える。

PHP

HTMLファイルの中に直接スクリプトを書く形式。HTMLとスクリプトの部分がキレイに分かれているため、ソースは読みやすい。

```
<html>
<head><title> ~ </title></head>
<body>
あなたで
<?php
$filename = "count.dat";
$file = fopen($filename, "r");
?>
人目です。
```

この部分は通常のHTML

この部分がPHPのスクリプト

① count.php

CGI

スクリプトを直接実行する形式。ほとんどのCGIはPerlという言語で書かれる。HTMLの部分はスクリプトで書かなければならず、ソースは読みにくい。

```
#!/usr/local/bin/perl
print "Content-type: text/html\n\n";
print "<html>\n";
print "<head><title> ~ </title></head>\n";
print "<body>\n";
print "あなたで";
$filename = 'count.dat';
open(READ, $filename) || &error("Error");
print "人目です.\n";
```

HTMLもスクリプトとして記述する

② count.cgi

SSI

HTMLファイルからスクリプトを呼び出す形式。ソースは読みやすくなるが、ファイルが2つになってしまうので、管理がやや面倒になる。

```
<html>
<head><title> ~ </title></head>
<body>
あなたで
<!--#exec cmd="count.pl">
人目です。
```

外部のスクリプト(count.pl)を呼び出す

③ index.shtml

```
#!/usr/local/bin/perl
$filename = 'count.dat';
open(READ, $filename) || &error("Error");
```

スクリプトは別のファイルに記述する

④ count.pl

最新スクリプトを今日から使う

PHPの 実行環境を 手に入れよう

How to use PHP

① PHPは対応していない ウェブサーバーでは使えない

PHPが使えるのは、PHPがインストールされているウェブサーバーのみ。残念ながら、プロバイダーのホームページサービスでPHPの使えるところはほとんどない。

② オススメは無料サーバー

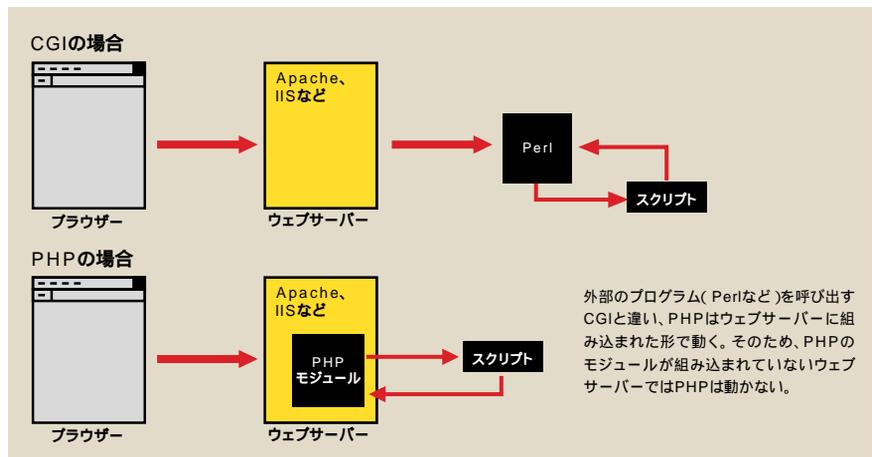
無料レンタルサーバーのなかには、PHPが使える業者もある。まずはこうしたサービスでPHPの使い方を覚えるのが簡単だ。

ホスティングサービスを使う

CGIと違い、PHPはウェブサーバーと一体化して動作するため、ウェブサーバーにPHPがインストールされていなければ、PHPは使えない。残念ながら、現在のところプロバイダーの個人ホームページサービスでは、PHPに対応しているところは皆無と言ってもいい状況だ。PHPを使いたい場合には、まずはPHPに対応したウェブサーバーを探さなければならない。

まず、候補として挙げられるのがホスティングサービスだ。PHPに対応しているホスティングサービスはまだあまり多くはないものの、下に示すように個人でも利用できる価格帯のサービスも増えてきている。

独自ドメインを使いたいという理由で、個人でもホスティングサービスを使うユーザーは増えている。現在、ホスティングサービスを検討しているなら、これからはディスク容量やCGIだけでなく、PHPにも対応しているかどうかをチェックしておこう。



PHPが使えるおもなホスティングサービス

名称	ホームページ	容量	料金	利用可能なSQL
AKIRA インターネット	www.akira.ne.jp	10Mバイト	月額 1,200円	MySQL、PostgreSQL
Kagoya Internet Routing	www.kagoya.net	100Mバイト	月額 2,500円	PostgreSQL (オプション料金・月額1,000円)
NSP Network Service	www.nspnet.ne.jp	30Mバイト	月額 3,000円	PostgreSQL
SPPD	www.sppd.ne.jp	10Mバイト	月額 1,000円	PostgreSQL (オプション料金・月額1,000円)
Webhosting.co.jp	www.webhosting.co.jp	200Mバイト	月額 3,900円	MySQL、mSQL
YI-WEB	www.yi-web.ne.jp	30Mバイト	月額 500円	PostgreSQL (ビジネスコース・月額1,980円)
アイル	home.isle.ne.jp	100Mバイト	月額 7,900円~	PostgreSQL
ペンギンインターネット	www.penguin.ne.jp	50Mバイト	年額 80,000円	PostgreSQL
ラビッドサイト	www.rapidsite.co.jp	175Mバイト	月額 10,000円	mSQL



