



Selbst programmiert

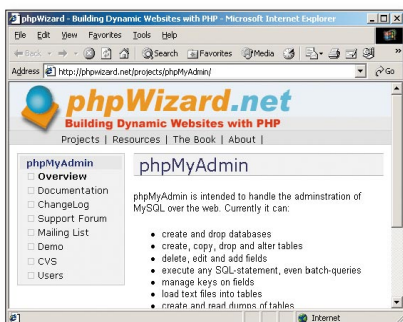
# Gästebuch mit PHP und SQL

Jede Webseite, die etwas auf sich hält, braucht ein Gästebuch, auf dem sich die Besucher der Seite eintragen und Nachrichten hinterlassen können.

Programmieren Sie doch Ihr eigenes Gästebuch mit PHP und MySQL selbst.

THOMAS WÖLFER

Sie benötigen ein installiertes PHP sowie ein lauffähiges MySQL auf Ihrem Rechner. Beide Tools finden Sie – zusammen mit einigen Hilfsprogrammen – auf der Heft-CD zu diesem Sonderheft.



**PHPMYADMIN IST DAS TOOL** der Wahl, wenn es um die Administration von MySQL-Datenbanken geht. Bei phpWizard finden Sie die Dokumentation dazu.

Wenn Sie bereits eine eigene Webseite haben, ist die Change groß, dass Ihr Hosting-Provider zumindest den PHP-Support bereits zur Verfügung stellt.

Der MySQL-Zugang hingegen ist nicht bei jedem Hosting-Angebot dabei. In diesem Fall müssen Sie unter Umständen erst nach einem neuen Provider suchen. Während dieser Zeit können Sie auch auf Ihrem eigenen Windows- oder Linux-Rechner programmieren.

## Die Grundlagen

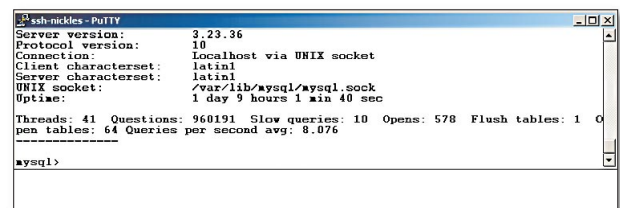
Da Sie im Folgenden ein Script erstellen werden, das auch auf eine MySQL-Datenbank zugreift, müssen Sie in der Lage sein, eine solche Datenbank zu verwalten. Das geht am einfachsten mit phpMyAdmin. Eine Kurzbeschreibung zu diesem Tool finden Sie im Kasten „MySQL administrieren mit phpMyAdmin“.

Die vorgestellte Gästebuch-Lösung soll nur als Beispiel dienen und ist dementsprechend einfach gestrickt. Eine Gästebuch-Meldung besteht deshalb gerade mal aus einem Datum, einer Überschrift und einem Fließtext. Für die interne Verwaltung wird außerdem eine eindeutige Nummer benötigt. Daraus resultieren vier Felder, die eine neue Tabelle definieren. Diese Tabelle legen Sie mit phpMyAdmin an.

In phpMyAdmin öffnen Sie dazu zunächst die Datenbank, in der die neue Tabelle definiert werden soll. Klicken Sie auf das kleine „+“-Zeichen neben dem Namen der Daten-

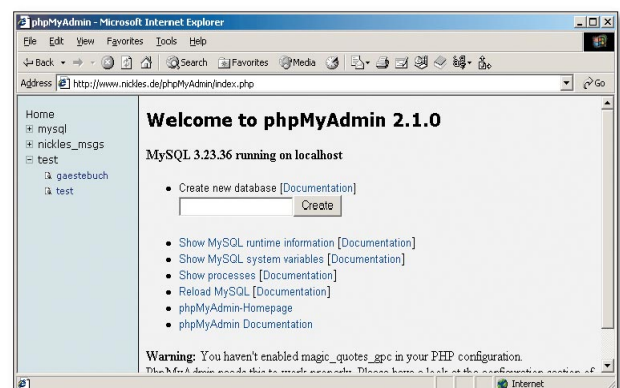
bank. phpMyAdmin zeigt nun alle bisher definierten Tabellen in dieser Datenbank an. Ganz unten auf der Seite befindet sich ein Feld mit der Überschrift *Create new table on database...*. Damit können Sie eine neue Tabelle anlegen. Allerdings müssen Sie den Namen der Tabelle sowie die Anzahl der Felder darin angeben. Als Name für die Tabelle wird in unserem Beispiel *gaestebuch* verwendet, die Anzahl der Felder ist wie bereits erläutert vier.

