CODERS.BAY

PHP 02

CODERS.BAY

Entscheidungen durch if-Abfragen

Verschiedene Vergleichsoperatoren verwenden

```php
<?php

$var1 = 2;
$var2 = 7;

if($var1 < 5 || $var2 == 7) {
    print "Variable 1 ist kleiner als 5 oder Variabel 2 beträgt 7";
}
```

Variable 1 ist kleiner als 5 oder Variabel 2 beträgt 7

```php
<?php

$var1 = 2;
$var2 = 7;

if(!($var2 < 5)) {
    print "Variable 2 ist nicht kleiner als 5";
}
```

Variable 1 ist nicht kleiner als 5

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

1. 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.

CODERS.BAY

Die Funktionalität eines Programms durch Schleifen erweitern

Kopfgesteuerte Schleifen: while und for

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

```php
<?php

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

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

```php
<?php

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

```
Produkt: Semmel Preis: 1.99 Euro

Produkt: Laugenbrezl Preis: 2.99 Euro
```

```php
<?php

$produkt[0]['Produktname'] = "Semmel";
$produkt[0]['Preis'] = 1.99;
$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]['Anzahl'] = 8;

$i = 0;
/* Hier muss ich vorher wissen wie viele Einträge das Array hat */
while ($i < 3) {
    if($produkt[$i]['Anzahl'] > 0) {
        print "<p>Produkt: " .$produkt[$i]['Produktname']. " Preis: "
            .$produkt[$i]['Preis']. " Euro</p>\n";
    }
    $i++;
}
```

Fussgesteuerte Schleifen: do while

Hallo
Hallo
Hallo
Hallo
Hallo
Hallo
Hallo
Hallo
Hallo
Hallo

```php
<?php

$i = 0;

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

```php
<?php

$anzahl = 50;
$bedingung = true;

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

Aktuelle Anzahl: 50
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: 41
Aktuelle Anzahl: 40
Aktuelle Anzahl: 39
Aktuelle Anzahl: 38
Aktuelle Anzahl: 37

Foreach-Schleifen für die Arbeit mit Arrays

```php
<?php

$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
<?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;

foreach($produkt as $inhalt) {
    if($inhalt['Anzahl'] > 0) {
        print "<p>Produkt: " .$inhalt['Produktname'].
            " Preis: " .$inhalt['Preis']. " Euro.</p>\n";
    }
}
```

Produkt: Bohrmaschine Preis: 45.99 Euro.

Produkt: Bandschleifer Preis: 85.99 Euro.

```php
<?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;

foreach($produkt as $ebene1) {
    foreach($ebene1 as $ebene2) {
        print $ebene2. "<br>\n";
    }
    print "<br>";
}
```

```
Bohrmaschine
45.99
6

Kreissäge
79.99
0

Bandschleifer
85.99
11
```

Foreach-Schleifen für die Arbeit mit Arrays

```php
<?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;
$produkt[2]['Anzahl'] = 11;
foreach($produkt as $ebene1) {
    foreach($ebene1 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.

CODERS.BAY

Funktionen in PHP

Der Aufbau einer Funktion in PHP

```php
<?php

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

Guten Morgen

```php
<?php

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

```php
<?php

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

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

Guten Morgen Herr Müller

Rückgabewerte der Funktionen

```php
<?php

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

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

```php
<?php

function verdoppelung_quadrat($wert) {
    $doppelt = $wert*2;
    $quadrat = $wert*$wert;
    $ergebnis = array(
        'Verdopplung' => $doppelt,
        '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";
```

Der doppelte Wert dieser Zahl beträgt 6.
Das Quadrat dieser Zahl beträgt 9.

Eine Funktion in das PHP-Programm einbinden

❖ File: verdopplung.php

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

T Hauptfile

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

Funktionen aus der PHP-Bibliothek verwenden

```php
<?php

    phpinfo();
```

PHP Version 8.0.0

System	Windows NT DESKTOP-JLIL3B5 10.0 build 19042 (Windows 10) AMD64
Build Date	Nov 24 2020 21:54:22
Build System	Microsoft Windows Server 2016 Standard [10.0.14393]
Compiler	Visual C++ 2019
Architecture	x64
Configure Command	cscript /nologo /e:jscript configure.js "--enable-snapshot-build" "--enable-debug-pack" "--with-pdo-oci=c:\php-snap-build\dep-aux\oracle\x64\instantclient_12_1\sdk,shared" "--with-oci8-12c=c:\php-snap-build\dep-aux\oracle\x64\instantclient_12_1\sdk,shared" "--enable-object-out-dir=../obj/" "--enable-com-dotnet=shared" "--without-analyzer" "--with-pgo"
Server API	Apache 2.0 Handler
Virtual Directory Support	enabled
Configuration File (php.ini) Path	*no value*
Loaded Configuration File	D:\xampp\php\php.ini
Scan this dir for additional .ini files	(none)
Additional .ini files parsed	(none)
PHP API	20200930
PHP Extension	20200930
Zend Extension	420200930
Zend Extension Build	API420200930,TS,VS16
PHP Extension Build	API20200930,TS,VS16
Debug Build	no
Thread Safety	enabled
Thread API	Windows Threads
Zend Signal Handling	disabled
Zend Memory Manager	enabled
Zend Multibyte Support	provided by mbstring
IPv6 Support	enabled
DTrace Support	disabled
Registered PHP Streams	php, file, glob, data, http, ftp, zip, compress.zlib, compress.bzip2, https, ftps, phar
Registered Stream Socket Transports	tcp, udp, ssl, tls, tlsv1.0, tlsv1.1, tlsv1.2, tlsv1.3
Registered Stream Filters	convert.iconv.*, string.rot13, string.toupper, string.tolower, convert.*, consumed, dechunk, zlib.*, bzip2.*

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

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