

6 Felder/ Arrays

Bislang wurden in den Beispielen Variablen verwendet, die jeweils nur einen einzelnen Wert speichern. In diesem Kapitel lernen Sie Arrays kennen. Arrays sind spezielle Variablen, die nicht nur einen, sondern beliebig viele Werte (auch unterschiedlichen Datentyps) speichern können. Im Sprachgebrauch von Entwicklern ist die englische Bezeichnung „Arrays“ üblich. Mitunter werden Arrays auch Feldvariablen oder kurz Felder genannt. (PHP 7.0 – Dynamische Webseiten erstellen, Herdt Verlag, 2016)

- Im numerisch indizierten Array werden die einzelnen Werte (value) innerhalb des Array über einen Index, eine fortlaufende Nummer, angesprochen.
- Im assoziativen Array werden die einzelnen Werte (value) innerhalb des Arrays über einen eindeutigen Schlüssel, key genannt, angesprochen.

6.1 Numerisch indizierte Felder

→ Testen Sie folgendes Beispiel selbst:

```
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8">
</head>
<body>
<?php
    $tp = array(17.5, 19.2, 21.8, 21.6, 17.5);
    $tp[5] = 20.2;
    $tp[6] = 16.6;

    for($i=0; $i<=6; $i = $i+1)
        echo "Temperatur $i: $tp[$i]<br>";
?>
</body>
</html>
```

Einfaches numerisch indiziertes Array
`$staedte = array("Frankfurt",
 "Berlin", "Bern"); //erstellen`

Index	0	1	2
Wert	Frankfurt	Berlin	Bern

`$staedte[] = "Graz"; //hinzufuegen`
`$staedte[] = "Rom";`
`$auswahl = $staedte[2]; //zugreifen`

Array mit for-Schleife durchlaufen

→ Was geschieht, wenn Sie selber indizieren? Testen Sie:

```
$staedte = array(1 => "Frankfurt", "Berlin",  

    "Bern");  

$staedte[34] = "Graz";  

$staedte[] = "Rom";  

$staedte[5] = "Hamburg";  

$staedte[] = "Köln";  

echo "<pre>";  

print_r($staedte);  

echo "</pre>";
```

```
Array
(
    [1] => Frankfurt
    [2] => Berlin
    [3] => Bern
    [34] => Graz
    [35] => Rom
    [5] => Hamburg
    [36] => Köln
)
```

Lösung

Das Array `$staedte` wird über das Schlüsselwort `array()` angelegt und mit Werten gefüllt. Sie weisen einen Indexwert zu, indem Sie den Indexwert zusammen mit den Zeichen `=>` vor den gewünschten Eintrag schreiben. In diesem Fall wird dem Eintrag `Frankfurt` der Index `1` zugewiesen. Da es sich um den ersten Eintrag im Array handelt, wird der Indexwert `0` nicht vergeben. Bei einer automatischen Indizierung wird die Nummerierung einfach nach dem höchsten vorhandenen Wert fortgesetzt, also keine eventuell vorhandenen Lücken aufgefüllt. `Berlin` erhält automatisch den Index `2`, `Bern` den Index `3`.

6.2 Assoziative Felder / Assoziative Arrays

Eine andere Art von Arrays sind assoziative Arrays. Bei diesen werden für den Zugriff keine fortlaufenden Indizes benutzt, sondern Schlüssel, über deren Wert auf die einzelnen Werte zugegriffen werden kann. Der Vorteil assoziativer Arrays besteht darin, dass Sie zwei Informationen (ein Informationspaar) ablegen können. Die Schlüssel sind frei wählbar und tragen häufig eine Bedeutung – im Gegensatz zu einem numerischen und meist nicht bedeutungstragenden Index. (PHP 7.0 – Dynamische Webseiten erstellen, Herdt Verlag, 2016)

→ Testen Sie folgendes Beispiel:

```
<?php
$tp = array("Montag"=>17.5, "Dienstag"=>19.2, "Mittwoch"=>21.8);
$tp["Donnerstag"] = 21.6;
$tp["Freitag"] = 17.5;
$tp["Samstag"] = 20.2;
$tp["Sonntag"] = 16.6;

// Ein bestimmtes Element
echo "<p>" . $tp["Montag"] . "</p>";

// Alle Keys und Values aus dem Feld
echo "<p>";
foreach($tp as $name=>$wert)
    echo "$name, $wert<br>";
echo "</p>";

// Nur alle Values aus dem Feld, zum Berechnen des Mittelwerts
$summe = 0;
foreach($tp as $wert)
    $summe = $summe + $wert;
$mittelwert = $summe / 7;
echo "<p>Mittelwert: $mittelwert</p>";
?>
```

Array mit foreach –
Schleife durchlaufen.
(Siehe nächste Seite.)

→ Ersetzen Sie in dem obigen Beispiel die Zahl 7 derart, dass die Berechnung auch für andere Anzahlen möglich ist.

