

## Aufgabe 1

### (Voraussetzungen / Anmeldung)



Es soll ein Webshop erstellt werden. Zunächst soll sich der Kunde vor dem Besuch des Webshops anmelden. Unter seiner Kennung kann er dann unterschiedliche Produkte auswählen und in seinen virtuellen Warenkorb legen. Den Warenkorb kann er sich jederzeit anzeigen lassen, Artikel daraus entfernen oder die Anzahl der Produkte darin erhöhen. Nach Abschluss des Einkaufs kann der Kunde „zur Kasse“ gehen und erhält eine Rechnung über die Produkte seines Einkaufs. Rechnungen werden im System gespeichert und können jederzeit vom Kunden eingesehen, aber nicht nachträglich verändert werden. Der Warenkorb des Kunden soll nach abgeschlossenem Einkauf automatisch gelöscht werden.

Für die Realisierung dieses Webshops muss eine Datenbank gemäß folgender Anforderungsliste umgesetzt werden.

#### **Anforderungsliste:**

1. Welche Produkte werden in dem Laden angeboten?
2. Zu einem Produkt muss der Name, die Beschreibung und der Preis gespeichert werden.
3. Welche Produktkategorien werden unterschieden?
4. Lediglich eine Beschreibung zur Kategorie ist notwendig.
5. Ein Produkt wird immer genau einer Kategorie zugeordnet.
6. In einer Kategorie werden mehrere Produkte zusammengefasst.
7. Welche Hersteller gibt es?
8. Zu einem Hersteller interessieren der Name, die Webadresse und die E-Mail-Angabe.
9. Ein Hersteller produziert unterschiedliche Produkte.
10. Ein Produkt wird immer nur von einem Hersteller produziert.
11. Welche Kunden kaufen in dem Webshop?
12. Zu dem Kunden sollen folgende Daten erhoben werden: Kennung, Vorname, Nachname, Straße, PLZ, Ort, Kontonummer, BLZ, Institut, Passwort
13. Ein Kunde wählt mehrere Produkte aus.
14. Ein Produkt kann von mehreren Kunden ausgewählt werden.
15. Die Anzahl, wie oft ein Kunde ein bestimmtes Produkt ausgewählt hat (bzw. bestellen möchte), soll gespeichert werden.
16. Welche Rechnungen gibt es?
17. Bei der Rechnung sind das Datum und die Uhrzeit der Rechnungserstellung wichtig.
18. Eine Rechnung ist immer genau einem Kunden zugeordnet.
19. Ein Kunde kann natürlich mehrere Rechnungen erhalten.
20. Eine Rechnung betrifft in der Regel mehrere Produkte.
21. Ein Produkt ist Bestandteil von mehreren Rechnungen.
22. Auch bei der Rechnung soll die Anzahl gespeichert werden, wie häufig die einzelnen Produkte geordert worden sind.

## Aufgabe 1.1

Ergänzen Sie in Ihrer Indexdatei **htdocs/EigenerName/index.html** die folgende Zeile:

```
<a href = 'Webshop/ws1.php'> Shop anmelden </a><br>
```

## Aufgabe 1.2 (Datenbank anlegen)

1. Setzen Sie zunächst die Anforderungsliste als ER-Modell um.
2. Erzeugen Sie mit phpMyAdmin eine Datenbank mit Namen **EigenerName\_WS**.
3. Legen Sie in dieser Datenbank die benötigten Tabellen fest.
4. Tragen Sie in Ihre Datenbank die folgenden Datensätze ein.

### Hersteller

Name	Web	eMail
Hewlett Packard	www.hp.de	info@hp.de
Siemens	www.siemens.de	support@siemens.de
Medion	www.medion.de	support@medion.de

### Kategorien

Beschreibung
Scanner
Monitore
Drucker

### Produkte

Name	Beschreibung	Preis
HP ScanJet3300C	Flachbettscanner	99
HP ScanJet 2220A	Flachbettscanner	56
HP LaserJet 3477C	Laserdrucker	299
HP LaserJet 7769C	Farblaserdrucker	1590
MD 1772 JC	Monitor	150
MD 6155 AH	17 Zoll TFT Bildschirm	350
MD 1334 S	Flachbettscanner	65
MD 2443 S	Flachbettscanner	76
SI 1221 C	19 Zoll Monitor	200
SI 7822 TFT	21 Zoll TFT-Monitor	569
SI P 3244	DIN A1 Plotter	1590
SI D1121 C	Farblaserdrucker	547

