

Tipi di input widget

Textbox

```
<form action="action_page.php">
```

```
<b>Name:</b>
```

```
<input type="text" name="name" size="20" maxlength="40" >
```

```
<br><br>
```

```
<input type="submit" >
```

```
</form>
```

Tipi di input widget

Menu a tendina (scelta esclusiva)



```
<form action="action_page.php">
  <select name="cars">
    <option value="volvo">Volvo</option>
    <option value="saab">Saab</option>
    <option value="fiat">Fiat</option>
    <option value="audi">Audi</option>
  </select>
  <br><br>
  <input type="submit">
</form>
```

Tipi di input widget

Checkbox (elenco di possibili scelte non mutuamente esclusive)

```
<form action="action_page.php">
```

```
<input type="checkbox" name="vehicle[]" value="Bike">I have a bike  
<br>
```

```
<input type="checkbox" name="vehicle[]" value="Car">I have a car  
<br>
```

```
<input type="submit">
```

```
</form>
```

```
<html>           // Crea un form che invia un array al server
<head>
  <title>Tell Us About Yourself</title>
</head>
<body>
<form action = "handle_about.php" method="post">
<fieldset>
  <legend>Enter your information in the form below:</legend>
  <p>
<b>Name:</b> <input type="text" name="name" size="20" maxlength="40"/>
  </p>
  <p>
<b>Interests:</b>
  <input type="checkbox" name="interests[]" value="Music" /> Music
  <input type="checkbox" name="interests[]" value="Movies" /> Movies
  <input type="checkbox" name="interests[]" value="Books" /> Books
  <input type="checkbox" name="interests[]" value="Skiing" /> Skiing
  <input type="checkbox" name="interests[]" value="Napping" /> Napping
  </p>
</fieldset>
<div align="center">
<input type="submit" name="submit" value="Submit My Information" />
</div>
</form>

</body> </html>
```

E' possibile evitare di inserire «manualmente» una checkbox per ogni opzione che si vuole elencare utilizzando PHP: si definisce un array con le opzioni fra cui si vuole scegliere e poi un ciclo foreach per generare la sequenza di checkbox.

E' infatti possibile scrivere script che mescolano HTML e PHP inserendo le sezioni PHP in un tag

```
<?php  
    codice  
?>
```

Per essere sicuri che l'interprete PHP venga attivato in corrispondenza delle sezioni PHP il file che si crea dovrà avere estensione .php .

```
<html>           // Crea un form che invia un array al server (VERSIONE PHP)
<head>
    <title>Tell Us About Yourself</title>
</head>
<?php $hobbies=array('Music','Movies','Books','Skiing','Napping'); ?>
<body>
<form action = "handle_about.php" method="post"> 
<fieldset>
    <legend>Enter your information in the form below:</legend>
    <p>
<b>Name:</b> <input type="text" name="name" size="20" maxlength="40"/>
    </p>
    <p>
<b>Interests:</b>
<?php foreach($hobbies as $value) {
    echo '<input type="checkbox" name="interests[]" ' . " value=$value>
$value <br>";
}
?> </p>
</fieldset>
<div align="center">
<input type="submit" name="submit" value="Submit My Information" />
</div>
</form>
</body> </html>
```

// handle_about.php

```
<?php
// If everything was filled out, print the message.
if ($_POST['name'] && $_POST['interests'])
{
    $name=$_POST['name'];
    echo"Thank you, <b>$name</b>. You entered your interests
as:<ul>";

    // Print each interest.

    foreach ($_POST['interests'] as $value)
    { // Loop through each.
        echo "<li>$value</li>\n";
    }

    Echo '</ul>'; // Close the list.
}
else
{ // One form element was not filled out properly.
    echo '<p><font color="red">Please go back and fill out the form
again. </font></p>';
}
?>
```

Ordinamento di array

```
$names = array ('George Michael', 'Ann', 'Buster');
```

```
sort($names);    ordina i valori eliminando la correlazione con le chiavi  
rsort($names);  ordinamento inverso
```

```
asort($names);  ordina i valori mantenendo la correlazione con le chiavi  
arsort($names); ordinamento inverso
```

```
ksort($names);   ordina in base alle chiavi  
krsort($names);  ordinamento inverso
```

```
// Create the array.
$movies = array (
10 => 'Casablanca',
9 => 'To Kill a Mockingbird',
2 => 'The English Patient',
8 => 'Sideways',
7 => 'Donnie Darko'
);

// Display the movies in their original order.
echo '<p>In their original order:<br /><pre>Rating  Title';
foreach ($movies as $key => $value) {
    echo "$key\t$value\n"; }

echo '</pre></p>';

// Display the movies sorted by title.
echo '<p>Sorted by title:<br /><pre>Rating  Title';
asort($movies);
foreach ($movies as $key => $value) {
    echo "$key\t$value\n"; }
echo '</pre></p>';

// Display the movies sorted by rating.
echo '<p>Sorted by rating:<br /><pre>Rating  Title';
krsort($movies);
foreach ($movies as $key => $value) {
    echo "$key\t$value\n";
}
echo '</pre></p>';
```



Alcuni esempi tratti da:

<http://www.w3schools.com/html/>

Tutorial con esercizi e simulatore interattivo.

Esercizio 1

Realizzare una pagina HTML che contenga un form che consenta all'utente di inserire una data (giorno mese anno ora) mediante quattro menu a tendina, attivando poi uno script PHP che fornisca un messaggio di conferma del tipo: "Il suo appuntamento è fissato per il giorno alle ore ...".

Provare ad usare sia il metodo GET che il metodo POST e osservare le differenze.

Esercizio 1b

Realizzare uno script in HTML/PHP che contenga un form che consenta all'utente di inserire una data (giorno mese anno ora) mediante quattro menu a tendina i cui valori sono definiti all'interno di un array, attivando poi uno script PHP che fornisca un messaggio di conferma del tipo: "Il suo appuntamento è fissato per il giorno alle ore ...".

Provare ad usare sia il metodo GET che il metodo POST e osservare le differenze.

Esercizio 1c

Usando il metodo GET, richiamare lo script PHP che dà conferma dell'appuntamento introducendo a mano la URL con i parametri della richiesta-