

# Lösungen zu Übungsaufgaben

Quelle: Maurice (2015); Theis (2016)

## Teil: Datentypen, Operatoren, Kontrollstrukturen

### Kapitel: "4. Datentypen"

#### Aufgabe 4.1:

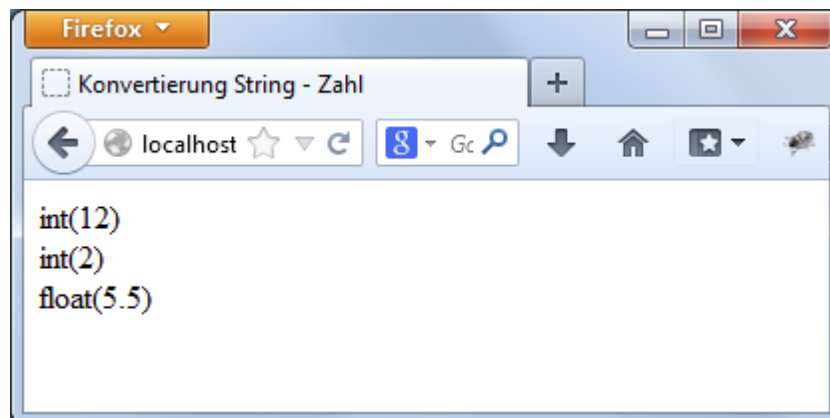
Welche Ausgabe wird durch das folgende Programm erzeugt?

```
01 $str1 = "10 Eier";  
02 $str2 = "Schachtel mit 10 Eiern";  
03 $str3 = "3.5 Äpfel";  
04 $erg1 = $str1 + 2;  
05 var_dump($erg1);  
06 echo "<br />\n";  
07 $erg2 = $str2 + 2;  
08 var_dump($erg2);  
09 echo "<br />\n";  
10 $erg3 = $str3 + 2;  
11 var_dump($erg3);
```

#### Lösungshinweis:

##### *Konvertierung von String in Zahlen:*

Bei der Konvertierung in eine Zahl gilt folgendes Prinzip: Wenn ein String mit einer Zahl beginnt, wird diese genommen und der Rest verworfen. Wenn der String nicht mit einer Zahl beginnt, wird der String zu 0 konvertiert. Das gilt aber nur für das Ergebnis, der Inhalt der Variablen selbst bleibt unverändert. In Zeile 4 wird »10 Eier« + 2 berechnet. Das Ergebnis ist 12. Das Ergebnis von »Schachtel mit 10 Eiern« + 2 ist hingegen 2. Denn »Schachtel mit 10 Eiern« beginnt nicht mit einer Zahl und wird als 0 ausgewertet. »3.5 Äpfel« + 2 (Zeile 10) ergibt dann entsprechend 5.5 und ist ein Float. Die folgende Abbildung zeigt die Ausgabe.



#### Aufgabe 4.2:

Ändern Sie das Beispiel *zufallsbilder.php* so ab, dass zufällig einer von mehreren Texten angezeigt wird. Dafür müssen Sie natürlich zuerst ein Array mit mehreren Strings definieren!

```
// zufallsbilder.php

<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8" />
<title>Zufallsbilder</title>
</head>
<body>
<?php
    $bilder = array("blumen.jpg", "boot.jpg",
                    "landschaft.jpg", "stadt_am_meer.jpg",
                    "strand.jpg");
    $max = count($bilder) - 1;
    $zufallszahl = rand(0, $max);
    echo "<img src='$bilder[$zufallszahl]' height='200'
                                         width='150' />";
?>
</body>
</html>
```

#### Lösung:

```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8" />
<title>Zufallstexte</title>
</head>
<body>
<?php
    $texte = array("Lorem ipsum dolor sit amet",
                    "Überall dieselbe alte Leier. Das Layout ist fertig, der Text lässt auf sich warten.",
                    "Weit hinten, hinter den Wortbergen, fern der Länder Vokalien und Konsonantien leben
                    die Blindtexte.",
                    "Manchmal benutzt man Worte wie Hamburgefonts, Rafgenduks oder Handgloves, um
                    Schriften zu testen." );

    $max = count($texte)-1;
    $zufallszahl = rand(0, $max);
    echo $texte[$zufallszahl];
?>
</body>
```