5. Ordnen Sie die Produkte mit Hilfe der Fremdschlüssel sinnvoll den Herstellern und den Kategorien zu!

**Aufgabe 1.3 (Include-Datei anlegen)**

1. Legen Sie in Ihrem Verzeichnis ein Unterverzeichnis **Webshop** an. Erzeugen Sie in diesem Verzeichnis die Datei **hilfs\_funktionen.php**.
2. In dieser Datei **hilfs\_funktionen.php** sollen folgende Funktionen erzeugt werden:
  - a) **function db\_connect (\$dbName)**
    - baut die Verbindung zur Datenbank **\$dbName** auf und
    - liefert den Datenbankhandler **\$dbh** zurück.
  - b) **function db\_query (\$sql, \$dbh)**
    - führt die Anfrage **\$sql** auf der Datenbank aus und
    - liefert das Abfrageergebnis als Array zurück.
  - c) **function db\_show\_query (\$sql, \$dbh)**
    - nutzt die Funktion **db\_query**, um das Ergebnis der Datenbankanfrage tabellarisch darzustellen (vergleiche „Gästebuch anzeigen“).
  - d) **function db\_create\_checkbox (\$sql, \$dbh, \$name)**
    - nutzt die Funktion **db\_query**, um das Ergebnis der Datenbankanfrage tabellarisch darzustellen. Gleichzeitig wird vor der Tabelle eine weitere Spalte mit Checkboxes („**\$name**“) erzeugt, die das Auswählen mehrerer Datensätze erlaubt (vergleiche „Gästebuch löschen“).  
ACHTUNG: als „**\$name**“ muss ein Array übergeben werden!
  - e) **function db\_exist (\$sql, \$dbh)**
    - nutzt die Funktion **db\_query**, um zu überprüfen ob, die Anfrage ein Ergebnis zurückliefert oder nicht.
    - Rückgabewert **true**, wenn **ein** Ergebnis zurückgeliefert wurde.
    - Rückgabewert **false**, wenn **kein** Ergebnis zurückgeliefert wurde.
  - f) **function db\_close(\$dbh)**
    - schließt die Datenbankverbindung.
  - g) **function zeigeParameter()**
    - soll alle **\$\_POST** und alle **\$\_SESSION** Parameter mit Schlüsselwert anzeigen.
3. Erzeugen Sie in diesem Verzeichnis die Datei **shop1.php**. Diese Datei soll die Datei **hilfs\_funktionen.php** einbinden und die Tabelle **Produkte** tabellarisch ausgeben.
4. Testen Sie die Funktionen aus.

**Beispiel für die Funktion „zeigeParameter()“**

```

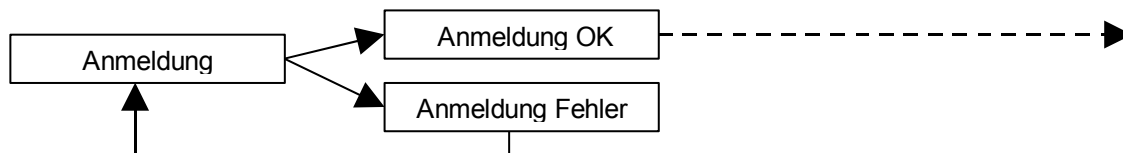
function zeigeParameter()
{
    echo "<h3>Parameter:</h3>";

    // Session Parameter
    echo "<h4>Sitzungsparameter SESSION</h4>";
    foreach ($_SESSION as $schluessel => $wert)
    {
        echo "<br> Schlüssel: $schluessel, Wert: $wert";
    }

    // Übergabeparameter
    echo "<h4>Formularparameter POST</h4>";
    foreach ($_POST as $schluessel => $wert)
    {
        echo "<br> Schlüssel: $schluessel, Wert: $wert";
    }

    echo "<br><hr><br>";
}

```

**Aufgabe 1.4 (Anmeldung)**

Die Anmeldung erfolgt über ein Anmeldeformular, in dem der Benutzer seine Kennung und sein Passwort angeben muss. Stimmen diese Informationen mit einem Datensatz in der Tabelle „Kunde“ überein, wird dem Kunden in einem weiteren Formular seine Login-Kennung angezeigt und bestätigt, dass die Anmeldung erfolgreich war. Auf Knopfdruck gelangt er dann zur Produktauswahl (Aufgabe 2). Andernfalls wird in einem weiteren Formular das Scheitern der Anmeldung angezeigt und der Benutzer gelangt auf Knopfdruck wieder in das Anmeldeformular.

