

PHP

PHP heißt PHP Hypertext Preprocessor und ist eine Skriptsprache. Sie eignet sich besonders gut dazu dynamische, interaktive Webseiten zu gestalten oder und damit verbunden auch Datenbankanfragen durchzuführen.

^^ Grundlagen ^^

Das wichtigste ist zu realisieren, dass PHP ein Serverscript ist, und, dass nur in Auszeichnungssprachen wie HTML eingebettet ist.

Was das heißt? Es bedeutet, dass das Script auf dem Server verarbeitet wird und der User eine fertige HTML Datei auf seinem Bildschirm erhält. Er sieht nichts mehr vom PHP Quelltext.

Damit dies auch in der Praxis funktioniert braucht man einen speziellen PHP Server.

PHP Dateien erkennt man grundlegend daran, dass sie die Endung '.php' aufweisen. Je nach Autor wird es unterschiedlich in das Dokument eingebunden.

1.	<code><?php echo '<h1>Überschrift</h1>'; ?></code>
2.	<code><? echo '<h1>Überschrift</h1>'; ?></code>
3.	<code><script language="php"> echo '<h1>Überschrift</h1>'; </script></code>
4.	<code><# echo '<h1>Überschrift</h1>'; #></code>

In jedem Fall sieht würde der vom Browser interpretierte Quelltext folgendermaßen aussehen:

```
test.php:
<h1>Überschrift</h1>
```

Ich persönlich bevorzuge die 2. Variante.

Ein Tip:

<?='<h1>Überschrift</h1>'?> ist eine Abkürzung für
<? echo '<h1>Überschrift</h1>'; ?>

^^ VARIABLEN ^^

Es gibt verschiedene Arten von Variablen:

Am einfachsten einzubinden sind Dokument-interne Variablen.
Beispiel:

```
<?
$name = "Torben Brodt";
$email = "mail@torbenbrodt.de";
?>

<html>
<head><title>www.easy-coding.de</title></head>
<body>
Schreib mir eine E-Mail: <br />
<a href="mailto:<?=$email?>"><?=$name?></a>
</body>
</html>
```

Wichtig und sehr nützlich ist es, Variablen über die URL weiterzugeben.
Allerdings werden diese anders ausgelesen.

Beispiel:

<http://www.easy-coding.de/battle.php?email=mail@torbenbrodt.de>

```
<html>
<head><title>www.easy-coding.de</title></head>
<body>
Schreib mir eine E-Mail:<br />
<a href="mailto:<?=$_GET[email]?>">Torben Brodt</a>
</body>
</html>
```

Mehrere Variablen gibt man unter Verwendung eines '&' weiter. Z.B. durch folgende URL

<http://easy-coding.de/battle.php?id=1&name=grafikbattle&bl.....>

Ein Formular gibt man folgendermaßen aus:

```
seite1.php
<html>
<head><title>www.easy-coding.de</title></head>
<body>
Geben sie im unteren Feld ihre Webmaster-ID ein:<br />
<form action="seite2php" method="post" target="_new">
<input type="text" name="webmaster_id" />
<input type="submit" />
</form>
</body>
</html>

seite2.php
<html>
<head><title>www.easy-coding.de</title></head>
<body>
Ihre Webmaster-ID lautet: <?=$_POST[webmaster_id]?>
</body>
</html>
```

Der Variablenname ergibt sich dabei aus dem Name des INPUT-Feldes.

Sollten sie die Variable abwechselnd aus einem POST-Formular oder einer GET-Url beziehen, dann benutzen sie folgende Syntax:

<code>\$_REQUEST[variablenname]</code>

^^ if & else – Anweisungen ^^

Ein absolutes Muss in Sachen PHP sind die if/else/elseif Anweisungen.

