

PHP: Einfache Formulare

ChatGPT erklärt: Wozu braucht man Formulare?



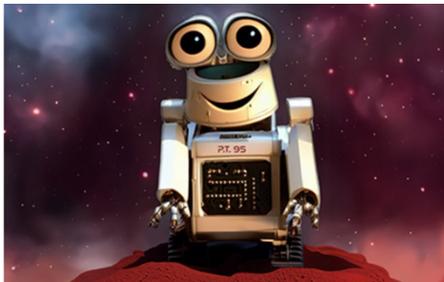
Formulare im Internet werden verwendet, damit Menschen Informationen auf Websites eingeben können. Das ist nützlich, wenn ...

1. Kontakt aufgenommen werden soll: Wenn du eine Nachricht an eine Website oder ein Unternehmen senden möchtest, z.B. eine Beschwerde.
2. Sachen bestellt werden sollen: Wenn du online einkaufen möchtest, verwendest du ein Formular, um deine Adresse und deine Zahlungsinformationen einzugeben.
3. Du Dich irgendwo anmelden möchtest: Wenn du dich auf einer Website registrieren möchtest, gibst du Informationen wie deinen Benutzernamen und dein Passwort in ein Anmeldeformular ein.

Unser Ziel

Wir benötigen eine Formular mit einem Textfeld, in das ein Name eingegeben werden kann. Dieser Name soll dann vom Webserver verarbeitet werden.

HTML-Formular **willkommen.html**



Willst du mit mir chatten?

Gib einfach hier deinen Namen ein:



PHP-Datei **auswertung.php**



Super!

Hallo Detlef!
Herzlich willkommen.

Ich würde auch gerne Detlef heißen ...

Wie erstellt man ein Formular?

Formulare werden in **HTML** entworfen. Die **Daten** des Formulars werden dann **an eine php-Seite geschickt**, die diese Daten auswertet.



Ein **einfaches HTML-Formular** besteht aus:

1. Einem `<form>` Tag, das das gesamte Formular umschließt.
2. Input-Feldern, in denen die Benutzer Informationen eingeben können, z.B. `<input>` für kurze Antworten und `<textarea>` für längere Texte.
3. Einem "Absenden"-Button, normalerweise mit `<input type='submit'>`.

Beispiel

```
<!DOCTYPE html>
<html lang="de">

<head>
  <title>Einfaches Formular</title>
</head>

<body>

  <h2>Kontaktformular</h2>

  <form action="auswertung.php" method="POST">
    <input type="text" name="tfName">
    <br>
    <input type="submit" name="btAbschicken"
value="Abschicken">
  </form>

</body>

</html>
```

Die eingegebenen Daten werden an diese Webseite geschickt!

Info:

method gibt an, wie die Daten abgeschickt werden. Bei **POST** werden die Daten verborgen (was insbesondere bei Passwörtern extrem wichtig ist!). Wenn man **GET** schreibt, kann man die Daten in der Adresszeile des Browsers lesen.

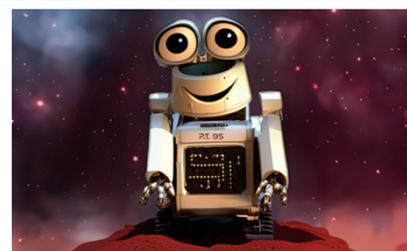
Formularelemente haben ein Attribut **name**. Das bekommt einen möglichst verständlichen Wert. Buttons bekommen zu Beginn ein **bt** (etwa **btAbschicken**), Textfelder ein **tf** (etwa **tfName**).

value ist der Wert, den das Element hat. Würde man beim Inputtype **text** einen value vergeben, dann ... Ach, probieren Sie das doch einfach selbst aus.

Aufgabe



Erstellen Sie die Webseite **willkommen.html**! Die Daten dieser Webseite sollen an **auswertung.php** geschickt werden.



Willst du mit mir chatten?

