

Ausgabe von Daten als Tabelle

Bisher haben wir die Ergebnisse der Datenbankabfrage immer mit `<p>` oder `
` untereinander geschrieben. Jetzt wollen wir aus den Ergebnissen eine richtige Tabelle bauen, die etwa so aussieht:

```
SELECT * FROM artikel a WHERE a.bezeichnung = 'Sonnenhut'
```

Artikelname	Preis	Herstellungsland
Sonnenhut	8.9	CN
Sonnenhut	7.95	CN
Sonnenhut	4.95	ES

Und das geht so:

Nachdem wir wie gewohnt alle Vorbereitungen getroffen haben (Werte aus Formular holen, DB-Verbindung öffnen, Abfrage definieren und Abfrage ausführen) müssen wir die Ergebnisse zu einer schönen HTML-Tabelle zusammenbauen.

1. Tabelle starten und Kopfzeile ausgeben

PHP-Code	erzeugt	HTML-Code im Browser
<pre>// HTML-Tabelle bauen // Erst mal HTML-Tabelle starten und Kopfzeile ausgeben echo "<table>\n"; echo "<tr>\n"; echo "<th>Artikelname</td>\n"; echo "<th>Preis</td>\n"; echo "<th>Herstellungsland</td>\n"; echo "</tr>\n";</pre>	<pre>// erste Zeile Start // Spaltenüberschrift 1 // Spaltenüberschrift 2 // Spaltenüberschrift 3 // erste Zeile Ende</pre>	<pre><table> <tr> <th>Artikelname</td> <th>Preis</td> <th>Herstellungsland</td> </tr></pre>

PRO-Wissen: Wenn wir ein `\n` in PHP echoen, dann führt das zu einem Zeilenumbruch im Quelltext. Für die Anzeige im Browser ändert sich gar nichts, aber wenn man in den Quelltext schaut, wird's einfach übersichtlicher. `\t` fügt einen Tabstopp ein.

PHP-Code erzeugt HTML-Quelltext



```
echo "<h2>Liste</h2>";
echo "<ul>";
echo "<li>Listenpunkt</li>";
echo "</ul>";

echo "\n\n";
echo "<h2>Liste</h2>\n";
echo "<ul>\n";
echo "\t<li>Listenpunkt</li>\n";
echo "</ul>\n";
```

```
<h2>Liste</h2><ul><li>Listenpunkt</li></ul>

<h2>Liste</h2>
<ul>
\t<li>Listenpunkt</li>
</ul>
```

2. Daten in HTML-Tabellenzeilen setzen

Anfang und Ende jeder Tabellenzeile wird mit `<tr>` bzw. `</tr>` markiert. Dazwischen kommen die einzelnen Zellen mit `<td>` bzw. `</td>`. Also los!

```
// Jetzt die Daten zeilenweise ausgeben
foreach ($ergebnis as $zeile) {
    $artikelname      = $zeile['bezeichnung'];
    1 $preis           = $zeile['preis'];
    $land             = $zeile['herstellungsland'];
    2 echo "<tr>\n"; // neue Tabellenzeile
    echo "<td>$artikelname</td>\n"; // erste Spalte
    3 echo "<td>$preis</td>\n"; // zweite Spalte
    echo "<td>$land</td>\n"; // dritte Spalte
    4 echo "</tr>\n"; // Zeile ist zuende
}
```

Auch wenn es kompliziert aussieht, ist es doch recht durchschaubar:

1. Wir speichern die einzelnen Werte in Variablen.
2. Wir echoen den Zeilenanfang.
3. Wir echoen die einzelnen Zellen/Spalten. Die bestehen aus
 - a. einem `<td>`,
 - b. aus dem Wert, den wir aus der DB geholt haben, und
 - c. aus einem schließenden `</td>`
4. Wir echoen das Zeilenende.