Danach müssen die Tabellen-Felder definiert werden. phpMyAdmin prä-

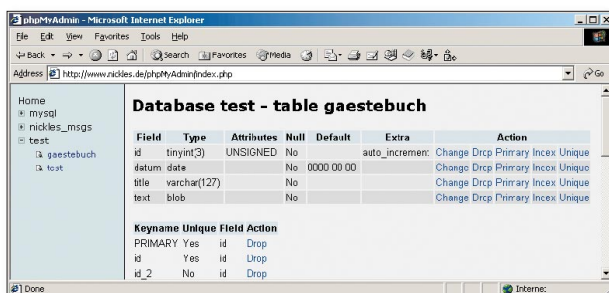


**EINE MYSQL-DATENBANK** können Sie von der Kommandozeile aus verwalten. Das ist aber reichlich umständlich.

sentiert dazu ein Formular, in dem die Namen und Datentypen der einzelnen Felder angegeben werden können. Außerdem können Sie auf diesem Formular weitere Parameter für die Felder definieren. Diese weiteren Parameter dienen zum Beispiel der Index-Bildung



**MIT PHPMYADMIN LEGEN SIE** die neuen Datenbanken und Tabellen an. Auch die Gästebuch-Tabelle macht da keine Ausnahme.



**DIE GÄSTEBUCH-TABELLE** ist relativ leicht zu überschauen und setzt sich aus gerade mal vier Feldern zusammen.

oder beschreiben die Art und Weise, in der MySQL das Feld verwalten soll.

Das Feld mit der fortlaufenden Nummer erhält den Namen „id“. Dieses Feld ist das einzige, bei dem Sie noch viele weitere Parameter angeben müssen. Um sicherzustellen, dass diese Nummer für Ihr Gästebuch wirklich eindeutig ist, müssen Sie MySQL über diesen Wunsch informieren.

Als Datentyp verwenden Sie *TINYINT*, als Attribut *UNSIGNED* – es wird also keine News mit negativen Nummern geben. Geben Sie außerdem an, dass das Feld „NOT NULL“ werden darf, und verwenden Sie als *default*-Wert die „0“.

Schließlich müssen Sie unter den *Extra*-Parametern noch *auto\_increment* auswählen. Das stellt sicher, dass sich MySQL selbst um die fortlaufende Nummerierung dieses Feldes kümmert.

## TOOLS FINDEN

Für ein Gästebuch mit PHP und MySQL benötigen Sie sowohl PHP als auch MySQL.

PHP finden Sie unter

[www.php.net](http://www.php.net)  
und MySQL unter  
[www.mysql.com](http://www.mysql.com).

Diese Tools sind auch auf der Heft-CD zu diesem Heft zu finden und zwar im „Tools“-Verzeichnis. Darunter befinden sich sowohl die Quellcodes zu PHP als auch die Binaries für Win32 – also die Dateien, die Sie benötigen, wenn Sie PHP unter Windows statt unter Linux/Unix verwenden möchten. Dazu gehört auch der PHP-Installer, der die Installation und Konfiguration unter Windows vornimmt. Ferner finden Sie in diesem Verzeichnis ein aktuelles MySQL für Linux und eines für Windows sowie das phpMyAdmin-Werkzeug, das im MySQL per auf PHP basierende Webseiten zu verwalten ist.

der Index-Bildung in der Realität bei Ihnen anbietet, müssen Sie später anhand des Ihnen konkret vorliegenden Problems selber entscheiden. Für das Gästebuch ist dies jedenfalls die richtige Auswahl.

Sie benötigen nun noch die drei Felder, die für das Gästebuch tatsächlich notwendig sind. Das sind die Felder *datum*, *titel* und *text*. *datum* soll dabei das Datum des Gästebuch-Eintrags aufnehmen. Sie verwenden also den Datentyp *date* und lassen alle anderen Parameter im phpMyAdmin leer.

*titel* wird die Überschrift des Gästebuch-Eintrags aufnehmen. Dieser wird künstlich in der Länge begrenzt.