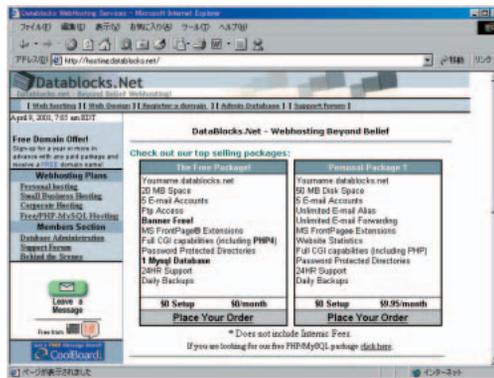
無料のレンタルサーバーを使う

さらに手軽にPHPを使ってみたいという人におすすめなのが、無料でホームページ用のディスクスペースを貸してもらえ、レンタルサーバーサービスだ。さすがにPHPが使える無料レンタルサーバーは数が少ないが、それでも下に挙げたようにいくつかの業者が無料でPHPの使えるサーバーを提供している。

もちろん、こうしたサービスの場合には、ホームページに「バナー」広告が出たり、定期的にアンケートに答えなければならぬといった制限は存在する。しかし、まずPHPがどのようなものかを試してみるには、無料で試せるこうしたサービスがうってつけだ。今回の記事を読んでPHPを実際に試してみたいと思ったら、まずはこうした無料サービスに申し込んでみることから始めよう。

自分でサーバーを立ち上げる

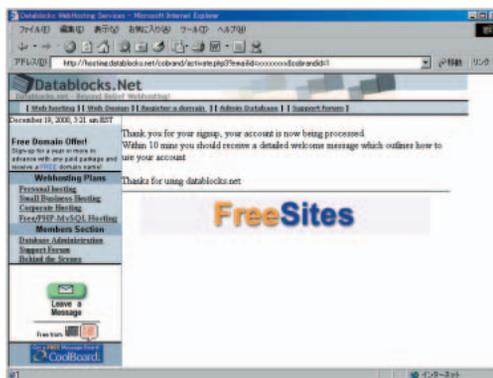
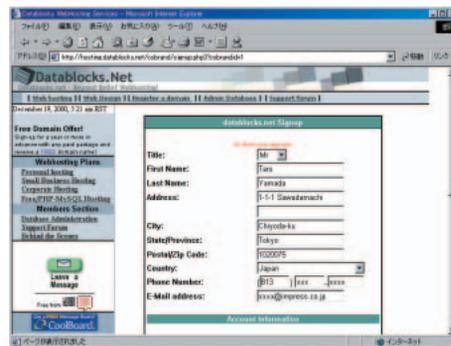
さらにPHPをじっくりと使ってみたいという人には、自分でPHP対応のサーバーを立ち上げてしまうのもいいだろう。インストールは少々面倒だが、ウィンドウズ98のような普通のパソコン用OSでもPHPを動かすことは可能だ。誰にでもおすすめできるわけではないが、挑戦してみたいという人は、172ページにインストール方法を解説したので、試してみしてほしい。



① 今回は無料レンタルサーバーのなかからDatablocksの利用方法を紹介する。ユーザー登録は、ホームページの「The Free Package!」欄の「Place Your Order」をクリックする。

② hosting.datablocks.net

④ 名前、住所、自分のメールアドレス、希望するユーザー名などを入力する。登録が完了すると、確認用のメールが届く。



③ ウェブブラウザで、届いたメールに記載されているURLにアクセスする。数十秒ほどして、左のような画面になれば登録はすべて終了だ。あとは、FTPサーバーやメールサーバーなどが記述されたメールが届くので、これを使ってホームページが作れる。

PHPが使えるおもな無料レンタルサーバー

名称	ホームページ	容量	ユーザーが利用できるURL	利用可能なSQL	備考
Spaceports	www.spaceports.com	無制限	**. spaceports.com / ユーザー名/	MySQL	月間転送量 20GBバイトまで
Datablocks	hosting.datablocks.net	20MBバイト	ユーザー名. datablocks.net	MySQL	
freedom2surf	www.freedom2surf.net	20MBバイト	www. ユーザー名.f2s.com	MySQL、PostgreSQL	
PORT5	www.port5.com	15MBバイト	ユーザー名. port5.com	MySQL	
MILLTO CLUB	www.millto.net	5MBバイト	**. millto.net / ユーザー名/	MySQL	日本語サイト