Gib einfach hier deinen Namen ein:

Wie kommt die php-Seite an die abgeschickten Daten?

Hier ein **sehr vereinfachtes** Beispiel, wie die Daten ausgewertet werden können, die an den Webserver (genauer: die php-Seite auf dem Webserver) geschickt wurden:



Beispiel

```
<html>
<head>
  <title>Anmeldung</title>
</head>
<body>
<?php
$name = $_POST['tfName'];
echo "Name: ";
echo "<br>";
echo $name;
?>
</body>
</html>
```

Der php-Code ist hier einfach in den <body> einer HTML-Seite eingebettet. Dadurch ist es möglich, die Seite mit HTML „hübsch“ zu machen!

Info:
\$_POST ist ein Array und enthält alle Werte aus dem Formular. \$_POST['tfName'] enthält also den Wert aus dem Eingabefeld mit dem Namen **tfName**.

Mit **\$name** deklarieren wir eine Variable, in der wir diesen Wert abspeichern. Den können wir dann später verwenden.



Anführungszeichen

Wir können in HTML und PHP nach unserem Belieben einfache Anführungszeichen (Hochkommata) oder doppelte verwenden. Beides geht:

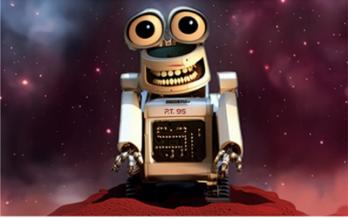
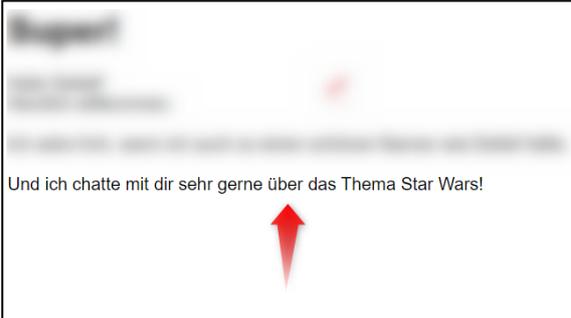
```
echo 'Name: ' ;
echo "Name:" ;
```

Wir machen das so:

- In HTML verwenden wir immer doppelte Anführungszeichen, also etwa
`<input type="text" name="tfText">`
- In PHP verwenden wir grundsätzlich einfache Anführungszeichen (Hochkommata); wenn etwas ausgegeben wird, verwenden wir doppelte:
`$name = $_POST['tfName'];`
`echo "Dein Name ist: " . $name;`

Wenn das mal schief geht, ist das kein Drama!

Aufgabe

	<p>Erstellen Sie die Webseite <code>auswertung.php</code>, die die empfangenen Daten auswertet!</p> <p><u>Tipp:</u> Im <code><body></code> Tag können Sie die Seite gestalten, wie Sie es gewohnt sind: Also z.B. das Bild mit dem <code></code> Tag einfügen. Erst dann, wenn Sie die empfangenen Daten brauchen (hier: <code>\$_name</code>), müssen Sie einen php-Bereich einfügen!</p>	 <p>Super!</p> <p>Hallo Detlef! Herzlich willkommen. Ich würde auch gerne Detlef heißen ...</p>
	<p>Kopieren Sie die Seite <code>willkommen.html</code> und benennen Sie die Kopie um in <code>willkommen_2.php</code>! Diese Seite schickt die eingegebenen Daten an <code>auswertung_2.php</code>!</p> <p>Erweitern Sie das Formular wie im Screenshot erkennbar.</p> <p>Nach dem Abschicken soll die Ausgabe so aussehen:</p>	 <p>Willst du mit mir chatten?</p> <p>Gib einfach hier deinen Namen ein: <input type="text"/></p> <p>Über welches Thema willst du chatten? <input type="text"/></p> <p><input type="button" value="Abschicken"/></p>  <p>Und ich chatte mit dir sehr gerne über das Thema Star Wars!</p>