Verwenden Sie als Datentyp daher *VARCHAR* und geben Sie als *Length* 127 an. Eine Überschrift darf also maximal 127 Zeichen lang werden.

Schließlich definieren Sie noch den Nachrichten-Text mit dem Feld *text*. Dieses wird ein BLOB-Feld ohne Längenbegrenzung.

Klicken Sie schließlich auf *Save* – das speichert die Tabellendefinition in der Datenbank. Die Definition der Tabelle ist damit abgeschlossen, und im Prinzip haben Sie damit auch das Datenbank-Back-end fertig implementiert. Wenn Sie es wünschen, können Sie mit phpMyAdmin ab sofort eigene Gästebuch-Einträge vornehmen, aber das ist vermutlich nicht der Sinn der Sache.

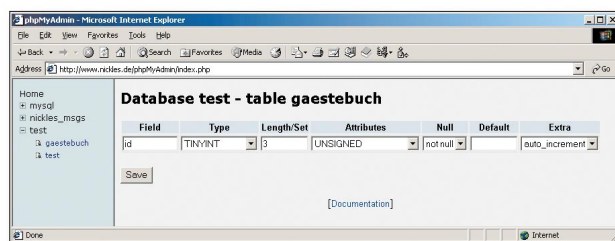
Nun müssen Sie das Gästebuch selbst

Zuletzt klicken Sie noch die drei abschließenden Options-Felder an. Diese dienen im Wesentlichen der Bildung des Index – die Nummer wird einzigartig (unique) sein und den primären Index für die Tabelle bilden. Ob das sinnvoll ist oder sich eine andere Art

programmieren. Das geschieht mit einer PHP-Datei, die verschiedene Funktionen enthält. (Sie finden den kompletten Quell-Code zu einem funktionierenden Gästebuch auf der Heft-CD.)

Der Hauptteil des Programmes erzeugt im Wesentlichen den HTML-Quell-Code für den Seitenkörper. Außerdem rufen Sie hier verschiedene Funktionen auf und reagieren auf etwaige Parameter:

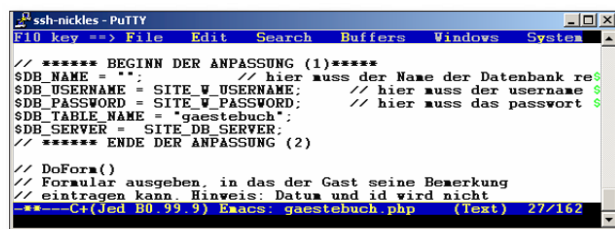
```
print '<html><head><title>Das  
eigene Gästebuch</title></head>  
<body>';  
print '<font face=Verdana>';  
print '<h2>Hallo auf dem  
Gästebuch! Trag doch auch was  
ein!</h2>';  
// formular anzeigen  
DoForm();  
// liegen daten zum eintragen  
an ?  
if( IsSet( $cmd )) InsertData(  
$titel, $text);  
// inhalt des gaestebuches  
anzeigen;  
ListContent();  
// ende der seite.  
print '</body></html>';
```



**DATENTYPEN, FELDERANZAHL UND SONSTIGE PARAMETER** können Sie auch nachträglich mit phpMyAdmin leicht ändern.

Alle aufgerufenen Funktionen werden weiter unter erläutert. Zunächst verwenden Sie den *print*-Befehl von php, um die üblichen HTML-Elemente auszugeben. Dazu gehört auch ein einleitender Text: Wie Sie dies auf Ihrer eigenen Seite gestalten, bleibt Ihnen überlassen.

Rufen Sie die Funktion *DoForm()* auf – diese wird im Wesentlichen das Formular anzeigen, das der Surfer verwenden kann, um einen neuen Gästebuch-Eintrag vorzunehmen.