Hier die Vergleichsoperatoren:

==	Ist gleich
!=	Ist ungleich
>	Ist größer als
<	Ist kleiner als
>=	Ist größer gleich
<=	Ist kleiner gleich

Zur Anwendung:

```

<html>
<head><title>www.easy-coding.de</title></head>
<body>
<?
$achim=26;
$alfred=12;
if(!isset($_POST[submit]))
// Wenn die Variable SUBMIT noch nicht gePOSTet wurde, dann soll folgendes
ausgegeben werden
{
?>
Wer ist älter Achim oder Alfred?
<form action="index.php" method="post">
<input type="text" name="person" />
<input type="submit" />
</form>
<?
}
else {
// Wenn die Variable SUBMIT gepostet wurde, dann soll folgendes verarbeitet
werden
if($achim > $alfred)
{
$aelter = "Achim";
// Wenn die Variable ACHIM größer ist als die Variable ALFRED, dann ist die
Variable AELTER = Achim.
} elseif($achim < $alfred)
{
$aelter = "Alfred";
}
if($_POST[person]) == "$aelter")
// Wenn das gepostete Feld mit dem Namen PERSON den selben Inhalt hat, wie
die Variable AELTER, dann soll er folgenden Text ausgeben
{
echo '$aelter ist die korrekte Antwort.';
} else {
// Ist die Variable PERSON ungleich AELTER wird folgendes ausgegeben
echo '$aelter ist leider die falsche Antwort.';
}
}
}

```

```
?>
</body>
</html>
```

Andere Operatoren:

&&	und
	oder
!	Logisches Nicht
/ * - +	Mathematische Funktionen

Mit ' && ' und ' || ' lassen sich die Vergleichsoperatoren verknüpfen.

```
if(5 > 1 && 3 < 10)
{
    echo '5 ist größer als 1 UND 3 ist kleiner als 10';
}

if(5 > 1 && 3 > 10)
{
    echo 'entweder ist 5 größer als 1 ODER 3 ist größer als 10';
}
```

^^ SWITCH ^^

Die SWITCH-Anweisung ist gleichbedeutend einer Reihe von if- Anweisungen

```
http://www.easy-coding.de/geheim.php?variable=1

switch($_GET[variable])
{
    case 1:
        echo 'hinter eins, verbirgt sich codewort alpha';
        break;
    case 1:
        echo 'hinter zwei, verbirgt sich codewort beta';
        break;
    default:
        echo 'Sie müssen schon was tun, um etwas zu erhalten';
}
```

^^ Schleifen ^^

for-Schleifen werden verwendet, wenn dem Programmierer die Anzahl der Wiederholungen bekannt ist. Daher kann man auch von einer Zahlen-Schleife sprechen.

while-Schleifen sind Bedingungs Schleifen. Sie werden endlos wiederholt bis eine Bedingung eintrifft.

```
<html>
<head><title>www.easy-coding.de</title></head>
<body>
<?
for($i=1; $i<=5; $i++)
{
echo '$i<br />';
// Gibt die Zahlen 1 bis 5 untereinander aus
}
?>
</body>
</html>
```

2 andere Beispiele:

for(\$i=10; \$i<=20; \$i=\$i+2)	10,12,14,16,18,20
for(\$i=20; \$i>=10; \$i=\$i-2)	20,18,16,14,12,10

Auf die Verwendung von while-Schleifen werde ich später im Zusammenhang mit MySQL zurückkommen.

^^ Arrays ^^

```

<?
$a = array(
  "bmwz4" => array(
    "ps" => "200 PS",
    "0-100" => "4,5s",
    "preis" => "40.000 EUR"
  ),
  "opel_corsa" => array(
    "ps" => "50 PS",
    "0-100" => "22s",
    "preis" => "2.000 EUR"
  )
);
?>

<?=$a["bmwz4"]["preis"]?>
//gibt 40.000 EUR aus
<?=$a["opel_corsa"]["ps"]?>
//gibt 50 PS aus

```

^^ Wichtige Befehle ^^

eregi

```

<?
$i=0;
if(eregi($_POST[suchwort],$text)
{
  $i++;
}
?>

```

Das Suchwort hat <?=\$i?> Vorkommnisse im Text.

