

Diese Sequenz erstellt das Texteingabefeld mit dem Namen Familienname und der Länge 45.

```
<input type=text name="Familienname" size = 45>
```

Mit der Erstellung des Submitbuttons ist das HTML-Formular fertig.

```
<input type=submit name ="submit" value =senden>
</form>
```

Nach einem Zeilenumbruch (
) beginnt das PHP-Skript. Der Client ruft das Skript erneut auf und übergibt dabei dem Skript in der Variablen \$Familienname den im Textfeld eingetragenen String. Der echo-Befehl des Skripts gibt diesen Wert aus. Der Rest des Skripts beendet den HTML-Code.

```
<br>
<? echo $Familienname;
?>
</body>
</html>
```

2.4 Zugriff auf die Datenbank

Um auf die Datenbank zugreifen zu können, sind folgende Informationen notwendig:

- **Auf welchem Server liegt die Datenbank?**
Es ist nicht notwendig, dass der Webserver und der Rechner der Datenbank identisch sein müssen. Es ist notwendig, entweder die IP-Adresse oder den Namen des Rechners zu kennen. (Im Fall des selben Rechners wird der Name localhost verwendet.)
- **Wie heißt die Benutzerkennung unter dem der Datenbankzugriff erfolgen soll?**
Diese Benutzerkennung ist nicht jene des Betriebssystems. Der Benutzer root der Datenbank ist daher nicht der gleiche, wie der Benutzer root des Betriebssystems.
- **Das Passwort der dazugehörigen Benutzerkennung**
Da das Skript am Server zuerst ausgeführt wird, gehen diese Daten (Benutzerkennung und Passwort) nicht an den Client weiter. (Es sei denn, der Programmierer sieht dies ausdrücklich vor.)
- **Den Namen der Datenbank, mit der gearbeitet werden soll**

Das folgende PHP-Skript verwendet die Funktion mysql_connect um die Verbindung zum Datenbanksystem herzustellen. \$link_id liefert einen Handle zurück, falls die Verbindung hergestellt werden konnte. Dieser Handle wird für die weiteren Datenbankaktivitäten benötigt.

```
<html><body>
<?
$MySQL Host = "localhost";
$MySQL User = "michi";
$MySQL Passw = "12345";
// mit der Datenbank verbinden
$link_id = mysql_connect("$MySQL Host",
                        "MySQL User",
                        "MySQL Passw"
                    );
if (! $link_id )
    echo "<br> ".
        "Die Verbindung zur Datenbank konnte nicht hergestellt werden".
        "<br>";
else
    echo "<br> ".
        "Die Verbindung zur Datenbank ist
hergestellt <br>";
?>
</html></body>
```

2.5 Eine erste Abfrage

Nachdem die Verbindung zur Datenbank erstellt wurde, ist der Zugriff auf die Daten möglich. Dazu sind folgende Voraussetzungen notwendig:

- Ein Verbindungshandle, wie im vorigen Kapitel beschrieben,
- den Namen der Datenbank, die die Daten liefern soll,
- eine SQL-Zeichenkette, für die Abfrage der Daten aus der Datenbank.
- Das folgende Skript stellt die Verbindung zur Datenbank VIT her, und übergibt dieser den

SQL-String. Rückgabewert ist ein Handle, mit dessen Hilfe, die aus der Datenbank gewonnenen Werte dem Skript übergeben werden können.

```
<html><body>
<?
$MySQL Host = "localhost";
$MySQL User = "michi";
$MySQL Passw = "12345";
$DataBase = "VIT";
$SQL = "Select FamilienName from Mitglieder;";
// mit der Datenbank verbinden
$link_id =mysql connect("$MySQL Host",
                        "MySQL User",
                        "MySQL Passw "
                    );
// Abfrage durchführen
$res = mysql_db_query ($DataBase,$SQL);
if (! $res )
    echo "<br> Die Abfrage konnte nicht durchgeführt werden <br>";
else
    echo "<br> Die Abfrage war erfolgreich <br>";
?>
</html></body>
```

2.6 Ausgabe der Daten in HTML

Die Ausgabe einer Tabelle beginnt mit

```
echo "<table border=";
```

Um für den Tabellenkopf die Spaltennamen ausgeben zu können, werden in einer Schleife alle Spaltennamen durchlaufen. mysql_num_fields(\$res) gibt an, wie viele Spalten die Abfrage hat; mysql_field_name(\$res,\$i) gibt den Namen an der Position \$i an.

```
echo "<tr>";
for ($i = 0; $i < mysql_num_fields($res); $i++)
{
    echo "<th>";
    echo mysql_field_name($res,$i);
    echo "</th>";
}
echo "</tr>";
```

Verwendet wird mysql_fetch_row (\$res), um einen kompletten Datensatz zu bekommen. Jeder Aufruf dieser Funktion erhöht automatisch den Datensatzzeiger. Solange es Daten gibt, wird das Array \$zeile gefüllt, gibt es keine Daten mehr, wird der Wert false zurückgegeben.

```
while ($zeile = mysql_fetch_row ($res) )
{
    echo "<tr>"; //eine Zeile beginnt
    for ($i = 0; $i < mysql_num_fields($res); $i++)
        echo "<td>". $zeile[$i]."</td>";
    echo "</tr>\n"; // eine Zeile endet
}
```

Fehlt nur mehr der Befehl zum Beenden der Tabelle:

```
echo "</table border=";
```