**DAS GÄSTEBUCH-SCRIPT IST** von Haus aus dafür vorbereitet, von Ihnen angepasst zu werden. Dadurch können Sie das Script sehr einfach auf Ihrem Server verwenden.



```
function DoForm()
{
    print '<form action=gaestebuch.php>';
    print 'Überschrift: <input type=text name=titel width=40><br>';
    print 'Dein Kommentar:<br>';
    print '<textarea name=text cols=40 rows=5></textarea><br>';
    print '<input type=hidden name=cmd value=enter>';
    print '<input type=submit value=Eintragen>';
    print '</form>';
}

// InsertData()
// Eintragen eines neuen Datensatzes in die MySQL Datenbank.
// Die Daten dazu stammen aus dem Formular, das von DoForm()
// ausgegeben und dann auf der Webseite angezeigt wird.
// Hinweis: die Daten müsssten eigentlich noch Escaped
// werden, ansonsten führen Hochhaken etc. im Text zu
// Ärger.
function InsertData( $titel, $text)
{
    // verbindung zur datenbank
    // herstellen
    $c = mysql_connect(
        $DB_SERVER, $DB_USERNAME,
        $DB_PASSWORD);

    // sql-query zusammenbauen
    $s = "insert into $DB_
        TABLE_NAME ( datum, title,
        text) values( NOW(),
        '$titel', '$text')";

    // sql-query an datenbank
    // server schicken
    $r = mysql_db_query(
        $DB_NAME, $s, $c);

    // falls fehler aufgetreten:
    // protokollieren
    if( ! $r) print "<b>Fehler
        beim eintragen: $s </b>";
}

```

**DAS GÄSTEBUCH-SCRIPT** können Sie mit jedem Editor bearbeiten. Natürlich auch unter Windows mit VC++.

Im Folgenden prüfen Sie, ob ein bestimmter Parameter gesetzt ist. Der Parameter *cmd* wird dann gesetzt sein, wenn ein Anwender soeben das Gästebuch-Formular ausgefüllt und abgeschickt hat. Dieses Formular hat ein spezielles *hidden*-Feld mit dem Namen „cmd“ - und nachdem das Formular einfach an die Datei „gaestebuch.php“ zur Auswertung geschickt wird, ist in einem solchen Fall dieser Parameter gesetzt. Dann sind aber auch andere Parameter gesetzt und zwar für jedes im Formular ausgefüllte Feld einer. Diese heißen *titel* und *datum* und werden von Ihnen an die Funktion *InsertData()* übergeben. Diese wird sich um das Eintragen der Daten in der Datenbank kümmern.

Bevor Sie nun mit *print* das Ende der HTML-Seite ausgeben, rufen Sie noch *ListContent()* auf. Diese Funktion gibt den Inhalt des Gästebuchs aus.

Die Seite enthält immer das Formular zum Eintragen eines neuen Textes, zeigt den Inhalt des Gästebuchs an und kümmert sich außerdem um die Verarbeitung neuer Einträge.

Zunächst zur Funktion *DoForm()*:

```
print '<form action=gaeste
    buch.php>';
print 'Überschrift: <input
    type=text
    name=titel
    width=40><br>';
print 'Dein Kommentar:
    <br>';
print '<textarea name=text
    cols=40 rows=5>
    </textarea><br>';
print '<input type=hidden
    name=cmd value=enter>';
print '<input type=submit
    value=Eintragen>';
print '</form>';
```

Diese Funktion gibt nur den für ein einfaches Formular benötigten HTML-Code aus. Dabei sind ein paar Dinge zu beachten. Zum einen ist das Ziel des Formulars (*action=*) die gleiche Datei, die

das Gästebuch auch enthält. Wie gesagt, das Gästebuch-Programm kümmert sich nicht nur um die Anzeige, sondern auch um die Auswertung neuer Einträge. Das hat unter anderem den Vorteil, dass ein Besucher, der einen neuen Eintrag vornimmt, nach dem Absenden seines Eintrags sofort wieder auf der Gästebuchseite landet und darauf auch direkt seinen neuen Eintrag wiederfindet.

Zum anderen sind die Namen der einzelnen Felder wichtig. Bei PHP werden beim Auswerten der Formular-Daten automatisch Variablen mit diesen Namen angelegt und dem Ziel-Script übergebene. Darum konnten Sie im Hauptprogramm auch die Variable *text* und *titel* verwenden.

Zum dritten gibt es das „versteckte“ Feld „cmd“. Auch dieses Feld wird in eine PHP-Variable umgewandelt, sodass Sie den Inhalt dieser Variable im

Hauptprogramm abfragen können, um herauszufinden, ob neue Daten vorliegen.