**Aufgaben:**

1. Verändern Sie die Datei **ws1.php** so, dass darin das Anmeldeformular enthalten ist.
2. Realisieren Sie die Formulare zur Anmeldung, Anmeldebestätigung und zum Anmeldefehler ebenfalls in die Datei **ws1.php**!
3. Nach erfolgreicher Anmeldung speichern Sie die Login-Kennung in einer „Session“-Variablen.

**Beispiel: Mehrere Formulare in einer Datei**

```

<?php
if (isset ($_POST["schalter_OK"]))
{
?>
    <html><body>
    <form action='Beispieldatei.php' method='post'>
        ...
        <input type='Submit' name='schalter_weiter' value='Weiter'>
    </form>
    </body></html>
<?php
}
else
{
?>
    <html><body>
    <form action= '<?php echo "$_SERVER['PHP_SELF']"; ?>'
        method= 'post'>
        ...
        <input type = 'Submit' name='schalter_OK' value='OK'>
    </form>
    </body></html>
<?php
}
?>

```

**2. Formular**

Wird nur angezeigt, wenn die Variable „schalter\_OK“ gesetzt ist.

Verweis auf eigene Datei.

**1. Formular**

Wird immer angezeigt, wenn die Variable „schalter\_OK“ nicht gesetzt ist.

**Sessions (Sitzungen)**

Bisher konnten lediglich Formularvariablen von einer Datei zur nächsten mit Hilfe des **\$\_POST**-Arrays weitergeben werden. Wenn man aber beliebig viele Variablen für sämtliche Dateien einer Sitzung bereitstellen möchte, kommt man mit dieser Methode nicht weiter. Dazu gibt es in PHP das Session-Konzept, das wie folgt funktioniert.

1. Mit dem Befehl „**session\_start()**“ wird eine Session (Sitzung) eröffnet.
  - a) Eine eindeutige Session ID (kurz SID) wird generiert.
  - b) Diese SID wird in einem Cookie auf dem Client gespeichert.
  - c) Auf dem Server wird passend zur SID eine Datei angelegt.
2. Mit Hilfe des Arrays **\$\_SESSION** können beliebige Variablen der Session zugeordnet werden. Die Variablen werden in der Serverdatei gespeichert.
3. Will man auf die Session-Variablen auf dem Server zugreifen, muss zunächst die passende SID ermittelt werden. Dies geschieht auch über den Befehl „**session\_start()**“.
4. Anschließend kann auf die Variablen über das Array „**\$\_SESSION**“ zugegriffen werden.

**WICHTIG**

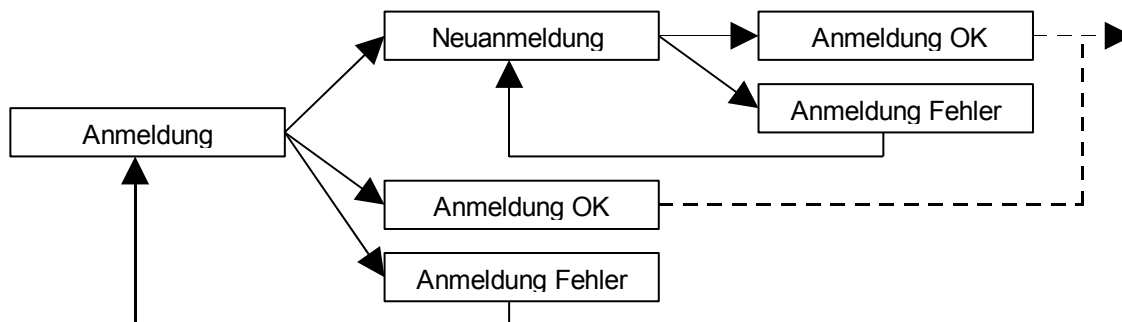
Der Befehl **session\_start()** sollte am Anfang einer Datei stehen!

## Beispiel: Verwenden von Session-Variablen

```
<?php
session_start();           // Session wird eröffnet
$var = "Ein Beispiel";
$_SESSION["var"] = $var;   // Die Variable wird auf der
// Session registriert
?>
```

```
<?php
session_start();           // Die eigene Session wird ermittelt
$var = $_SESSION["var"];   // Die Session-Variable wird
// ausgelesen
?>
```

## Aufgabe 1.5 (Zusatzaufgabe)



Ändern Sie die Anmeldung so ab, dass auch Neuanmeldungen möglich sind, wenn eine Kennung für den Benutzer noch nicht existiert.

