

## **Datenbanksysteme II**

### **SS 2008 – 2. Praktische Übung – Datenbankgestützte Webanwendung mit PHP**

In dieser praktischen Übung erstellen Sie eine einfache Webanwendung mittels PHP, die auf eine existierende Datenbank zugreift. Das grundlegende Vorgehen ist u.a. im Skript zur Vorlesung (Kapitel 2) illustriert.

Nach dem Durcharbeiten dieser praktischen Übungen sollten Sie in der Lage sein, einfache Webanwendungen zu schreiben, die

- eine Datenbankverbindung auf- und wieder abbauen,
- SQL-Anweisungen auf Basis von Nutzereingaben an die Datenbank senden und
- die Ergebnisse von SQL-SELECT-Anweisungen als Webseite darstellen

können. Sofern Sie keine Zwischenklausur schreiben, werden diese Fähigkeiten zusammen mit dem Inhalt der ersten praktischen Übung im Rahmen einer praktischen Klausur getestet, deren erfolgreiches Bestehen (neben der erfolgreichen Bearbeitung der LOTS-Übungsblätter) Voraussetzung für die Teilnahme an der Abschlussklausur DBS2 ist.

*Allgemeine Hinweise zu PHP:* Falls Sie noch keine Erfahrung mit PHP besitzen, bietet das „PHP Praxisbuch“ (<http://www.selfphp.de/praxisbuch/index.php>) einen guten Einstieg. Es beinhaltet sowohl eine Beschreibung der benötigten Software als auch eine komplette Übersicht der Sprachelemente und Syntax. In den Rechner-Pools der Informatik (Johannisgasse 26, 3. Etage) ist bereits sämtliche benötigte Software zur PHP-Programmierung (inklusive mächtiger Entwicklungsumgebungen wie Eclipse) installiert.

*Allgemeine Hinweise zur Entwicklung:* Sie können das praktische Übungsblatt auf einem Rechner Ihrer Wahl bearbeiten, wobei Sie gegebenenfalls Software (z.B. einen PHP-fähigen Webserver) installieren müssen. Auf den Solaris-Rechnern im CIP-Pool des Instituts ist bereits alles installiert. Beachten Sie dazu die folgenden Hinweise:

- Sofern Sie noch keinen Nutzernamen/Passwort für die erste praktische Übung angefordert haben, erhalten Sie diese Daten, indem Sie eine E-Mail an [juseks@informatik.uni-leipzig.de](mailto:juseks@informatik.uni-leipzig.de) mit Betreff „DBS2 Praktische Übung <Matrikel>“ schreiben, wobei Sie <Matrikel> durch Ihre Matrikelnummer ersetzen.
- Mit dem Ihnen zugewiesenen Account können Sie sich auf einem der CIP-Pool-Solaris-Rechner (Johannisgasse 26, 3. Etage) einloggen (Remote-Login auf [uservX.informatik.uni-leipzig.de](http://uservX.informatik.uni-leipzig.de) mit X=1..2).
- Im Home-Verzeichnis Ihres Accounts finden Sie ein Verzeichnis `public_html`, in das Sie Ihre erstellten Programme (d.h. Ihre .php-Files) legen müssen. Bitte ändern Sie

nichts an der Konfiguration dieses Verzeichnisses, also insbesondere nicht den Namen und die Rechte), da dieses Verzeichnis für den Webserver zugreifbar sein muss.

- Ihre beiden Dateien `authors.php` und `books.php` müssen für den Server lesbar sein, was sie mit dem Unix-Befehl `chmod go+r authors.php books.php` erreichen.
- Um Ihr Programm zu testen, geben Sie in Ihrem Webbrowser die Adresse `http://leutsch.informatik.uni-leipzig.de:8080/~db2uebXXX/authors.php` ein (XXX mit Ihren Nutzernamen abgleichen). Der Webserver geht dann automatisch in Ihr `public_html`-Verzeichnis und liest die `authors.php`-Datei. Beachten Sie, dass die Ladezeit einige Sekunden betragen kann. Wählen Sie zum Testen den Autor „Rahm, Erhard“ aus. In der anschließenden Buchausgabe müssen sieben Bücher erscheinen (Web & Datenbanken; Datenbanksysteme - Konzepte und Techniken der Implementierung [2 Auflagen]; Mehrrechner-Datenbanksysteme [2 Auflagen]; Hochleistungs-Transaktionssysteme; Synchronisation in Mehrrechner-Datenbank-Systemen).