</html>

## Kapitel: "5. Operatoren"

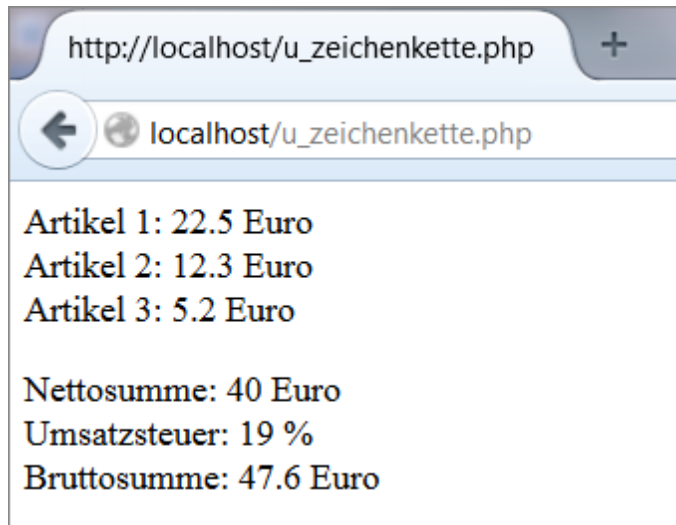
### Aufgabe 5.1:

Berechnen Sie in einem PHP-Programm den Bruttopreis eines Einkaufs. Es werden insgesamt drei Artikel eingekauft. Die Nettopreise der einzelnen Artikel betragen 22,50 €, 12,30 € und 5,20 €.

Der Bruttopreis berechnet sich bekanntlich aus dem Nettopreis zuzüglich 19 % Umsatzsteuer.

In die Berechnung muss also der Faktor 1.19 eingehen.

Die Ausgabe des Programms im Browser sollte wie folgt aussehen.



### Lösung:

```
<html>
<body>
<?php
    /* Artikelpreise */
    $art1 = 22.5;
    $art2 = 12.3;
    $art3 = 5.2;

    /* Berechnung */
    $ust_prozent = 19;
    $ust_faktor = 1 + $ust_prozent / 100;
    $netto = $art1 + $art2 + $art3;
    $brutto = $netto * $ust_faktor;

    /* Ausgabe */
    echo "<p>Artikel 1: $art1 Euro<br />";
    echo "Artikel 2: $art2 Euro<br />";
    echo "Artikel 3: $art3 Euro</p>";
    echo "<p>Nettosumme: $netto Euro<br />";
    echo "Umsatzsteuer: $ust_prozent %<br />";
```

```

    echo "Bruttosumme: $brutto Euro</p>";
?>
</body>
</html>

```

## Kapitel: "6. Kontrollstrukturen"

### Aufgabe 6.1:

Erweitern Sie das folgende Programm mit den unterschiedlichen Begrüßungen je nach Uhrzeit so, dass immer der Seitenhintergrund unterschiedlich eingefärbt wird! Die Farbe des Seitenhintergrunds bestimmen Sie über CSS. Wenn Sie im head-Bereich Folgendes eingeben ...

```

<style>
body {
background-color: yellow;
}
</style>

```

... wird beispielsweise der Hintergrund der Seite gelb.

```

<?php
    date_default_timezone_set("Europe/Berlin");
    $uhrzeit = date("H");
    if ($uhrzeit < 5 || $uhrzeit > 20) {
        $gruss = "Gute Nacht";
    } elseif ($uhrzeit < 11) {
        $gruss = "Guten Morgen";
    } elseif ($uhrzeit < 15) {
        $gruss = "Guten Mittag";
    } elseif ($uhrzeit < 18) {
        $gruss = "Guten Nachmittag";
    } else {
        $gruss = "Guten Abend";
    }
?>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8" />
<title>Unterschiedliche Begrüßung </title>
</head>
<body>
<?php
    echo $gruss;
?>
</body>
</html>