まずは基本的なスタイルを理解する

PHPを 使ってみよう

初級編

Script Guide

- ① **ファイルの拡張子は「.php」**
PHPのファイルは「index.php」のように、拡張子を「.php」とする。
- ② **スクリプトは「<?php ~ ?>」で囲む**
HTML部分はそのまま書いて、PHPスクリプトの部分を「<?php ~ ?>」で囲む。
- ③ **文字列の表示は「print」**
「print "~"」とすると、その文字列が表示される。

文字列を表示するスクリプト

それでは、さっそくPHPを実際に使ってみよう。まずは、もっとも簡単な文字列を表示するスクリプトだ。右の「print.php」が、実際のPHPのスクリプトファイルだ。PHPのファイルは「.php」という拡張子になる（サーバーによっては「.php3」の場合もある）。

このファイルを、FTPでサーバーにアップロードする。前ページで紹介した「Datablocks」の場合には、FTPサーバーは「ftp.datablocks.net」、アップロードするディレクトリーは「web」だ。アップロードが終わったら、ウェブブラウザで「http://ユーザー名.datablocks.net/print.php」にアクセスしてみよう。「PHPのテスト」と表示できれば成功だ。

```
<html>
<head><title>TEST print</title></head>

<body>

<?php
print "PHPのテスト";
?>

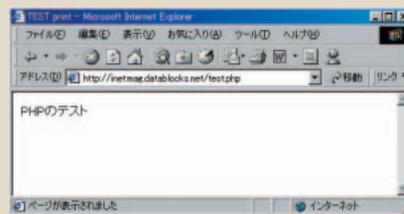
</body>
</html>
```

このあたりは通常のHTMLと同様

スクリプト開始
文字を表示するには、**print** ~ "と"
スクリプト終了

こども通常のHTMLと同様

print.php



インストールされているPHPの情報を見る

PHPでは「<?php」から「?>」の間がスクリプトとして解釈される。さきほどのスクリプトでは、文字列を表示する「print」を使ったが、これだけならわざわざPHPを使う必要もないし、あまりメリットも感じられないだろう。

そこで、もう少し派手な動作をするスクリプト「phpinfo.php」を試してみよう。さきほどのスクリプトとほとんど変わらないように見えるが、実際に表示させると驚くほど複雑なページになるだろう。ファイル中では「phpinfo();」になっている部分が、サーバー側でPHPのインフォメーションに置き換えられるため、このような表示になるのだ。

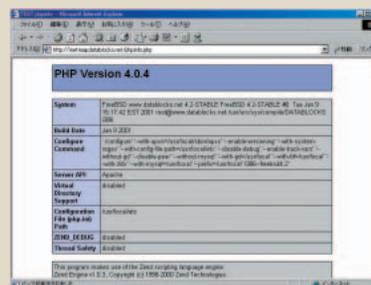
```
<html>
<head><title>TEST phpinfo</title></head>
<body>

<?php
phpinfo();
?>

</body>
</html>
```

この部分が、PHPのインフォメーションに置き換わる

phpinfo.php



① 文字化けした場合は

日本語のスクリプトが文字化けする場合には、PHPファイルの漢字コードをEUCに変換してみよう。それでもダメな場合には、日本語が使えないサーバーなので、残念ながらあきらめるしかない。



使用しているブラウザ名を表示する

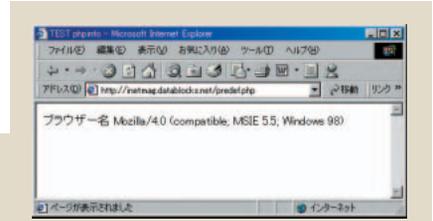
PHPを使えば、アクセスしてきたユーザーの環境も簡単に調べられる。右の「predef.php」は、アクセスしてきたユーザーのブラウザの種類を表示するスクリプトだ。こうしたユーザー情報をCGIで表示させるには面倒なプログラムが必要となるが、PHPであれば「\$_HTTP_USER_AGENT」という変数が用意されている。ここでは、この変数をprint文で表示しているだけの簡単なスクリプトになっている。

このほかにも、ユーザーのIPアドレスや、どのページからアクセスしてきたかを表す変数があるので(囲み参照)、「\$~」の部分を書き換えて試してみよう。

```
<html>
<head><title>TEST phpinfo</title></head>
<body>

<?php
print "ブラウザ名 $_HTTP_USER_AGENT";
?>

</body>
</html>
```



変数はprintのなかに直接書ける

predef.php

① その他の環境変数

「\$_HTTP_USER_AGENT」以外にも、以下のような環境変数が利用できる。

- HTTP_REFERER
リンク元のURLを表す環境変数。
どのページからアクセスしてきたかがわかる。
- REMOTE_ADDR
アクセスした人のIPアドレスを表す環境変数。