### Anmeldung

#### Neuanmeldung

Sie besitzen noch keine Kennung für unseren WEB-Laden  
Dann bitte hier NEU anmelden:

Kennung neu anlegen

#### Anmeldung

Sie sind bereits Kunde unseres WEB-Ladens!  
Dann bitte mit Kennung und Passwort anmelden:

Kennung:

Passwort:

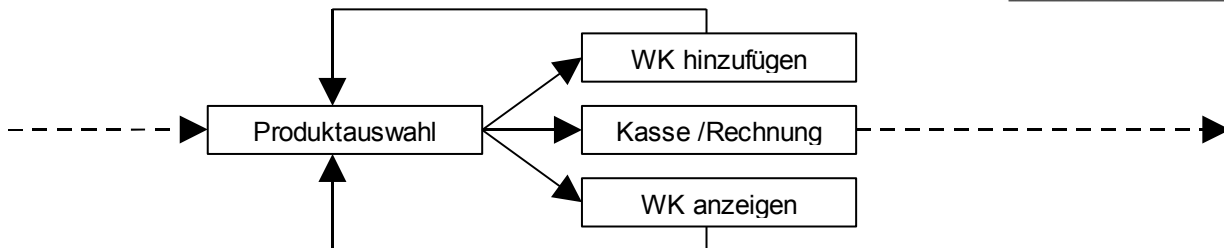
Anmelden Zurücksetzen

## Aufgabe 2 (Produktauswahl / Warenkorb)



Nach der Anmeldung gelangt man zur Produktauswahl.

### Aufgabe 2.1 (Produktauswahl)



In der Produktauswahl werden alle Produkte aufgelistet. Über Checkboxes können die Produkte, die in den Warenkorb (WK) gelegt werden sollen, ausgewählt werden.

#### WK hinzufügen

Über den Schalter **zum Warenkorb hinzufügen** werden die ausgewählten Produkte in den Warenkorb gelegt. Produkte die bereits im WK sind, sollen nicht mehr hinzugefügt werden. In einem zusätzlichen Formular **WK hinzufügen** wird für jedes Produkt angezeigt, ob es korrekt in den WK übernommen worden ist oder sich bereits dort befindet. Über einen Schalter gelangt man aus diesem Formular zurück in die Produktauswahl.

#### WK anzeigen

Der Schalter **Warenkorb anzeigen** öffnet ein Formular, das alle Artikel des Warenkorbs tabellarisch auflistet. Über einen Schalter gelangt man aus diesem Formular ebenfalls zurück in die Produktauswahl.

#### Kasse / Rechnung

Über den Schalter **zur Kasse** gelangt man in das Rechnungsformular, hier werden die Produkte des Warenkorbs angezeigt. Datum und Uhrzeit (aktuelle Stunde) werden ermittelt und damit ein neuer Rechnungsdatensatz angelegt. Alle Produkte aus dem Warenkorb werden der Rechnung zugeordnet. Der Warenkorb selbst wird geleert. Über einen Schalter gelangt man zum Verabschiedungsformular (Aufgabe 3).

1. Erzeugen Sie eine Datei `produktauswahl.php`
2. Realisieren Sie die Formulare **Produktauswahl**, **Warenkorb hinzufügen**, **Warenkorb anzeigen** und **Rechnung** in dieser Datei!

#### Datumsermittlung in PHP

```

<?php
    $stunde = date("H"); // H = Stunde in 24-Stunden Format
    $datum = date("d.m.Y"); // d = Tag, m = Monat, Y = Jahr
    echo "$stunde $datum"; // Ausgabe: 15 31.12.2011
?>
  
```



## Aufgabe 2.2 (Zusatzaufgabe)

1. Realisieren Sie in der Produktauswahl eine Filterfunktion, mit der es möglich ist nur noch Produkte einzelner Kategorien anzuzeigen.
2. Für die Auswahlliste der Kategorien sollte eine eigene Funktion in der Datei **hilfs\_funktionen.php** angelegt werden:
  - a) **function db\_create\_list (\$sql, \$dbh, \$size, \$name)**

### Produktauswahl

Sie sind mit der Kennung *willi* angemeldet!