```

### Lösung:

Um den Seitenhintergrund unterschiedlich einzufärben, wird eine neue Variable benötigt. Im Beispiel heißt sie \$hg. Parallel zur Variablen \$gruss wird sie jeweils befüllt, aber mit unterschiedlichen Farben. Die Farbangabe wird dann innerhalb des Kopfbereichs ausgegeben:

```
body {  
    background-color: <?php echo $hg; ?>;  
}
```

Das gesamte Beispiel sieht folgendermaßen aus:

```
<?php  
    date_default_timezone_set("Europe/Berlin");  
    $uhrzeit = date("H");  
    if ($uhrzeit < 5 || $uhrzeit > 20) {  
        $gruss = "Gute Nacht";  
        $hg = "#886CEA";  
    } elseif ($uhrzeit < 11) {  
        $gruss = "Guten Morgen";  
        $hg = "#66EDF0";  
    } elseif ($uhrzeit < 15) {  
        $gruss = "Guten Mittag";  
        $hg = "#D1F066";  
    } elseif ($uhrzeit < 18) {  
        $gruss = "Guten Nachmittag";  
        $hg = "#F09366";  
    } else {  
        $gruss = "Guten Abend";  
        $hg = "#F066DF";  
    }  
?  
  
<!DOCTYPE html>  
<html>  
<head>  
<meta charset="UTF-8" />  
<title>Unterschiedliche Begrüßung </title>  
<style>  
body {  
    background-color: <?php echo $hg; ?>;  
}  
</style>  
</head>  
<body>  
    <?php  
        echo $gruss;  
    ?>  
</body> </html>
```

## Aufgabe 6.2:

Schreiben Sie ein Programm (Datei *u\_tabelle.php*), in dem mithilfe zweier geschachtelter for-Schleifen das »kleine Einmaleins« in einer Tabelle ausgegeben wird. Die Ausgabe soll wie folgt aussehen.



1	2	3	4	5	6	7	8	9	10
2	4	6	8	10	12	14	16	18	20
3	6	9	12	15	18	21	24	27	30
4	8	12	16	20	24	28	32	36	40
5	10	15	20	25	30	35	40	45	50
6	12	18	24	30	36	42	48	54	60
7	14	21	28	35	42	49	56	63	70
8	16	24	32	40	48	56	64	72	80
9	18	27	36	45	54	63	72	81	90
10	20	30	40	50	60	70	80	90	100

Lösung:

Das Programm *u\_tabelle.php*:

```
<html>
<body>
<table border="1">
<?php
    for ($z=1; $z<=10; $z++)
    {
        echo "<tr>";
        for ($s=1; $s<=10; $s++)
        {
            $erg = $z * $s;
            echo "<td align='right'>$erg</td>";
        }
        echo "</tr>";
    }
?>
</table>
</body>
</html>
```

### Aufgabe 6.3:

Erstellen Sie ein kleines Computerspiel (Datei *u\_while.php*). Zwei Spieler würfeln gegeneinander (Zufallszahlengenerator). Die Würfe jedes Spielers sollen addiert werden. Sobald einer der beiden Spieler oder beide Spieler nach einer Spielrunde den Wert 25 erreicht oder überschritten haben, ist das Spiel zu Ende

### Lösung:

Das Programm *u\_while.php*:


```
<html>
<body>
<?php
    /* Initialisierung */

    $spieler1 = 0;
    $spieler2 = 0;

    /* Spiel */
    echo "<table border='1'>";
    echo "<tr><td>Spieler 1</td><td>Spieler 2</td></tr>";
    while ($spieler1 < 25 && $spieler2 < 25)
    {
        $spieler1 = $spieler1 + rand(1,6);
        $spieler2 = $spieler2 + rand(1,6);
        echo "<tr><td align='right'>$spieler1</td>"
            . "<td align='right'>$spieler2</td></tr>";
    }
    echo "</table>";

    /* Ende */
    if ($spieler1 > $spieler2)
        echo "<p>Spieler 1 hat gewonnen</p>";
    else if ($spieler1 < $spieler2)
        echo "<p>Spieler 2 hat gewonnen</p>";
    else
        echo "<p>Unentschieden</p>";

?>
</body>
</html>
```



Spieler 1	Spieler 2
5	4
6	9
9	14
13	19
17	20
19	24
21	27

Spieler 2 hat gewonnen