ブラウザの種類ごとに表示するページを変える

こうしたユーザー情報を使えば、ブラウザによって表示されるページを変えるという動作も簡単にできる。

右の「browser.php」がその例で、スクリプト全体は「if ~ elseif ~ else ~ endif」という構造になっていて、条件(ブラウザの種類)によって表示される内容が変化するようにしている。

注目してほしいのは、PHPのスクリプトが「<?php ~ ?>」という形で何回も登場している点だ。このように、ファイル全体はスクリプトとして解釈されるが、HTMLの部分はそのまま書けばいいというのが、PHPを使う大きなメリットである。

```
<html>
<head><title>TEST browser check</title></head>
<body>

<?php if(ereg("MSIE", $_HTTP_USER_AGENT)) : ?>
<h1>MSIE用のページ</h1>
MSIEでアクセスすると、ここが表示されます。<BR>
普通にHTMLで書けるので、とてもラクです。<BR>

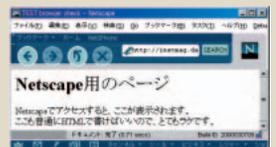
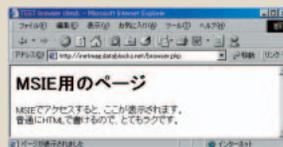
<?php elseif(ereg("Netscape", $_HTTP_USER_AGENT)) : ?>
<h1>Netscape用のページ</h1>
Netscapeでアクセスすると、ここが表示されます。<BR>
これも普通にHTMLで書けばいいので、とてもラクです。<BR>

<?php else : ?>
<h1>その他のページ</h1>
MSIE、Netscape以外でアクセスすると、ここが表示されます。<BR>
これも普通にHTMLで書けば大丈夫です。<BR>

<?php endif ; ?>

</body>
</html>
```

browser.php



① ブラウザーの判別方法

そのほかのブラウザを判別するには、スクリプトの「MSIE」や「Netscape」の部分を以下のように変えればよい。

- iモード「DoCoMo」
- J-PHONE「J-PHONE」
- ドリームキャスト「DreamPassport」
- ザウルス「sharp pda browser」

複雑な処理もCGIより簡単

PHPを 使ってみよう

中級編

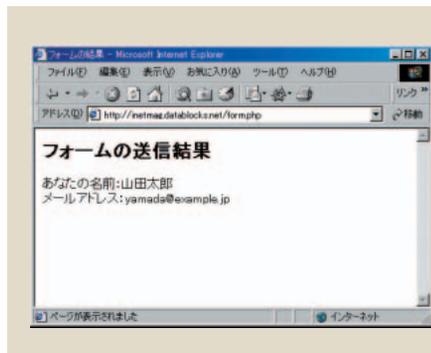
フォームの使い方をマスターする

フォームに入力された内容をもとに、表示される内容を変更するというのは、CGIでも基本的な動作だ。しかし、CGIでフォームを扱う際にはかなり複雑な手続きが必要となるのに対して、PHPではとても簡単にフォームの入力内容を参照できる。

まずは、フォームの入力を行う「form.html」を見てみよう。HTMLでは、<form>タグのなかでフォームを解釈するスクリプトを「action=" ~ "」として指定する。

呼び出されるスクリプトは「form.php」ようになる。PHPではフォームの「name」で指定した名前の変数に、入力された内容が格納されるようになっている。フォームで「name="username"」となっている項目は、「\$username」という変数で参照できるという仕組みだ。

これと同じことをPerlでやろうとすると、フォーム解析用のスクリプトが必要になり、かなり複雑になってしまう。PHPはホームページ用に開発された言語なので、特にこうした操作が簡単にできるのである。



```
<html>
<head><title> フォームの結果 </title></head>
<body>
<h2> フォームの送信結果 </h2>
<?php
print "あなたの名前: $username<br>¥n";
print "メールアドレス: $mailaddr<br>¥n";
?>
</body>
</html>
```

form.php

```
<html>
<head><title> フォームのテスト </title></head>
<body>
<h2> フォームの送信テスト </h2>
<form method="post" action="form.php">
名前
<input type="text" name="username" size="20"><br>
メールアドレス
<input type="text" name="mailaddr" size="40"><br>
<input type="submit" value="送信">
</form>
</body>
</html>
```