## Daten einfügen

Die Funktion *InsertData* ist schon etwas komplizierter:

```
function InsertData( $titel,
    $text)
{
    // verbindung zur datenbank
    // herstellen
    $c = mysql_connect(
        $DB_SERVER, $DB_USERNAME,
        $DB_PASSWORD);

    // sql-query zusammenbauen
    $s = "insert into $DB_
        TABLE_NAME ( datum, title,
        text) values( NOW(),
        '$titel', '$text')";

    // sql-query an datenbank
    // server schicken
    $r = mysql_db_query(
        $DB_NAME, $s, $c);

    // falls fehler aufgetreten:
    // protokollieren
    if( ! $r) print "<b>Fehler
        beim eintragen: $s </b>";
}

```

Die Funktion stellt zunächst eine Verbindung zum MySQL-Server her. Dazu verwenden Sie die Funktion *mysql\_connect()*, die mit drei Parametern aufgerufen werden muss. (Im Beispiel

## GÄSTEBUCH ANPASSEN

Das Gästebuch-Script, das Sie auf der CD finden, wurde von vornherein so geschrieben, dass Sie es leicht auf Ihrem eigenen Server einsetzen können. Später werden Sie sicherlich Veränderungen vornehmen wollen, doch zu Anfang ist es hilfreich, wenn Sie sofort ein laufendes Script haben.

Dazu sind aber verschiedenen Dinge anzupassen – diese finden Sie alle oben im Script unter der Markierung: \*\*\*\*\* BEGIN DER ANPASSUNG \*\*\*\*\*

Hier sind fünf Variable zu setzen, die Ihre Umgebung definieren. Dies sind die Variable *DB\_NAME*, *DB\_USERNAME*, *DB\_PASSWORD*, *DB\_TABLE\_NAME* und *DB\_SERVER*.

Hier die Bedeutung im Einzelnen:

### DB\_NAME:

Diese Variable legt den Namen Ihrer Datenbank fest. Das ist der Name, den Sie beim Anlegen der Datenbank in phpMyAdmin vergeben haben.

### DB\_USERNAME:

Das ist der Name eines Users mit Schreibrechten in der Datenbank. Ein einfacher User reicht nicht aus, da neue Daten in die Datenbank geschrieben werden sollen.

### DB\_PASSWORD:

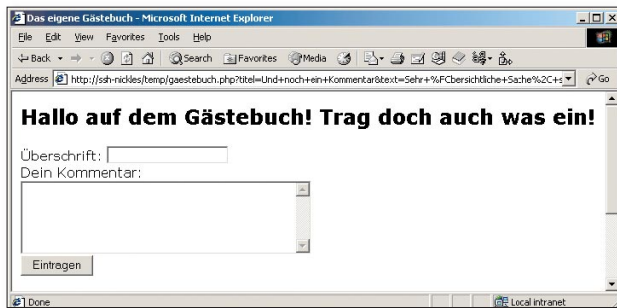
Dabei handelt es sich um das Passwort, das zu *DB\_USERNAME* gehört.

### DB\_TABLE\_NAME:

Das ist der Name der Tabelle, die das Gästebuch enthält. Wenn Sie Ihr Gästebuch anhand des Beispiels im Artikel erzeugt haben, ist das der Text „gaestebuch“.

### DB\_SERVER:

Dabei handelt es sich um den Namen des MySQL-Servers. Das ist normalerweise nur „localhost“. Ihr Hosting-Provider sollte Ihnen diese Information aber geben können.



**DAS GÄSTEBUCH-SCRIPT IN ACTION:** Noch ist kein Eintrag von Gästen da.

sind das globale Variable. Wozu diese Variablen gut sind, finden Sie im Kasten „Gästebuch anpassen“.

Der erste Parameter gibt den Namen Ihres MySQL-Servers an. Der zweite Parameter ist der Name eines Accounts mit Schreibrechten auf diesen Server und der dritte Parameter ist das zu diesem Account gehörende Passwort.

Bauen Sie nun den Query-String zusammen. Obwohl Sie nur zwei Daten – nämlich *text* und *titel* zum Eintragen haben, tragen Sie aber drei Daten ein. Den Wert für „datum“ lassen Sie von MySQL mit der Funktion *NOW()* bestimmen. (Diese Funktion wird nicht von PHP ausgeführt, sondern erst vom Datenbank-Server, also erst nachdem Sie den Query-String an den Server übertragen haben. Nachdem das „id“-Feld in der *gaestebuch*-Tabelle auf *auto-increment* gestellt wurde, braucht es hier nicht mit übergeben zu werden. MySQL wird den Wert für dieses Feld automatisch setzen und verwalten.

