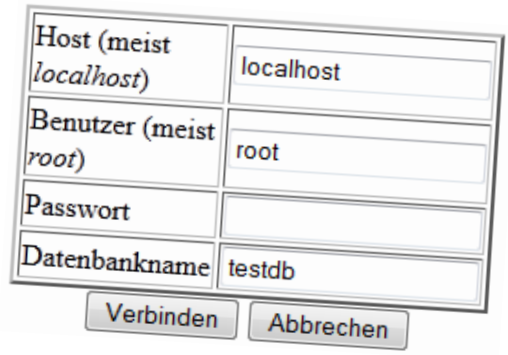
Formulargesteuert mit einer Datenbank verbinden

```
1  <html>
2  <head>
3  <title>Datenbank 1</title>
4  </head>
5  <body>
6  <form name="frmDBVerbindung" method="post" action="mysqlI_1Datenbankverbindung.php">
7  <table border="2" width="80&" align="center">
8  <tr><td>Host (meist <i>localhost</i>)</td><td><input name="host"type="text"
9         value="localhost"></td></tr>
10 <tr><td>Benutzer (meist <i>root</i>)</td><td><input name="benutzer"type="text"
11        value="root"></td></tr>
12 <tr><td>Passwort</td><td><input name="passwort"type="password"></td></tr>
13 <tr><td>Datenbankname</td><td><input name="datenbank" type="text" value="testdb"></td></tr>
14 </table>
15 <center><input type="submit" name="abschicken" value="Verbinden">
16 <input type="reset" name="abbrechen" value="Abbrechen"><br>
17 <?php
18        //Überprüfen ob das Formular abgeschickt wurde
19        //oder ob das Formular erstmals aufgerufen wurde
20        if (empty($_POST)) {exit;}
21        //Formulardaten einlesen
22        @$HOST=$_POST["host"];
23        @$BNAME=$_POST["benutzer"];
24        @$PWORT=$_POST["passwort"];
25        @$DBANK=$_POST["datenbank"];
26        // Verbindung mit MySQL aufnehmen; hier ohne Passwort
27        $verbindung=@mysql_connect($HOST,$BNAME,$PWORT)
28        or die('MSQL-VERBINDUNG FEHLGESCHLAGEN');
29        // Datenbank auswählen
30        @mysql_select_db($DBANK)
31        or die('DATENBANKVERBINDUNG FEHLGESCHLAGEN');
32        echo "Verbunden als <i>$BNAME</i> mit <i>$DBANK</i>";
33 ?>
34 </center></form>
35 </body>
36 </html>
```



Das Formular verwendet die POST-*Methode* → also muss man zum Auslesen der Formulareingaben das $_POST-Array heranziehen *(Z. 20 - Z.23) Erst nach* submit *ist das $_POST-Array gefüllt.*

Als *action* ruft es sich beim Abschicken selbst (*mysqlI_1Datenbankverbindung.php*) wieder auf (es könnte auch eine andere Datei aufgerufen werden)

In einer HTML-Tabelle kann der User Angaben zur Datenbank-Verbindung vornehmen. Über das *Value*-Attribut werden die Standardwerte vorgegeben.

// leiten einen einzeiligen Kommentar im PHP-Code ein. Funktioniert nur zwischen den PHP-Tags.

Z. 20: falls das Formular erstmals vom User aufgerufen wird, enthält das $_POST-Array noch keine Inhalte. Das würde zu einem Fehler führen. Zwar können die einzelnen Fehlermeldungen durch ein vorangestelltes @-Zeichen unterdrückt werden, eleganter aber ist es, das ganze PHP-Skript für den Fall abzubrechen, dass der User noch keine Eingaben vorgenommen hat.

Z.22 - Z.25: Die verschiedenen Inhalte aus dem $_POST-Array werden ausgelesen und Variablen übergeben. Diese werden in Z.27 und Z.30 den mysql-Befehlen als Parameter übergeben.

„*or die*" sendet den dahinter stehenden Text als Fehlermeldung, wenn der vorstehende Befehl misslingen sollte. (Z.28 u. Z.31)

Alle Datenbanken des MySQLServers in einem Kombinationsfeld ausgeben:

```
…
1  <td><select name="datenbank">
2         <?php
3         $lstDBs=mysql_list_dbs($verbindung);
4         if ($lstDBs)
5                {while ($zeile=mysql_fetch_row($lstDBs))
6                       {$dbname=$zeile[0];
7                              echo "<option>$dbname</option>\n";}}
8         ?>
9  </select></td>
```

Mit *<select><option>…</option><option>…</option></select>* erstellen Sie ein Kombinationsfeld. Sie füllen es mit einer While-Schleife (Z.5f.).

Verbunden als *root* mit *testdb*