



ENTSCHEIDUNGEN DURCH IF-ABFRAGEN

VERSCHIEDENE VERGLEICHSOPERATOREN VERWENDEN

ELSE UND ELSEIF

```
if($geschlecht == "m") {
    print "<h1>Hallo Herr ".$nachname."!</h1>\n";
} elseif ($geschlecht == "w") {
    print "<h1>Hallo Frau ".$nachname."!</h1>\n";
} else {
    print "Hi du";
}
```

```
if($geschlecht == "m") {
    print "<h1>Hallo Herr ".$nachname."!</h1>\n";
} else {
    print "<h1>Hallo Frau ".$nachname."!</h1>\n";
}
```

ÜBUNG

- Schreibe ein Programm, das ermittelt, ob die Warenbestände für ein bestimmtes Produkt auf 0 gefallen sind und gegebenenfalls eine Meldung ausgibt, dass der Artikel nicht mehr verfügbar ist.
- 2. Erstelle zunächst ein Array, in dem sich ein Produkt mit dem zugehörigen Preis und die Anzahl der verfügbaren Artikel befindet.

Schreibe nun ein Programm, das:

- 1. Den Käufer darüber informiert, dass der Artikel nicht verfügbar ist, falls der Warenbestand auf 0 steht.
- 2. Ausgibt, dass das Produkt versandkostenfrei geliefert wird, falls mindestens ein Artikel vorrätig ist und falls der Preis bei mindestens 20 Euro liegt
- 3. Anzeigt, dass für die Lieferung 5 Euro Versandkosten anfallen, falls der Artikel verfügbar ist, der Preis jedoch unter 20 Euro liegt.



DIE FUNKTIONALITÄT EINES PROGRAMMS DURCH SCHLEIFEN ERWEITERN

KOPFGESTEUERTE SCHLEIFEN: WHILE UND FOR

```
Hallo
Hallo
Hallo
Hallo
Hallo
Hallo
Hallo
Hallo
Hallo
```

Hallo

```
<?php

$i = 0;
while ($i < 10) {
    print "<p>Hallo\n";
    $i++;
}
```

```
<?php

$i = 0;
while ($i < 10) {
    $i++;
    print $i*$i."<br>}
}

1
4
9
16
25
36
49
64
81
100
```

```
<?php
                                              Produkt: Semmel Preis: 1.99 Euro
$produkt[0]['Produktname'] = "Semmel";
$produkt[0]['Preis'] = 1.99;
                                              Produkt: Laugenbrezl Preis: 2.99 Euro
$produkt[0]['Anzahl'] = 6;
$produkt[1]['Produktname'] = "Kornspitz";
$produkt[1]['Preis'] = 1.49;
$produkt[1]['Anzahl'] = 0;
$produkt[2]['Produktname'] = "Laugenbrezl";
$produkt[2]['Preis'] = 2.99;
$produkt[2]['Anzah1'] = 8;
$i = 0;
/* Hier muss ich vorher wissen wie viele Einträge das Array hat */
while ($i < 3) {
    if($produkt[$i]['Anzahl'] > 0) {
        print "Produkt: " .$produkt[$i]['Produktname']. " Preis: "
            .$produkt[$i]['Preis']. " Euro\n";
    $i++;
```

FUSSGESTEUERTE SCHLEIFEN: DO WHILE

Hallo
Hallo
Hallo
Hallo
Hallo
Hallo

Hallo

Hallo

Hallo

Hallo

```
<?php
$i = 0;

do {
    print "<p>Hallo\n";
    $i++;
} while ($i < 10);</pre>
```

```
<?php

$anzahl = 50;
$bedingung = true;

do {
    print "<p>Aktuelle Anzahl: " .$anzahl. "\n";
    $anzahl--;
    $bedingung = ($anzahl > 20);
} while ($bedingung);
```

Aktuelle Anzahl: 49
Aktuelle Anzahl: 48
Aktuelle Anzahl: 47
Aktuelle Anzahl: 46
Aktuelle Anzahl: 45
Aktuelle Anzahl: 44
Aktuelle Anzahl: 43
Aktuelle Anzahl: 42
Aktuelle Anzahl: 42