Den fertigen SQL-Query String senden Sie mit der Funktion *mysql\_db\_query()* an die Datenbank.

völlig anderes Query entsteht, als das, was Sie eigentlich absetzen wollten. Besonders schlecht sind dabei Hochhaken jeder Art im Query-String. Nachdem sowohl *text* als auch *titel* Werte sind, die von Ihren Besuchern eingegeben werden können, ist es durchaus möglich, dass diese Zeichen in den Strings enthalten sind. Aus diesem Grund müssten diese beiden Variablen bzw. deren Inhalte eigentlich vor dem Zusammenbauen des Query-Strings escaped werden.

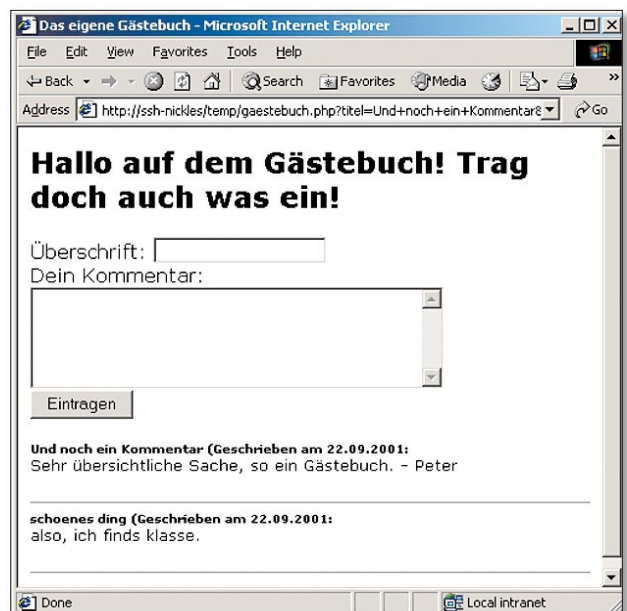
*mysql\_db\_query()* erhält drei Parameter. Der erste gibt den Namen der Datenbank an, mit der Sie

Wenn alles gut ging, sind die Daten danach eingetragen. An dieser Stelle bedarf das Programm aber eigentlich noch einer Erweiterung. Bei SQL ist es einfach, den Query-String zu zerstören, wenn er ungünstige Zeichen enthält. Dadurch kann es sogar dazu kommen, dass ein

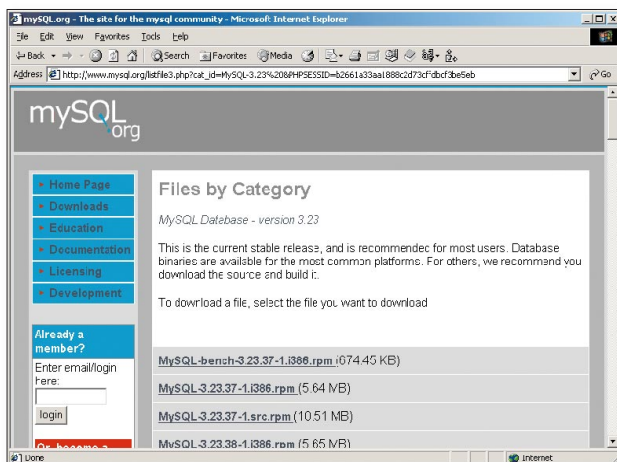
Schließlich überprüfen Sie noch den Rückgabe-Wert von *mysql\_db\_query()*. Liefert dessen Auswertung *False*, ist ein Fehler aufgetreten, ansonsten sind die Daten in der Datenbank eingetragen worden.

Jetzt bleibt nur noch die Funktion *ListContent()*. Diese ist für die Anzeige des Gästebuch-Inhalts zuständig.

In den Grundzügen funktioniert diese Funktion so ähnlich wie *InsertData()*. Auch hier stellen Sie zunächst eine Verbindung zum Datenbank-Server her, bauen dann den Query-String zusammen



**DAS GÄSTEBUCH MIT DEN ERSTEN EINTRÄGEN.** Die optische Gestaltung des Gästebuches ist Ihnen frei überlassen. Das Original-Script verwendet fast keinerlei Formatierungen.