form.php を呼び出す

form.html

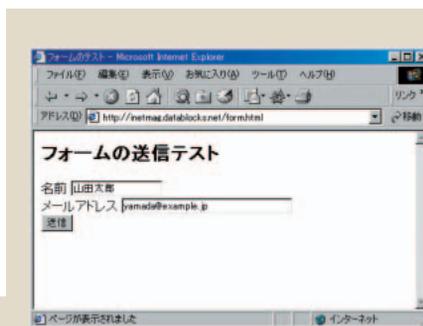
Script Guide

① フォームからの入力

フォームに入力された項目は、フォームタグの「name」で指定した名前と同じ変数名で参照できる。

② 外部のファイルを参照する

「include("ファイル名")」とすると、別のスクリプトファイルを読み込める。同じスクリプトを何度も利用する場合には、スクリプトを分けておくとう利だ。





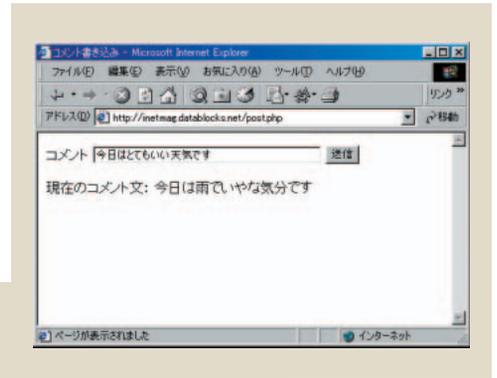
1行コメントを表示する

それでは、フォームに入力された内容をもとに、トップページにコメントを表示するスクリプトを作ってみよう。

まずはコメントを入力するスクリプト「post.php」を見てみよう。ここではフォームに入力された内容を、別のテキストファイルに保存する動作を行っている。このスクリプトでは、フォームを処理するスクリプトとして自分自身を呼び出して、コメント欄になにかが入力されている場合のみ、ファイルに書き込みを行うようになっている。

また、コメントを表示するスクリプトは「disp.php」として分離している。スクリプト内で別のスクリプトを呼び出す場合には、「include("ファイル名")」とすればよい。disp.phpは、コメントのファイルを読み込んでprint文で表示するスクリプトだ。

こうしてスクリプトを分離しているのは、実際にコメントを表示するページ(top.php)で再利用するためである。このように、同じスクリプトを何度も使う場合には、スクリプトを分けてincludeを使うのが便利だ。



```

<html>
<head><title> コメント書き込み </title></head>
<body>

<?php
if($comment != ""):
    $filename = "comment.txt";
    $file = fopen($filename, "w");
    fwrite($file, $comment);
    fclose($file);
endif;
?>

<form method="post" action="post.php">
コメント
<input type="text" name="comment" size="50">
<input type="submit" value="送信">
</form>

現在のコメント文: <?php include("disp.php"); ?>

</body>
</html>

```

書き込みの有無をチェック (\$comment)
コメントを記録するファイル名
ファイルを開く (書き込む場合は"w")
コメントを書き込む
ファイルを閉じる

自分自身を呼び出す

コメント文を表示

post.php

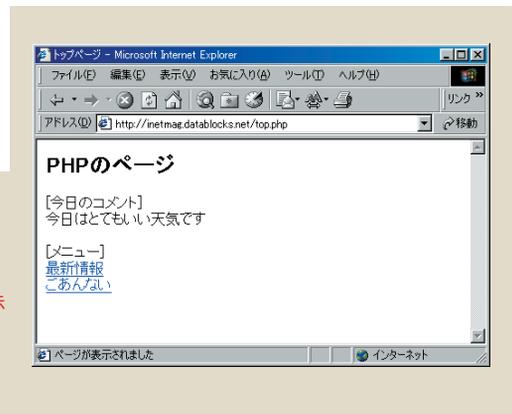
```

<?php
$filename = "comment.txt";
$file = fopen($filename, "r");
$data = fread($file, filesize($filename));
print $data;
fclose($file);
?>

```

コメントを記録するファイル名
ファイルを開く (読み込む場合は"r")
ファイルからデータを読み込む
データを表示する
ファイルを閉じる

disp.php



```

<html>
<head><title> トップページ </title></head>
<body>
<h2>PHPのページ</h2>
[今日のコメント]<br>
<?php include("disp.php"); ?><br>
<br>
[メニュー]<br>
<a href="whatsnew.php">最新情報</a><br>
<a href="info.php">ごあんない</a><br>

</body>
</html>

```

top.php

複雑な処理も CGI より簡単

PHPを 使ってみよう

上級編

アクセスカウンターを設置する

それでは、CGIでは定番のアクセスカウンターをPHPでも作ってみよう。かなり複雑になってきたが、この程度の行数の скриプトでアクセスカウンターが実現できるのも、PHPならではの特徴だ。

PHPで外部のファイルを扱う際には、だいたいこのスクリプトのような動作が基本的な流れになるので、ほかのスクリプトを作る際にも参考になるだろう。このスクリプトでは、カウンターのファイルから、アクセス数と前回アクセスのあったIPアドレスを読み込んで、前回のIPアドレスと今回のIPアドレスを比較して、異なっていれば数値に1を足して、再度ファイルに格納するという流れになっている。このスクリプトでは読み込んだファイルを上書きしているため、ファイルを開く「fopen」コマンドでは、ファイルのオープンモードに「r+」を指定している。

```
<html>
<head><title>アクセスカウンター</title></head>
<body>
<h2>アクセスカウンター</h2>
```