str_replace

```

<?
$comment = "Du bist ein Arschloch";
$comment = str_replace("Arschloch","XXX",$comment);
echo '$comment';
?>

```

phpinfo()

```

<?
phpinfo();
// Dieser Befehl spuckt alles mögliche über euren Server aus. Er zeigt auch die
erforderlichen Befehle an, um z.B. die Ip-Adresse zu ermitteln
?>

```

date

```
<?= date("d.m.Y G:i")?>  
// 07.08.2003 11:24  
// um einen durch time() produzierten timestamp auszulesen, benutzt man  
folgende Syntax  
<?  
$time = time();  
echo 'date("d.m.Y G:i",$time)'  
?>
```

include/ require

about.php

```
<table>
<tr>
  <td>Hobby</td>
  <td>Programmieren</td>
</tr>
</table>
```

index.php

```
<html>

<head><title>www.easy-coding.de</title></head>

<body>

<?
if($_GET[section]== "")
{
// Wenn der Variable section kein Wert zugeordnet ist, dann soll home.php
includet werden

  include("home.php");
} else {
  include("$_GET[section].php");
// der Wert section ist gleich dem Dateinamen + der Endung ' .php '
// wenn section=about ist, dann wird about.php includet
}
?>
</body>
</html>
```

Quelltext: index.php?section=about

```
<html>

<head><title>www.easy-coding.de</title></head>

<body>

<table>
<tr>
  <td>Hobby</td>
  <td>Programmieren</td>
</tr>
</table>

</body>
</html>
```

Legt euch erst mal eine Datei mit dem Namen **db.php** an. Diese includet ihr dann in jeder Datei, in der ihr eine Verbindung zur Datenbank herstellen wollt.

```
<?
$dbname=" ";
$dbhost=" ";
$dbuser=" ";
$dbpass=" ";

mysql_connect($dbhost,$dbuser,$dbpass);
mysql_select_db($dbname);
?>
```

Zu aller Anfang muss man sich eine Tabelle erstellen. Keine Angst, außer zur Installation wird man dies nicht mehr benötigen.

^^ NEWSCRIPT ^^

```

install.php
<?
include("db.php");

if($_POST[install] == "tables")
{
    mysql_query("CREATE TABLE news (
        // Eine Tabelle mit dem Namen news wird erstellt
        id int(11) auto_increment,
        // Ein Feld mit dem Namen ID und dem Typ INT wird erstellt
        titel varchar(255),
        beschreibung text,
        time varchar(20),
        PRIMARY KEY (id)
    ) ");
} else {
?>
<form action='<?=$_SERVER[PHP_SELF]?>' method='post'>
<input type='hidden' name='install' value='tables' />
<input type='submit' name='button' value=' Tabellen installieren '
</form>
<? } ?>

```

Deklariert man nach VARCHAR wird von links nach rechts gelesen. D.h. 19 ist größer als 20. Daher, verwendet man es nur für Text.

Zurück, zu dem Beispiel, in dem ich eine simple Newsdatenbank erstellt habe.

add.php

```
<? include ("db.php"); ?>
<html>
<head><title>www.easy-coding.de</title></head>
<body>
<?
$time = time();
// time() gibt die Sekunden seit der Beginn der UNIX Epoche aus
if (!isset($_POST[button])) {
?>
<form method="post" action="<?=$_SERVER[PHP_SELF]?>">
<input type="text" name="titel" /><br />
<textarea name="beschreibung"></textarea><br />
<input type="submit" name="button">
</form>
<?
} else {
mysql_query("INSERT INTO news(id,titel,beschreibung,time)
VALUES('','$_POST[titel]','$_POST[beschreibung]','$time')");
// Es wird eine neue Zeile erstellt
?>
Die Daten wurden eingetragen<br>
<a href="<?=$_SERVER[PHP_SELF]?>">neuer Eintrag</a><br />
<a href="index.php">zur Übersicht</a>
<?
}
?>

</body>
</html>
<? mysql_close(); ?>
```