**MYSQL GIBT ES IN DUTZENDEN VON DISTRIBUTIONEN:** Wenn Ihnen die Distribution der Heft-CD nicht zusagt, können Sie noch eine der anderen ausprobieren.

kommunizieren wollen. Der zweite Parameter ist Ihr Query-String und der dritte Parameter definiert die SQL-Verbindung, die verwendet werden soll. Darum müssen Sie den Namen der Datenbank und den der Verbindung angeben, weil Sie ja auch mit mehreren unterschiedlichen Datenbank-Servern oder mehreren Datenbanken auf dem gleichen Server arbeiten können.

men und übergeben ihn per *mysql\_db\_query()*.

Der Query-String ist ein wenig komplizierter. Nachdem Sie das Datum in „deutscher“ Form anzeigen möchten, müssen Sie MySQL dazu veranlassen, den Inhalt des *datum*-Feldes entsprechend zu formatieren. Das geht mit dem „DATE\_FORMAT“-Teil des Query-Strings:

```
$s = "select DATE_FORMAT  
      (datum, '%d.%m.%Y') as date,  
      title, text from "  
      $DB_TABLE_NAME . " order by id  
      desc";
```

Ferner sollen nicht alle Daten aus der Tabelle geliefert werden. Welche Felder Sie wünschen, müssen Sie nach dem SELECT-Statement angeben: In diesem Fall sind das alle Felder außer dem ID-Feld. Dafür sollen die Datensätze aber

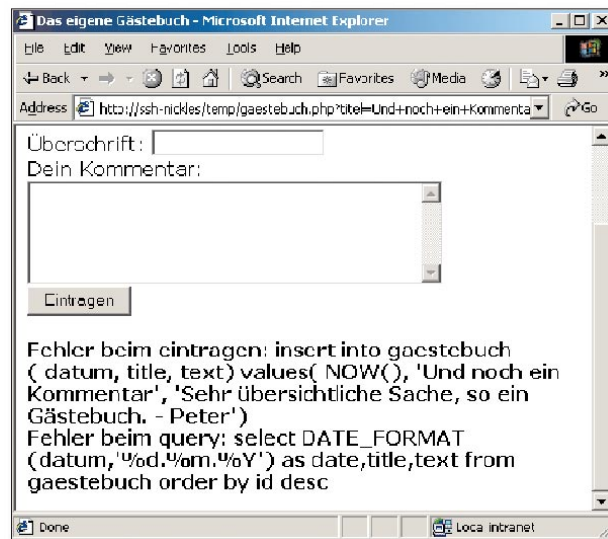


in absteigender Reihenfolge sortiert zu diesem Feld (order by id desc) geliefert werden. Das führt dazu, dass der Inhalt des Gästebuches so angezeigt wird, dass der neueste Eintrag an oberster Stelle steht.

Nach dem `mysql_db_query()` können Sie nun mit `mysql_num_rows()` die Anzahl der gelieferten Zeilen ermitteln: Für jeden Gästebuch-Eintrag gibt es eine Zeile.

```
$n = mysql_num_rows( $r );
```

Programmieren Sie eine Schleife über die Anzahl der Zeilen und erfragen bei jedem Iterations-Schritt die nächste Zeile von MySQL. Dazu gibt es eine ganze Reihe von Funktionen. Am be-



**SO SIEHT ES AUS, WENN ETWAS SCHIEFGEHT**, beim Eintragen. Meist hat es damit zu tun, dass Sie einen falschen Server- oder Usernamen angegeben haben. Alternativ stimmt das Passwort nicht.

sten verwenden Sie hier `mysql_fetch_object()`, denn das liefert Ihnen pro Zeile ein PHP-Objekt zurück. Dieses Objekt hat Member-Variable, die den

Namen der ursprünglich erfragten Felder entsprechen. So können Sie einfach auf diese Felder unter Ihrem richtigen Namen zugreifen:

```
for( $i=0; $i<$n;$i++)
{
    $o = mysql_fetch_object( $r );
    print "$o->title (Geschrieben am  
    $o->date: <br>";
    print $o->text;
    print '<hr size=1>';
}
```

In diesem Beitrag haben Sie erfahren, wie Sie schnell ein eigenes kleines Gästebuch-System programmieren können. Einige wichtige Aspekte bleiben noch zu implementieren.