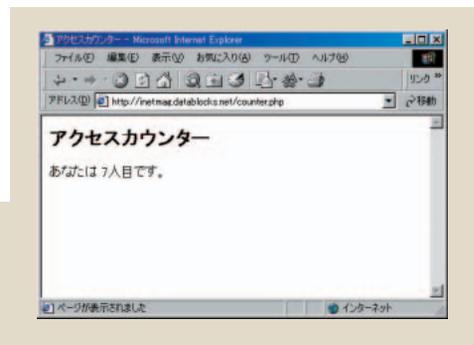
```
あなたは
<?php
$filename = "count.dat";
$file = fopen($filename,"r+");
flock($file, 2);
$count = chop(fgets($file, 20));
$prev = fgets($file, 20);
if($prev != $REMOTE_ADDR) $count++;
print $count;
rewind($file);
fwrite($file, "$count\n");
fwrite($file, $REMOTE_ADDR);
ftruncate($file, ftell($file));
flock($file, 3);
fclose($file);
?>
人目です。
</body>
</html>
```

counter.php

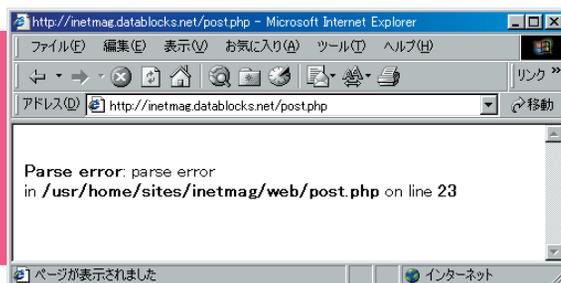
アクセス数を記録するファイル名
ファイルを「読み書き」モードで開く
ファイルロック開始
ファイルから1行目を読み込む (chop は行末の改行コードを取るため)
ファイルから2行目を読み込む
前回と違うIPアドレスならプラス1
アクセス数を表示する
ファイルの先頭にもどる
1行目にカウンターの数値を書き込む
2行目にIPアドレスを書き込む
ファイルの書き込みを終了する
ファイルロック終了
ファイルを閉じる

Script Guide

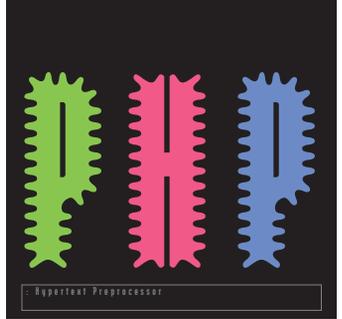
- ① ファイルの読み書き
ファイルの読み出しと書き込みを同時に行う場合には、fopenのモードを「r+」とする。
- ② データベースとの連携
PHPは、MySQLやPostgreSQLといったデータベースサーバーと組み合わせやすいように作られている。ただし、SQLサーバーの使い方はあまり簡単ではない。



① スクリプトがうまく動かないときは
PHPスクリプトに間違いがあった場合には、右のようなエラー画面が表示される。CGIの場合には単にエラーであるというメッセージしか出ないが、PHPの場合には上から何行目でエラーになったかが表示される。このため、PHPはスクリプトの間違いを修正しやすいというメリットもある。



PHPのエラーメッセージ。ここでは、上から23行目でエラーが出ていることがわかる。

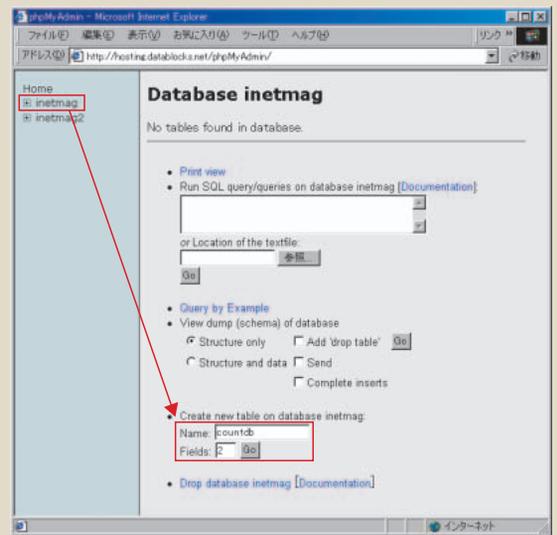


データベースと連携したスクリプトを作る

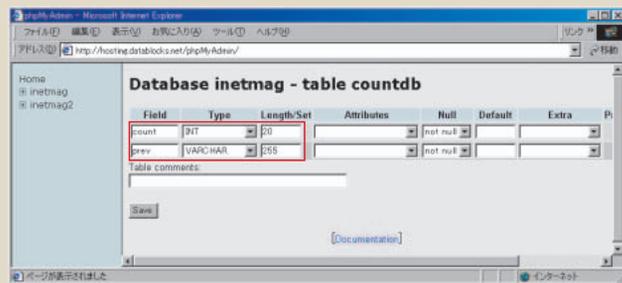
最後に、PHPの特徴であるデータベースと連携したスクリプトを紹介しよう。ここでは、無料でPHPが利用できるレンタルサーバー「Datablocks」での設置方法を紹介する。DatablocksではデータベースとしてMySQLが利用でき、MySQLの管理はウェブ上でできるようになっているので、図の①と②の操作を行ったのち、下の「counter2.php」を試してみよう。スクリプトの機能としては前ページのアクセスカウンターと同じものだが、データファイルの管理はMySQLで行っているのが異なる点である。

