

Formulare in HTML

In Formularen kann der „Besucher“ einer Homepage Informationen über die Tastatur oder die Maus eingeben, die der Anbieter der Homepage dann auswerten kann.

Formulare können aus Textfeldern (Text- oder Edit-Boxen), Optionsfeldern (RadioButtons), Kontrollfeldern (CheckBoxen) und Befehlsschaltflächen (Buttons) bestehen.

Ein Formular wird eingeleitet durch den Tag `<form>`. Innerhalb dieses einleitenden Tags müssen zwei Angaben gemacht werden.

Der Anfang jeder HTML 5 Seite deren Formatierung auf einer externen CSS-Datei beruht, sollte im Wesentlichen folgende Elemente enthalten. (Pfad- sowie Dateinamen natürlich angepasst!)

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link href="css/style.css" rel="stylesheet" type = "text/css">
  </head>
  .
  .
</html>
```

I. method = (“post“ oder “get“)

Das Attribut `method` bestimmt, nach welcher HTTP-Übertragungsmethode die Formulardaten an ihr Ziel gelangen. Bei „post“ gelangen die Formulardaten über den Standardeingabekanal an den Webserver. Bei „get“ werden die Formulardaten als Parameter an die Aufrufadresse angehängt.

II. action = (“Wohin die Daten des Formulars geschickt werden sollen“)

Angabe des Pfades eines CGI-Skriptes auf dem Web-Server (früher: einer Email-Adresse)

Ein Formular wird mit dem üblichen Ende Tag `</form>` abgeschlossen.

Beispiel: `<form method = “post“ action = php-skripte/resond.php>`

HTML-Tags für die meisten Steuerelemente eines Formulars

Die meisten Formularelemente (mit Ausnahme eines Listenfeldes und eines mehrzeiligen Textfeldes) werden durch den `INPUT`-Tag definiert! Um welches Steuerelement es sich letztlich handelt wird durch das Attribut `TYPE` bestimmt.

- | | |
|--|---|
| ▪ <code><input type = “text“></code> | Einzeiliges Textfeld |
| ▪ <code><input type = “date“></code> | Datumfeld |
| ▪ <code><input type = “time“></code> | Zeitfeld |
| ▪ <code><input type = “email“></code> | E-Mailfeld, validiert ob Eingabe einer E-Mailadresse entspricht. |
| ▪ <code><input type = “password“></code> | Passwordfeld |
| ▪ <code><input type = “number“></code> | Feld für die Eingabe einer Zahl mit Möglichkeit der Eingrenzung |
| ▪ <code><input type = “checkbox“></code> | Kontrollfeld |
| ▪ <code><input type = “radio“></code> | Optionsfeld |
| ▪ <code><input type = “button“></code> | Button-Elemente sind erst in Verbindung mit <u>Java-Script</u> sinnvoll, da sie in HTML selbst keine Funktion haben. |
| ▪ <code><input type = “submit“></code> | Spezielle Befehlsschaltfläche. Durch Anklicken werden die im Formular eingetragenen Informationen an den im <code>form</code> -Tag spezifizierten Server geschickt. |
| ▪ <code><input type = “reset“></code> | Spezielle Befehlsschaltfläche. Durch Anklicken werden alle gemachten Eintragungen im Formular gelöscht. |

Textfelder:

Einzeilige Textfelder: (TextBox)

Weitere Auswahlmöglichkeit(en) des Attributs `type` stehen zur Verfügung.

`type = “password“` Eingaben erfolgen verdeckt.

Neben dem Attribut `type` können noch weitere Attribute beim `input`-Tag des einzeiligen Textfeldes angegeben werden:

- `name` = "eindeutiger Name des Textfeldes im Formular"
- `size` = positive Ganzzahl, angezeigte Feldlänge
- `maxlength` = positive Ganzzahl, interne Feldlänge

Beispiel:

Name: `
 <input type = "text" name = "name" size = 30 maxlength = 50>`

Name:

Hinweis: Das **name-Attribut** muss immer vorhanden sein, da der Web-Server dadurch eindeutig die Inhalte identifizieren kann.