### *Allgemeine Hinweise zu Datenbankverbindungen mit PHP*

Zur Verbindung mit der Datenbank verwenden Sie folgende Verbindungsparameter:

- Hostname des Datenbankservers: `lips.informatik.uni-leipzig.de`
- Portnummer: `5432`
- Name der Datenbank: `Bibliothek`
- Nutzername: `visitor01`
- Passwort: `qitulats`

Der Verbindungsaufbau erfolgt mittels PDO (PHP Data Objects, siehe [www.php.net/pdo](http://www.php.net/pdo)):

```
$dbh = new PDO("pgsql:host=lips port=5432 dbname=Bibliothek
               user= visitor01 password=qitulats ");
```

Mit Hilfe des resultierenden Verbindungsobjekts `$dbh` können anschließend Anfragen an die Datenbank gestellt werden:

```
$result = dbh->query ("SELECT-ANFRAGE", PDO::FETCH_ASSOC);
```

## 1. Aufgabe (authors.php: Auslesen aller Autoren)

a) Erstellen Sie eine PHP-Datei "authors.php", die in einer Tabelle alle Autoren mit Vor- und Nachnamen auflistet.

Nachdem eine Anfrage mittels `$result = dbh->query ("SELECT-ANFRAGE", PDO::FETCH_ASSOC)` abgeschickt worden ist, liegt das Ergebnis in dem 2-dimensionalen Array `$result->data` vor. Zum Auslesen kann eine `for`-Schleife, die über die Zeilen iteriert (beginnend bei 0 bis `$result->rows-1`) verwendet werden (eine weitere Möglichkeit zur Iteration bietet `foreach`). Die Spalten können über die Attributnamen angesprochen werden, z.B. `$result->data[$i]["isbn"]` (mit `$i` als Schleifenvariable).

Zur Auflistung aller Autoren in einer Tabelle muss Ihr PHP-Programm HTML-Code ausgeben, der folgender Struktur entspricht.

```
<table>
  <tr><td>Heinz Mustermann</td></tr>
  <tr><td>Max Schuster</td></tr>
  ...
</table>
```

b) Erweitern Sie Ihr Programm aus Aufgabe a) derart, dass jeder Autorname mit einem Hyperlink hinterlegt ist, der auf eine PHP-Datei "books.php" verweist und als Parameter die AutorId enthält.

Für jeden Autornamen wird ein Hyperlink mit folgender Struktur gebildet:

```
<a href="books.php?author=0815">Heinz Mustermann</a>
```

wobei "0815" für die jeweilige AutorId steht.

## 2. Aufgabe (books.php: Auslesen aller Bücher eines Autors)

Erstellen Sie eine PHP-Datei "books.php", die alle Bücher eines (als Parameter) übergebenen Autors tabellarisch darstellt. Dabei sollen alle Bücher mit Titel, ISBN, Jahr, Verfasser, Preis und Währung ausgegeben werden, bei denen der ausgewählte Autor als Verfasser beteiligt ist (`rolle='v'`).

Das Auslesen des "author"-Parameters wird in PHP mit `$_GET['author']` realisiert.

## Anlage: Datenbankschema

Ihre PHP-Programme sollen auf eine bestehende SQL-Bibliotheksdatenbank zugreifen. Der für Sie relevante Teil des Bibliotheksschemas lautet wie folgt:

```
autor(  
    autorid          INT NOT NULL PRIMARY KEY,  
    nachname        VARCHAR(30) NOT NULL,  
    zusatz          VARCHAR(10),  
    vornamen        VARCHAR(30)  
);  
  
buch(  
    buchid          INT NOT NULL PRIMARY KEY,  
    titel           VARCHAR(255),  
    isbn            VARCHAR(15),  
    auflage         VARCHAR(70),  
    jahr            SMALLINT,  
    preis           NUMERIC(10,2),  
    waehrung        VARCHAR(15),  
    signatur        VARCHAR(30),  
    verlagsid       INT,  
    FOREIGN KEY(verlagsid) REFERENCES Verlag (verlagsid)  
);  
  
buch_aut(  
    buchid          INT NOT NULL,  
    autorid         INT NOT NULL,  
    rolle           CHAR(1),  
    rang            SMALLINT,  
    PRIMARY KEY(buchid, autorid),  
    FOREIGN KEY (buchid) REFERENCES buch (buchid),  
    FOREIGN KEY (autorid) REFERENCES autor (autorid)  
);
```