3. Tabelle im HTML beenden

Wir echoen zum Schluss einfach ein schließendes `</table>`:

```
// Jetzt HTML-Tabelle schließen
echo "</table>\n";
```

PRO-Wissen: Damit die Tabelle schöner aussieht, können Sie sie mit CSS stylen. In unserem Fall ist das schon gemacht, der CSS-Code sieht so aus:

```
/* Tabelle mit CSS etwas stylen */
table {
    border-collapse: collapse; /* Entfernt den Abstand zwischen den Rahmen */
    width: auto; /* Passt die Breite der Tabelle an den Inhalt an */
}
td, th {
    border: 1px solid black; /* Setzt einen schwarzen Rahmen um jede Zelle */
    text-align: left; /* Textausrichtung in den Zellen */
    padding: 8px; /* Abstand zwischen Text und Rahmen */
}
th {
    background-color: #ffa1b1; /* Hintergrundfarbe für Kopfzeilen */
}
```

Wenn Sie nicht wollen, dass die Tabelle über die ganze Breite geht, können Sie dem `.container` noch ein `margin: 0px 22%`; geben (Außenabstand oben/unten: `0px`; links/rechts jeweils 22% der Container-Breite).



Aufgabe 1: Buttkes Shop Codeergänzung

Öffnen Sie das Projekt *db3-uebung1-buttke*. Sie finden dort ein Formular (*formular.php*) und eine Auswertungsdatei (*auswertung.php*). Der PHP-Teil in *auswertung.php* ist unvollständig – reparieren Sie das!

TIPP: Eigentlich ist das genau das gleiche, wie oben erklärt. Versuchen Sie es mithilfe der Kommentare im Code von *auswertung.php* selbstständig zu machen (ohne zu spicken)!



Aufgabe 2: Barbie-Lexikon in Tabelle

Öffnen Sie das Projekt *db3-uebung2-barbiepersonen*. Sie finden dort ein Formular (*formular.php*) und eine Auswertungsdatei (*barbie_info.php*).

Wieder soll der Name der Person und die Beschreibung ausgegeben werden, allerdings diesmal in Tabellenform:

Suchergebnis

Name	Beschreibung
Glitter Gary	Gary ist der exzentrische Nachbar und geheimer Superheld, dessen Kostüm vollständig mit funkelnden Pailletten bedeckt ist. Er ist immer zur Stelle, um den Tag mit einem Hauch von Glanz zu retten, und seine Spezialfähigkeit ist es, Konfetti-Regen zu erzeugen, wann immer er möchte.

TIPP:

Falls Sie es nicht sowieso schon tun – kopieren Sie sich die Grundbausteine der Datenbankverbindung aus einem alten Projekt! 😊



Aufgabe 3: Barbie-Shop

Öffnen Sie das Projekt *db3-uebung3-barbieshop*. Sie finden auf der Startseite (*start.php*) einige Formulare bzw. Links, die auf unterschiedliche Auswertungsdateien führen:

»Barbie Bewitched« Shop

Start

Zusatzaufgabe 3d

Showcase

Alle Artikel schön präsentiert!

Zeig her!

Führt zu showcase.php

Aufgabe 3a

Übersicht

Alle Artikel in Tabellenform

Tabelle zeigen

Führt zu uebersicht.php

Aufgabe 3b

Sortierte Übersicht

Alle Artikel sortiert nach

Anzeigen

Führt zu uebersichtSortiert.php

Aufgabe 3c

Suche

Suchen nach

Suchen

Führt zu suche.php

Die Tabelle *artikel* in der Datenbank *barbieshop.sqlite* hat diese Struktur:

	id	name	farbe	preis
1	1	Großer Kuschelbär	beige	49.99
2	2	Riesiger Teddybär-Sitzsack	beige	119.99
3	3	Schergewichtige Plüsch-Löwe	beige	79.99
4	4	Schergewichtige Plüsch-Känguru	beige	79.99
5	5	Seifenblasenmaschine	blau	11.49
6	6	Riesiger Schaumstoff-Hammer	blau	8.99

Aufgabe 3a) Übersicht

Die arbeiten mit der Seite *uebersicht.php*. Dort sollen alle Artikel in Tabellenform ausgegeben werden:

