

Mailversand mit Hilfe von PHP

Im bestehendem Formular „**formular.html**“ muss man die nicht befriedigende Lösung mittels „mailto:“ ändern auf: action="formular1.php": (der Name kann auch anders lauten)

```
<h2>Formular</h2>
<form action="formular1.php" method="post">
<fieldset>
<legend>Benutzerdaten</legend>
```

Die Endung „.php“ weist auf ein PHP-Skript hin. PHP (Abkürzung für „PHP: Hypertext Preprocessor“) ist eine Scriptsprache mit einer an C bzw. C++ angelehnten Syntax, die hauptsächlich zur Erstellung von dynamischen Webseiten oder Webanwendungen verwendet wird. PHP ist eine Skriptsprache, deren Befehle direkt in den HTML-Quelltext geschrieben werden können.

Formulare werden unter Verwendung des HTML-<form>-Tags erstellt. Das <form>-Tag für jedes sinnvolle Formular sollte den Wert des action-Attributs setzen, das angibt, was der Browser mit dem Formular machen soll, wenn der Benutzer auf den Absenden-Button klickt. **Gibt man im action-Attribut einen Dateinamen an**, führt der Browser beim Absenden des Formulars stattdessen auf dem Server eine Anfrage nach dieser Ressource durch und schickt dabei die in das Formular eingetragenen Daten mit.

Hat das Skript, das die Formulardaten verarbeiten soll, den Namen formular1.php, muss das <form>-Tag, das es aus dem Formular adressiert, folgende Form haben:

```
<form action = "formular1.php" method = "post">
</form>
```

Info:

- Mit **action=""** werden die Daten weitergeleitet, wenn jemand auf den submit-Button drückt. Das Attribut <action> verweist auf das PHP-Auswertungsprogramm mit dem Namen „formular1.php“.
- Das **Attribut <methode>** verweist auf die Übermittlungsmethode <post>.

PHP-Script: formular1.php

Der PHP-Code steht im <body> immer in folgendem Code:

```
<?php ..... ?>
```

Damit man mit den Variablen leichter weiterarbeiten kann, übergibt man lieber die durch die Methode POST ausgelesenen Elemente an eine passende Variable:

```
$vorname = $_POST["vorname"];
$nachname = $_POST["nachname"];
$email = $_POST["email"];
```

echo: Mit „echo“ erstellt man in PHP eine Ausgabe.

mail(): Um mit PHP ein Mail zu senden, verwende die vorgefertigte Funktion „mail()“

Folgende drei Informationen werden von der Funktion mail() verlangt und müssen angegeben werden (verpflichtend):

- \$an die Empfängeradresse
- \$betreff die Betreff der Nachricht
- \$msg der Inhalt der Nachricht (message)

\n erzeugt einen Zeilenumbruch

Um Sätze zusammen zu ketten verwendet man **den Punkt**. Dieser steht im Code immer am Satzende. Davor ist ein Leerschritt und ein Anführungszeichen, danach der Zeilenumbruch und wieder ein Anführungszeichen, gefolgt vom Text.

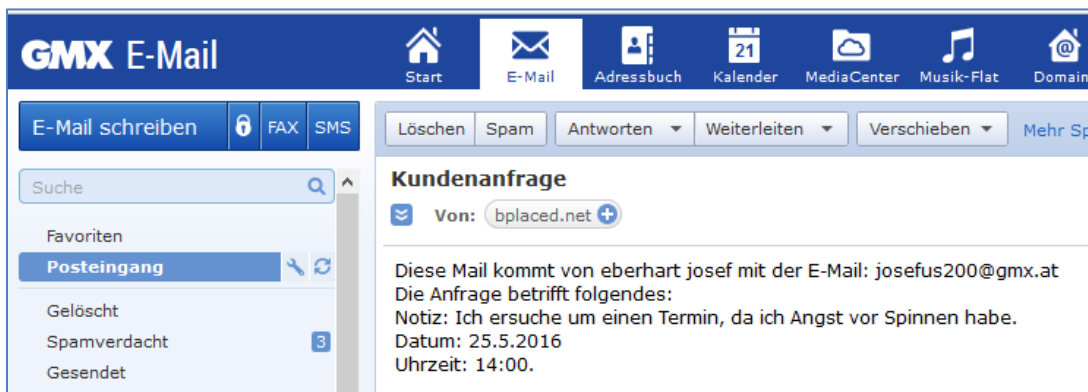
Beispiel:

```
echo "<h2>Guten Tag, $vorname $nachname \n </h2>" .  
    "Wir haben die E-Mail gesendet an josefeberhart@gmx.at.";
```

Damit wir eine Antwort für den Nutzer erstellt, damit er weiß, dass er erfolgreich ein Mail versendet hat.



Das Mail an den Administrator sieht folgendermaßen aus:



Code-Beispiel:

```
1 <!doctype html>
2 <html>
3 <head>
4 <meta charset="utf-8">
5 <title>PHP-Antwort</title>
6 </head>
7 <body>
8 <?php
9 $vorname = $_POST["vorname"];
10 $nachname = $_POST["nachname"];
11 $notiz = $_POST["notiz"];
12 $email = $_POST["email"];
13 $datum = $_POST["datum"];
14 $uhr = $_POST["uhr"];
15 echo "<h2>Guten Tag, $vorname $nachname \n </h2>" .
16     "Wir haben die E-Mail gesendet an josefeberhart@gmx.at.";
17
18 $an = "josefeberhart@gmx.at";
19 $betreff = "Kundenanfrage";
20 $msg = "Diese Mail kommt von $vorname $nachname mit der E-Mail: $email \n" .
21     "Die Anfrage betrifft folgendes: \n" .
22     "Notiz: $notiz \n" .
23     "Datum: $datum \n" .
24     "Uhrzeit: $uhr.";
25 mail($an, $betreff, $msg);
26 ?>
27 </body>
28 </html>
```