index.php

```
<? include ("db.php"); ?>
<html>
<head><title>www.easy-coding.de</title></head>
<body>
<?
$result=mysql_query("SELECT * FROM news ORDER BY time DESC");
while($line=mysql_fetch_array($result))
// es werden alle Zeilen nach time absteigend sortiert ausgegeben
{
// alles was zwischen WHILE steht, wird so lange wiederholt bis die Bedingung
erfüllt ist / alles Informationen ausgelesen wurden
?>
<table>
<tr>
<td colspan="2" style="text-align:right"><a
href="edit.php?id=<?$line[id]=?>">edit</a> | <a
href="delete.php?id=<?$line[id]=?>">delete</a></td>
</tr>
<tr>
<td><?=$line[titel]?></td>
<td><?=$line[time]?></td>
</tr>
<tr>
<td colspan="2"><?=$line[beschreibung]?></td>
</tr>
</table><br />
<?
}
?>
</body>
</html>
<? mysql_close(); ?>
```

edit.php

```
<? include ("db.php"); ?>
<html>
<head><title>www.easy-coding.de</title></head>
<body>
<?
$result=mysql_query("SELECT * FROM news WHERE id = '$_GET[id]'");
$line=mysql_fetch_array($result);
if (!isset($_POST[button])) {
?>
<form method="post" action="<?=$_SERVER[PHP_SELF]?>">
<input type="hidden" name="id" value="<?=$_GET[id]?>" />
// damit die IP bei Update noch weiterverarbeitet wird, fließt sie hier erneut ins
Formular mit ein
<input type="text" name="titel" value="<?=$line[titel]?>" /><br />
<textarea name="beschreibung"><?=$line[beschreibung]?></textarea><br
/>
<input type="submit" name="button">
</form>
<?
} else {
mysql_query("UPDATE news SET titel='$_POST[titel]',
beschreibung='$_POST[beschreibung]' WHERE id='$_POST[id]'");
?>
Danke, Die Daten wurden geändert<br />
<a href="<?=$_SERVER[PHP_SELF]?>">neuer Eintrag</a><br />
<a href="index.php">zur Übersicht</a>
<?
}
?>

</body>
</html>
<? mysql_close(); ?>
```

```

delete.php
<? include ("db.php"); ?>
<html>
<head><title>www.easy-coding.de</title></head>
<body>
<?
if (!isset($_POST[button])) {
?>
<form method="post" action="<?=$_SERVER[PHP_SELF]?>">
<input type="hidden" name="id" value="<?=$_GET[id]?>" />
<input type="submit" name="button" value=" delete ">

</form>
<?
} else {
mysql_query("DELETE FROM news WHERE id='$_POST[id]");
?>
Danke, Die Daten wurden gelöscht<br />
<a href="<?=$_SERVER[PHP_SELF]?>">neuer Eintrag</a><br />
<a href="index.php">zur Übersicht</a>
<?
}
?>

</body>
</html>
<? mysql_close(); ?>

```

^^ BEISPIELE ^^

```

<?
$result=mysql_query("SELECT * FROM news);
$num = mysql_num_rows($result);
// mysql_num_rows zählt die Reihen
echo 'Es sind $num Newsbeiträge vorhanden';
?>

```

```

<table >
<?
$i=0;
$res = mysql_query("select * from toplist");
while($line = mysql_fetch_array($res))
{
    $i++;
    $division = $i/2;
    if(is_int($division))
    // prüft ob die Zahl eine ganze ist
    {
    // Wenn sie eine ganze ist, wird die Hintergrundfarbe schwarz gefärbt
    $var = "background-color:#000000;color:#ffffff";
} else {
// Andernfalls wird sie weiß gefärbt
    $var = "background-color:#ffffff;color:#000000";
}
?>
    <tr class="<?=$var?>">
        <td>rank: <?=$i?></td>
        <td ><?=$line[titel]?></td>
    </tr>
<?
}
?>
</table>

```