Aktuelle Anzahl: 40

Aktuelle Anzahl: 39

Aktuelle Anzahl: 38

Altrelle Anzahl: 37

Aktuelle Anzahl: 50

FOREACH-SCHLEIFEN FÜR DIE ARBEIT MIT ARRAYS

```
$produkt = array(
    1, "Bohrmaschine", 45, "Kraftvolle Bohrmaschine für Handwerker", 23
);

foreach ($produkt as $item) {
    print $item . "<br>\n";
}

1
Bohrmaschine
45
Kraftvolle Bohrmaschine für Handwerker
23
```

```
<?php
$produkt[0]['Produktname'] = "Bohrmaschine";
$produkt[0]['Preis'] = 45.99;
$produkt[0]['Anzahl'] = 6;
$produkt[1]['Produktname'] = "Kreissäge";
$produkt[1]['Preis'] = 79.99;
$produkt[1]['Anzahl'] = 0;
$produkt[2]['Produktname'] = "Bandschleifer";
$produkt[2]['Preis'] = 85.99;
$produkt[2]['Anzahl'] = 11;
                                     Produkt: Bohrmaschine Preis: 45.99 Euro.
foreach($produkt as $inhalt) {
                                     Produkt: Bandschleifer Preis: 85.99 Euro.
   if($inhalt['Anzahl'] > 0) {
        print "Produkt: " .$inhalt['Produktname'].
            " Preis: " .$inhalt['Preis']. " Euro.\n";
```

```
<?php
$produkt[0]['Produktname'] = "Bohrmaschine";
$produkt[0]['Preis'] = 45.99;
$produkt[0]['Anzahl'] = 6;
$produkt[1]['Produktname'] = "Kreissäge";
$produkt[1]['Preis'] = 79.99;
$produkt[1]['Anzahl'] = 0;
$produkt[2]['Produktname'] = "Bandschleifer";
$produkt[2]['Preis'] = 85.99;
                                       Bohrmaschine
$produkt[2]['Anzahl'] = 11;
                                      45.99
foreach($produkt as $ebene1) {
    foreach($ebene1 as $ebene2) {
                                       Kreissäge
        print $ebene2. "<br>\n";
                                      79.99
                                       0
    print "<br>";
                                       Bandschleifer
                                       85.99
                                       11
```

FOREACH-SCHLEIFEN FÜR DIE ARBEIT MIT ARRAYS

```
$produkt[0]['Produktname'] = "Bohrmaschine";
$produkt[0]['Preis'] = 45.99;
$produkt[0]['Anzahl'] = 6;
$produkt[1]['Produktname'] = "Kreissäge";
$produkt[1]['Preis'] = 79.99;
$produkt[1]['Anzahl'] = 0;
$produkt[2]['Produktname'] = "Bandschleifer";
$produkt[2]['Preis'] = 85.99;
$produkt[2]['Anzahl'] = 11;
foreach($produkt as $ebene1) {
    foreach($produkt as $name => $ebene2) {
        print $name . ": " .$ebene2. "<br>\n";
    }
    print "<br>
}
```

Produktname: Bohrmaschine

Preis: 45.99 Anzahl: 6

Produktname: Kreissäge

Preis: 79.99 Anzahl: 0

Produktname: Bandschleifer

Preis: 85.99 Anzahl: 11

ÜBUNG

- ❖ Erstelle drei Programme, die von 1 bis 10 zählen, verwende hierfür jeweils einmal eine while-, eine for- und eine do-while-Schleife
- ❖ Erstelle ein zusammengesetztes Array für das Sortiment eines Obst- und Gemüsehändlers mit Äpfel, Birnen, Tomaten und Zucchini. Die erste Ebene soll einen numerischen Index haben. Die zweite soll hingegen assoziativ sein (mit den Bezeichnungen Produkt, Preis und Sonderangebot). Bei der Angabe zum Produkt handelt es sich um eine Zeichenkette mit der entsprechenden Obst- oder Gemüsesorte, beim Preis um eine Zahl und die Angabe Sonderangebot soll eine boolesche Variable sein
 - Erstelle eine foreach-Schleife, die den Produktnamen und den Preis angibt. Sollte es ich dabei um ein Sonderangebot handeln, soll vor diesen Angaben "Achtung Sonderangebot!" erscheinen.
- Ergänze das vorherige Beispiel um die Ausgabe des entsprechenden Feldnamens. Das Sonderangebot kannst du weg lassen.