実際にSQLを扱うとなると専門の知識がかなり必要になるのが、PHPとSQLを組み合わせれば、ショッピングサイトのような複雑なサイトの構築にも対応できる。こうしたサイトの構築や管理を考えている人なら、迷わず今日からPHPを使い始めよう。個人ページから大規模サイトまで、PHPはあなたのウェブ作成のスキルを一段上のものしてくれるはずだ。

① Datablocksのトップページから「Admin Database」のリンク  をクリックする。ユーザー名とパスワードが要求されるので、FTPのアカウントと同じものを入力する。画面が表示されたら、右側のユーザー名をクリックして、右の画面のように入力して「Go」ボタンを押す。
<http://hosting.datablocks.net/phpMyAdmin/>



② データベースの編集画面になるので、左の画面のように各項目を埋めて、「Save」ボタンを押す。これでデータベースの準備は完了する。



```
<html>
<head><title> アクセスカウンター・2</title></head>
<body>
<h2> アクセスカウンター・2</h2>
```

```
あなたは MySQLのサーバー名
<?php
mysql_connect("localhost", "inetmag", "xxxxxxx");
mysql_select_db("countdb");
$result = mysql_query("select * from countdb");
$array = mysql_fetch_array($result);
$count = $array["count"];
$prev = $array["prev"];
if ($prev != $REMOTE_ADDR) $count++;
print $count;
$result = mysql_query("update countdb set count=$count, prev=$REMOTE_ADDR");
?>
人目です。
```

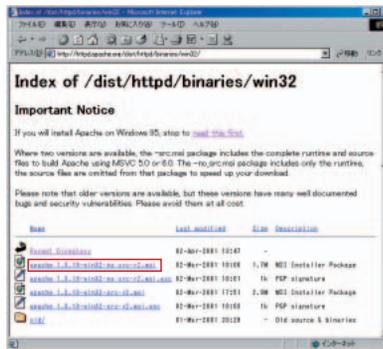
counter2.php

自宅のパソコンがサーバーになる

ウィンドウズに PHPを インストールしよう

Apacheのインストール

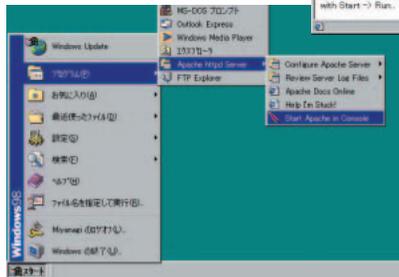
PHPでさらにいろいろと実験してみたいという人は、自分のパソコンにウェブサーバーとPHPをインストールしてしまうのがおすすめだ。ここでは、ウィンドウズマシンにウェブサーバー（Apache）とPHPをインストールする方法を紹介する。インストールの手順は少々面倒だが、これでスクリプトの動作確認はかなりラクになるはずだ。



① Apacheのダウンロード用のページから、「apache-1.3.19-win32-no_src-r2.msi」をダウンロードする（「1.3.19」の部分はバージョン番号なので、変わる可能性がある）。
<http://httpd.apache.org/dist/httpd/binaries/win32/>



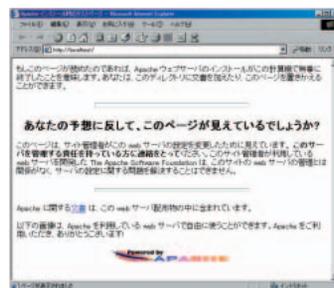
② ダウンロードしたファイルをダブルクリックすればインストールが開始される。



③ Apacheを動かすには、スタートメニューから「プログラム」「Apache httpd Server」「Start Apache in Console」を選ぶ。Apacheがきちんと動いている場合は、「Apache/1.3.19 (Win32) ~」と表示されたウィンドウが開く。Apacheを終了させる場合には、このウィンドウを閉じればよい。



④ インストールはウィザード形式なので、ほとんどは「Next」ボタンを押し続けていけば問題ない。ただし、ライセンスの内容を示す画面は、内容に同意のうえ、「I accept ~」をクリックしないと先に進めなしい。



⑤ Apacheが動いている状態で、ウェブブラウザのアドレス欄に「http://localhost/」と入力する。右のような画面が表示されていれば、インストールはすべて完了だ。