Sie werden im Laufe der Zeit ganz sicher ein Interface benötigen, mit dem Sie einen einmal eingegebenen Gästebucheintrag editieren oder löschen können. Für den Anfang können Sie dazu natürlich einfach phpMyAdmin verwenden.

Viel Spaß beim Lesen der Nachrichten Ihrer Gäste. UR

## MYSQL ADMINISTRIEREN MIT PHPMYADMIN

Die verschiedenen Kommandozeilen-Werkzeuge für MySQL sind für die Verwaltung des Datenbanksystems sicherlich praktisch. Benutzerfreundlich sind sie nicht.

Besonders dann, wenn Ihr MySQL auf einem entfernten Rechner läuft, zu dem Sie unter Umständen noch nicht einmal eine Telnet-Verbindung aufbauen können (oder dürfen) sind diese Werkzeuge nicht sehr hilfreich. Anders ist das mit phpMyAdmin: Dieses Werkzeug hat nicht nur eine einfach zu bedienende graphische Oberfläche, sondern kann auch einfach über den Browser bedient werden. Das macht phpMyAdmin zum perfekten Wartungs-Werkzeug für MySQL.

phpMyAdmin basiert vollständig auf php. Das bedeutet, dass Sie auf Ihrem Server eine aktuelle Version dieser Sprache benötigen.

Das Werkzeug "selbst" erhalten Sie unter der URL

<http://www.phpwizard.net/projects/phpMyAdmin/>

– oder Sie verwenden einfach die Version von dieser Heft-CD, die Sie im *tools*-Verzeichnis finden.

Sie können phpMyAdmin starten, indem Sie mit dem Browser in das Installations-Verzeichnis von phpMyAdmin navigieren. Haben Sie noch keine Datenbank angelegt, zeigt phpMyAdmin eine zweigeteilte Seite an. Im linken Bereich befindet sich nur ein Link mit der Bezeichnung "Home" – im rechten Bereich stehen die initial vorhandenen Befehle.

Wenn Sie mit phpMyAdmin neue Datenbanken anlegen, werden diese alle unterhalb von "Home" angezeigt; ist eine Datenbank definiert, kann diese auf der phpMyAdmin-Startseite ausgewählt werden. Daraufhin verändern sich die zur Verfügung stehenden Befehle auf dem rechten Teil dieser Seite.

Der zunächst wichtigste Befehl ist der Befehl *Create new database*. Unter diesem Befehl finden Sie ein Textfeld. Geben Sie dort den gewünschten Namen für die neue Datenbank an, und drücken Sie auf den Knopf *Create*. phpMyAdmin legt nun eine neue Datenbank an.

Ist die Datenbank angelegt, befinden sich gleich zwei neue Einträge unter "Home". Der eine trägt den Namen "mysql", der andere den Namen Ihrer Datenbank.

In der Abteilung „mysql“ können Sie die von mysql benötigten Tabellen verändern. Hier finden Sie die einfachste Möglichkeit zum Anlegen eines neuen Datenbank-Users.

Die zuvor angelegte Datenbank muss noch mit mindestens einer Tabelle bestückt werden. Klicken Sie dazu auf den Namen Ihrer Datenbank unter „Home“ – im unteren Bereich des daraufhin angezeigten Formulars finden Sie den Befehl *Create new table on database NameIhrerDatenbank*. Geben Sie im passenden Textfeld den Namen der Tabellen ein.

Außerdem müssen Sie festlegen, wie viele Spalten (Fields) die Tabelle haben soll. Dabei nur keine Angst, Sie können mit phpMyAdmin die Anzahl der Spalten und deren Datentypen nachträglich leicht verändern. Testweise tut es also auch eine Tabelle mit zunächst nur einer Spalte.

Drücken Sie dann auf *go* – schon ist die Tabelle angelegt und erscheint unterhalb des Namens Ihrer Datenbank im rechten Bereich von phpMyAdmin: Nun müssen Sie noch festlegen, welche Datentypen Ihre Spalte(n) haben sollen – phpMyAdmin unterstützt alle Datentypen von MySQL.