Artikelübersicht
Alle Artikel in übersichtlicher Tabellenform

id	name	farbe	preis
50	5 Kilo Knete	bunt	6.49
120	Blinkende Einhorn-Schuhe	lila	14.99
148	Blinkender Nasen-Karaoke-Mikrofon	pink	6.99
113	Blumenkette mit LED-Leuchten	lila	4.99
42	Cartoon-Kuschelkissen	bunt	9.99
179	Clownsfahrradhupe	rot	3.49
184	Clownsnase mit Klappaugen und Licht	rot	4.99

Sie brauchen KEINE Formulardaten, da die Seite immer das gleiche macht:

- DB-Verbindung herstellen
- SQL-Abfrage definieren, die alle Artikel holt, evtl. nach Name sortiert
- Tabelle mit allen Artikeln ausgeben
- DB-Verbindung beenden.

Aufgabe 3b) Sortierte Übersicht

Das gleiche wie 3a), aber diesmal sind die Artikel so sortiert, wie im Formular angegeben. Sie arbeiten in der Datei *uebersichtSortiert.php*.

- Holen Sie aus dem Formular in *start.php* den Wert aus dem select-Dropdown. Sie erhalten so den Feldnamen, nach dem sortiert wird.
- Geben Sie alle Artikel in Tabellenform aus, und zwar sortiert nach dem ausgewählten Kriterium (z.B. *farbe*, siehe Formular in *start.php*).

Im folgenden Beispielscreenshot wurde nach *farbe* sortiert.

Artikelübersicht (SORTIERT)
Alle Artikel SORTIERT in Tabellenform

id	name	farbe	preis
1	Großer Kuschelbär	beige	49.99
2	Riesiger Teddybär-Sitzsack	beige	119.99
3	Schwergewichtige Plüsch-Löwe	beige	79.99
4	Schwergewichtige Plüsch-Känguru	beige	79.99
5	Seifenblasenmaschine	blau	11.49
6	Riesiger Schaumstoff-Hammer	blau	8.99
7	Scherzartikel-Fliede mit LED-Licht	blau	2.49

3b ZUSATZAUFGABE (schwierig!):

Bauen Sie ins Formular in *start.php* zwei Radiobuttons ein, wo man auswählen kann, ob aufsteigend oder absteigend sortiert wird.

Aufgabe 3c) Nach Begriff suchen

Ins Formular gibt man eine Zeichenkette ein, als Ergebnis erhält man eine Tabelle, in der alle Artikel zu sehen sind, deren Namen diese Zeichenkette enthält. Hier wurde *winz* ins Formular eingegeben:

Suchergebnis			
id	name	farbe	preis
13	Winziger Schaumstoff-Hammer	blau	0.19
61	Winzige Gummienten-Familie	bunt	2.99
62	Winzige Zirkuszelt-Miniatur	bunt	2.49
63	Winziger Wasserspritzpistolen-Set	bunt	1.99
64	Winzige Domino-Steine	bunt	0.99
107	Winzige Spielzeug-Aliens	grün	1.99

Sie arbeiten auf der Seite *suche.php*:

- Holen Sie aus dem Formular von *start.php* den gewünschten Suchbegriff.
- Suchen Sie mit einer SQL-Abfrage alle Artikelnamen, die den Suchbegriff beinhalten (**LIKE!**) und geben Sie eine Tabelle mit diesen Artikeln aus.

3d) Alle Artikel schön ausgeben (Zusatzaufgabe)

Wie Aufgabe 3a), aber jetzt wollen wir es HTML-mäßig mal richtig krachen lassen:

Sie arbeiten mit der Seite *showcase.php*.

Definieren Sie (wie in 3a) eine Abfrage, die alle Artikel holt. Geben Sie alle Artikel in diesem Format aus:

```
<section class='produkt'>
  <h2>5 Kilo Knete</h2>
  <p>Farbe: bunt</p>
  <p class='preis'>6.49 €</p>
  <button>In den Warenkorb</button>
</section>

<section class='produkt'>
  <h2>Blinkende Einhorn-Schuhe</h2>
  <p>Farbe: lila</p>
  <p class='preis'>14.99 €</p>
  <button>In den Warenkorb</button>
</section>
```

etc.