FUNKTIONEN IN PHP

DER AUFBAU EINER FUNKTION IN PHP

```
<?php

function begruessung() {
    print "Guten Morgen";
}
begruessung();</pre>
```

```
function begruessung($text) {
    print $text;
}
begruessung("Guten Morgen");
```

Guten Morgen

```
function begruessung($text, $ansprache) {
   print $text . " " . $ansprache;
}

$gruss = "Guten Morgen";
$leser = "Herr Müller";
begruessung($gruss, $leser);
```

Guten Morgen Herr Müller

RÜCKGABEWERTE DER FUNKTIONEN

```
function verdoppelung($wert) {
    $wert = $wert*2;
    return $wert;
}
print verdoppelung(5);

/* oder */
function verdoppelungAlternative($wert) {
    return ($wert*2);
}
print verdoppelungAlternative(5);
```

```
<?php
function verdoppelung_quadrat($wert) {
    $doppelt = $wert*2;
    $quadrat = $wert*$wert;
    $ergebnis = array(
                                            Der doppelte Wert dieser Zahl beträgt 6.
        'Verdopplung' => $doppelt,
                                            Das Quadrat dieser Zahl beträgt 9.
        'Quadrat' => $quadrat
    return $ergebnis;
$rueckgabewert = verdoppelung_quadrat(3);
$verdoppelter wert = $rueckgabewert['Verdopplung'];
$wert zum quadrat = $rueckgabewert['Quadrat'];
print "Der doppelte Wert dieser Zahl beträgt " . $verdoppelter_wert .".<br>\n";
print "Das Quadrat dieser Zahl beträgt " . $wert zum quadrat .".<br>\n";
```

EINE FUNKTION IN DAS PHP-PROGRAMM EINBINDEN

File: verdopplung.php

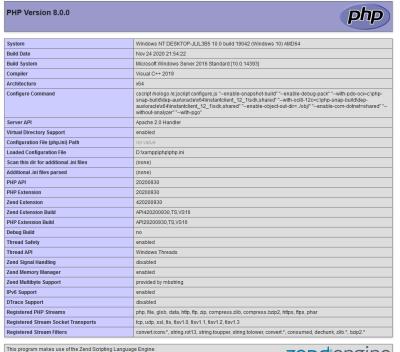
```
<?php
function verdopplung($wert) {
    return ($wert*2);
}</pre>
```

T Hauptfile

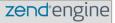
```
<?php
   include ("verdopplung.php");
   print verdopplung(4);</pre>
```

FUNKTIONEN AUS DER PHP-BIBLIOTHEK VERWENDEN





This program makes use of the Zend Scripting Language Engine: Zend Engine v4.0.0-dev, Copyright (c) Zend Technologies



Coders.Bay < 16 />

ÜBUNG

- 1. Erstelle eine Funktion, die die ersten zehn Potenzen einer Zahl berechnet. Dazu kommt der Operator ** zum Einsatz. Diesem wird die Basis vorangestellt und darauf folgt der Exponent. Das Hauptprogramm soll dabei den Ausgangswert übermitteln. Die Rückgabe erfolgt durch ein Array. Die Ausgabe des Arrays erfolgt im Hauptprogramm.
- 2. Führe nun auch die Ausgabe des Arrays in einer eigenen Funktion durch. Lagere beide Funktionen in eine separate Datei aus und binde diese in das Hauptprogramm ein
- 3. Suche in der Online-Dokumentation eine Funktion, um die Quadratwurzel einer Zahl zu bestimmen und eine weitere Funktion, mit der du eine Zufallsvariable generierst. Informiere dich über die Anwendung dieser beiden Funktionen. Erstelle nun ein Programm, dass die Wurzel eines zufälligen Werts berechnet und sowohl den Ausgangswert als auch das Ergebnis ausgibt.



ENDE

QUELLE: PHP & MYSQL FÜR EINSTEIGER ISBN: 978-3-96645-009-6