Filter: Monitore Filter anwenden

<input type="checkbox"/>	1	HP	Flachbrettscanner	99	Scanner	Hewlett Packard	www.hp.de	info@hp.de
<input type="checkbox"/>	2	HP	Flachbrettscanner	56	Scanner	Hewlett Packard	www.hp.de	info@hp.de
<input type="checkbox"/>	8	MD 1334 S	Flachbrettscanner	65	Scanner	Medion	www.medion.de	support@medion.de
<input type="checkbox"/>	9	MD 2443 S	Flachbrettscanner	76	Scanner	Medion	www.medion.de	support@medion.de
<input type="checkbox"/>	6	MD 1772 JC	Monitor	150	Monitore	Medion	www.medion.de	support@medion.de
<input type="checkbox"/>	7	MD 6155 AH	17 Zoll TFT Bildschirm	350	Monitore	Medion	www.medion.de	support@medion.de
<input type="checkbox"/>	10	SI 1221 C	19 Zoll Monitor	200	Monitore	Siemens	www.siemens.de	support@siemens.de
<input type="checkbox"/>	11	SI 7822 TFT	21 Zoll TFT-Monitor	569	Monitore	Siemens	www.siemens.de	support@siemens.de
<input type="checkbox"/>	3	HP LaserJet 3477C	Laserdrucker	299	Drucker	Hewlett Packard	www.hp.de	info@hp.de
<input type="checkbox"/>	4	HP DeskJet 855C	Farbtintendrucker	150	Drucker	Hewlett Packard	www.hp.de	info@hp.de
<input type="checkbox"/>	5	HP LaserJet 7769C	Farblaserdrucker	1590	Drucker	Hewlett Packard	www.hp.de	info@hp.de
<input type="checkbox"/>	12	SI P 3244	DIN A1 Plotter	1590	Drucker	Siemens	www.siemens.de	support@siemens.de
<input type="checkbox"/>	13	SI D 1121 C	Farblaserdrucker	547	Drucker	Siemens	www.siemens.de	support@siemens.de

zum Warenkorb hinzufügen
Warenkorb anzeigen
zur Kasse
Zurücksetzen

3. Ermöglichen Sie es dem Kunden ein Produkt auch mehrfach zu bestellen.  
Dazu erlauben Sie dem Kunden in dem Formular **Warenkorb anzeigen** die jeweiligen Anzahlen selbständig anzugeben bzw. zu verändern.

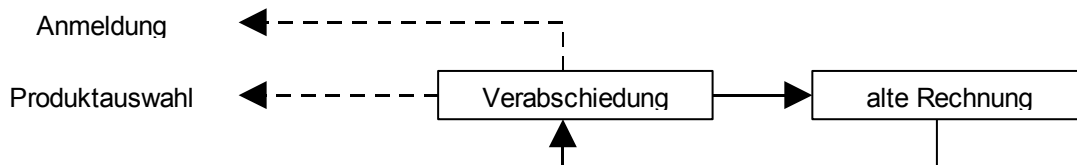


## Aufgabe 3 (Rechnung / Verabschiedung)



Nach dem Erzeugen der Rechnung gelangt man in das Verabschiedungsformular.

### Aufgabe 3.1 (Verabschiedung)



Auf dem Verabschiedungsformular wird für den Einkauf gedankt.

Des Weiteren befinden sich auf dem Formular zwei Links.

1. Ein Link führt zurück zur Anmeldung, um z.B. einen weiteren Einkauf unter einer anderen Kennung durchzuführen.
2. Der zweite Link führt zurück zur Produktauswahl, um einen weiteren Einkauf unter derselben Kennung zu tätigen.

### Aufgabe 3.2 (Alte Rechnungen)

In dem Verabschiedungsformular soll der Kunde die Möglichkeit erhalten alle alte Rechnungen einzusehen. Über den Schalter **alte Rechnungen** gelangt der Kunde in ein Formular, das alle alten Rechnungen tabellarisch auflistet. Das Datum, die Uhrzeit und der Gesamtbetrag der Rechnung sollen dabei angezeigt werden. Über einen weiteren Schalter kann der Kunde das Formular verlassen und zum Verabschiedungsformular zurückkehren.

3. Erzeugen Sie eine Datei **verabschiedung.php**
4. Realisieren Sie die Formulare **Verabschiedung** und **Alte Rechnungen** in dieser Datei!

### Aufgabe 3.3 (Zusatzaufgabe)

Verändern Sie das Formular **Alte Rechnungen** so, dass auch die Einzelprodukte und deren Preis zur jeweiligen Rechnung angezeigt werden.