Mehrzeilige Textfelder: (Memo- oder RichEdit-Feld)

Mehrzeilige Textfelder eignen sich gut für ausführlichere Bemerkungen, Kritiken, Anregungen, etc. Sie werden durch den Tag `<textarea>` eingeleitet. Die Größe wird durch die Anzahl der Zeilen und Spalten, die das mehrzeilige Textfeld besitzen soll, festgelegt.

Beispiel: `<textarea name = "Anregungen" rows = 10 cols = 40></textarea>`

Damit definiert man ein 10-zeiliges und 50-spaltiges Textfeld.

Kontrollfelder:

Ein Kontrollfeld kann ebenso wie ein Optionsfeld mit der Maus selektiert werden. Auswahl mehrere Kontrollkästchen einer Gruppe ist möglich! Kontrollfelder besitzen neben dem festgelegten Attribut `type = "checkbox"` weitere mögliche Attribute beim `input`-Tag.

- `name` = "Name ohne Leerzeichen und deutschen Umlaute "
 Bemerkung: Name muss nicht eindeutig sein → Gadgets mit gleichem Namen bilden eine Gruppe
- `value` = "Wert als String"
- `checked` Vorauswahl des Kontrollkästchens

Die Attribute *name* und *value* bestimmen, welcher Name und welcher Wert dem Server übermittelt wird, falls das Kontrollkästchen beim abschicken des Formulars selektiert wurde.

Beispiel:

`<input type = "checkbox" name = "Angabenueberpruefung" value="ok">` Ja, die gemachten Angaben sind korrekt und können abgesandt werden.

☒ Ja, die gemachten Angaben sind korrekt und können abgesandt werden!

Optionsfelder:

Optionsfelder besitzen die besondere Eigenschaft, daß immer nur eines einer Gruppe aktiv sein kann. Die Auswahl eines anderen Optionsfeldes bewirkt automatisch das Rücksetzen der anderen Felder der Gruppe.

Beispiel: `<input type = "radio" name = "versand" value = "Post">`Post

Versenden Sie die Ware mit:

☐ DPD

☐ UPS

☒ Post

Das Optionsfelder zur selben Gruppe gehören wird dadurch bestimmt indem ihr **name-Attribut** identisch ist.

Listenfelder:

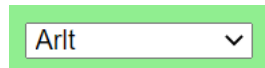
Das Erzeugen eines Listenfeldes erfolgt mit dem select-Tag. Innerhalb des `<select>` Tags sind die Attribute *name*, *value*, *size* und *multiple* möglich. *name* und *value* werden wie unter Kontrollfelder bereits beschrieben, verwendet. Die Angabe *size* = (wert) legt fest, wie viele Einträge der Liste für den Benutzer sichtbar sind. Per Voreinstellung kann immer nur ein Eintrag ausgewählt werden. Eine Mehrfachauswahl ist durch den Eintrag von *multiple* möglich.

Jeder einzelne Eintrag in einem Listenfeld wird durch den Tag `<option>` bestimmt. Direkt im Anschluss trägt man den Text ein. Soll ein bestimmter Eintrag im Anzeigefeld erscheinen muss zusätzlich im option-Tag der Eintrag *selected* erfolgen.

Beispiel eines Listenfeldes:

```
<label for="computerhaendler">Computerhaendler auswaehlen:</label>
<select name = "Computerhaendler" id="computerhaendler" size = 1>
  <option>Media Markt</option>
  <option selected>Arlt</option>
  <option>Saturn</option>
  <option>Elsaesser</option>
</select>
```

Computerhändler auswählen:



Arlt ist dabei vorausgewählt durch `<option selected>Arlt</option>`

Befehlsschaltflächen:

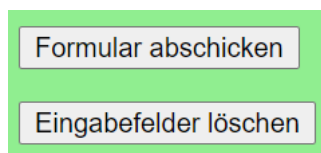
- Zum Übertragen der im Formular gemachten Eintragungen

```
<input type = "submit" value = "Formular abschicken">
```

- Zum Löschen der im Formular gemachten Eintragungen

```
< input type = "reset" value = "Eingabefelder löschen" >
```

Beispiel:



Daten gruppieren:

```
<fieldset>
  <legend>Ihre Daten</legend>
  <label for="name">Name</label><br />
  <input type="text" name="name" value="" id="name" size="50" maxlength="80">
  .
  .
</fieldset>
```