Install PHP

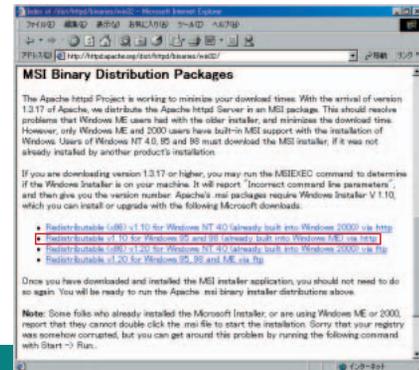
① ウェブサーバーは Apacheを使う

PHPを動かすには、ウェブサーバーのインストールが必要だ。ウェブサーバーには多くの種類があるが、PHPを使う場合にはもっとも対応が簡単なApacheをインストールしよう。

② PHPは手動でインストール

現在のウィンドウズ用のPHPには、インストール用のソフトは付属していない。ファイルの移動や設定はすべて手動で行うことになる。

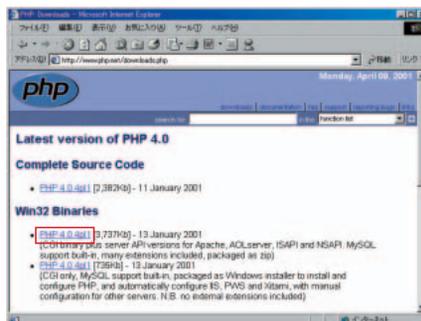
③ ダウンロードしたファイルが右のようなアイコンになっていない場合には、ページの下のほうにある「MSI Binary Distribution Packages」から、「v1.10」のファイルをダウンロードして、先にインストールしておく必要がある。





PHPのインストール

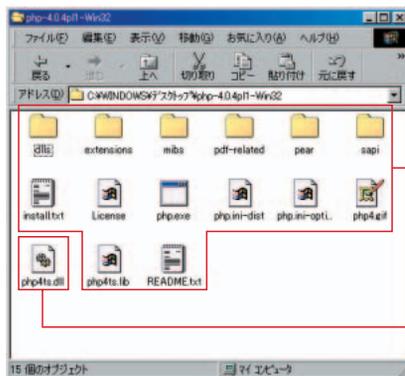
ウィンドウズ版のPHPにはインストーラーが付属していないので、解凍したファイルは自分でフォルダーを作成してそこに移動させる必要がある。あとは、Apacheの設定ファイルをPHPが利用できるように変更すれば準備は完了だ。



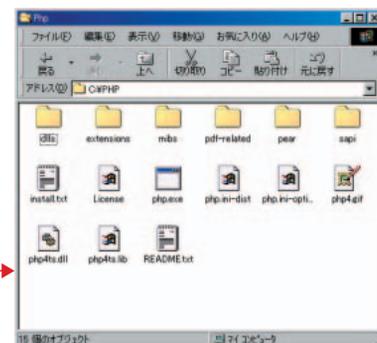
① PHPのダウンロード用ページで、「Win32 Binaries」と書かれているところにあるのが、ウィンドウズ用のPHPだ。2つのファイルがあるが、Apacheと組み合わせて使う場合には上のファイルをダウンロードする
www.php.net/downloads.php



② ダウンロードしたZIPファイルを解凍する。



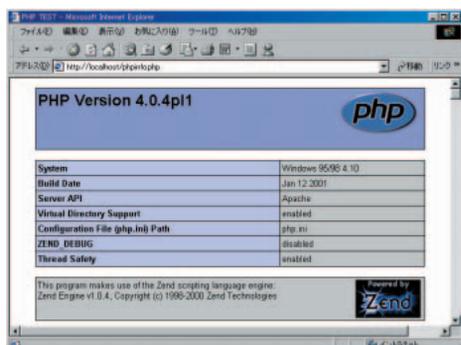
③ 解凍したファイルのうち、「php4ts.dll」というファイルを「C:\Windows\System」フォルダーに移動する。残りのファイルは、Cドライブに「PHP」というフォルダーを作り、そこにすべて移動する。



```
LoadModule php4_module c:/php/sapi/php4apache.dll
AddType application/x-httpd-php .php
```

PHPを使うには、この2行をhttpd.confの最後に追加する。

C:\Program Files\Apache Group\Apache\conf\httpd.conf



④ これで準備は完了。あとはHTMLファイルやPHPファイルを「C:\Program Files\Apache Group\Apache\htdocs」に置けば、ブラウザから「http://localhost/ファイル名」でアクセスできる。これまで作ってきたPHPファイルを置いて、実際に動くかどうかを確認しよう。

次号より PHP入門の連載開始!

今回の記事でPHPに興味を持った人のために、来月号からは「PHP入門」を連載で開始します。PHPの基礎知識からデータベースと連携した高度なテクニックまで、毎回実際のスクリプトを例題にして、PHPのノウハウを解説していきます。これまで、CGIは難しいからとあきらめていた人でも、PHPならきっと大丈夫。これからのウェブ作成には欠かせない最新スクリプト、PHPの新連載をどうぞお楽しみに。



[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