6.3 Daten extrahieren

`foreach()` erwartet die Angabe des assoziativen oder indizierten Arrays, dessen Elemente durchlaufen werden sollen. In der *ersten Variante* werden mithilfe von `foreach` bei jedem Schleifendurchlauf der Variablen `$wert` nacheinander die Werte der Array-Elemente zugewiesen. Diese Variante wird häufig für indizierte Arrays eingesetzt, da hier die automatisch vergebenen Indexwerte oft ohne Bedeutung sind und nicht weiter verarbeitet oder ausgegeben werden sollen.

```
foreach($feld as $wert) {  
    Anweisungsblock;  
}  
foreach($feld as $index =>  
    $wert) {  
    Anweisungsblock;  
}
```

Mithilfe der *zweiten Variante* können Sie zusätzlich den Index jedes Array-Elements auslesen. Bei jedem Durchlauf wird der Schlüssel des Array-Eintrags der Variablen `$index` zugewiesen, der dazugehörige Array-Wert der Variablen `$wert`. Diese Variante wird häufig bei assoziativen Arrays verwendet, wenn der Schlüssel für die weitere Verarbeitung benötigt oder ausgewertet wird. Aber auch bei indizierten Arrays kann diese Variante eingesetzt werden.¹

6.4 Kurzschreibweisen für Arrays

```
$feld = [Wert1, Wert2, Wert3, ...];
```

```
$staedte = ["Frankfurt", "Berlin", "Bern"];
```

- Das Schlüsselwort `array` entfällt.
- Statt runder Klammern werden eckige Klammern verwendet.

```
$hauptstaedte = ["Schweiz" => "Bern",  
    "Frankreich" => "Paris",  
    "Deutschland" => "Berlin",  
    "Spanien" => "Madrid"];
```

¹ PHP 7.0 – Dynamische Webseiten erstellen, Herdt Verlag, 2016, S. 80.

6.5 Mehrdimensionale Arrays

Im folgenden Beispiel sehen Sie mehrere Angaben (Vorname, Nationalität und Alter), die jeweils einen Eintrag in einer Array-Variablen bilden. Damit ist dieses Array ein mehrdimensionales Array. Bei jeder der Angaben sind Mehrfachwerte möglich. Mehrdimensionale Arrays sind verschachtelte Arrays, man spricht hier von äußeren und inneren Arrays.

Mehrdimensionale Arrays können sowohl numerisch indiziert als auch assoziativ sein. Eine Mischform aus numerisch indizierten und assoziativen Arrays ist ebenfalls möglich.²

6.5.1 Mehrdimensional indiziertes Array

Index 1	Index 2	Wert
0	0	Oliver
	1	Spanisch
	2	37 Jahre
1	0	Maria
	1	Deutsch
	2	23 Jahre
2	0	Oliver
	1	Englisch
	2	46 Jahre

```
$person = array(
    array("Oliver",
        "Spanisch",
        "37 Jahre"
    ),
    array("Maria",
        "Deutsch",
        "23 Jahre"
    ),
    array("Oliver",
        "Englisch",
        "46 Jahre"
    )
);
```

```
// (manuelle) Erweiterung des Arrays
durch einen neuen Eintrag
$person[3][0] = "Johanna";
$person[3][1] = "Schwedisch";
$person[3][2] = "19 Jahre";

// Zugriff auf eine bestimmte Angabe
in einem
// mehrdimensionalen numerischen
Array
// Ausgabe der Angaben von Maria
echo $person[1][0] . " ist " . $person[1][2] . " alt und spricht " . $person[1][1] .
".<hr>";
```

Erweitern eines mehrdimensionalen Arrays:

```
$person[3][0] = "Johanna";
$person[3][1] = "schwedisch";
$person[3][2] = "19 Jahre";
```

Oder

```
$person[] = array("Johanna", "schwedisch",
    "19 Jahre");
```

ÜBUNG

→ Geben Sie die Inhalte des Arrays als kleine Tabelle aus.

Name	Alter	Sprache
Oliver	Spanisch	37 Jahre
Maria	Deutsch	23 Jahre
Oliver	Englisch	46 Jahre
Johanna	Schwedisch	19 Jahre

² PHP 7.0 – Dynamische Webseiten erstellen, Herdt Verlag, 2016, S. 81.

6.5.2 Mehrdimensionales assoziatives Array

Um ein mehrdimensionales Array anzulegen, wird jedem Schlüssel ein weiteres Array mit Schlüssel-Wert-Paaren übergeben. Dabei wird jedem Schlüssel auf erster Ebene ein weiteres Array zugewiesen. In diesen verschachtelten Arrays werden dann jeweils mehrere Schlüssel mit Werten angegeben.³

```
$land = array(
    "Spanien" => array("Hauptstadt" => "Madrid",
        "Sprache" => "Spanisch",
        "Waehrung" => "Euro",
        "Flaeche" => "504645 qkm"
    ),
    "England" => array("Hauptstadt" => "London",
        "Sprache" => "Englisch",
        "Waehrung" => "Pfund Sterling",
        "Flaeche" => "130395 qkm"
    ),
    "Portugal" => array("Hauptstadt" => "Lissabon",
        "Sprache" => "Portugiesisch",
        "Waehrung" => "Euro",
        "Flaeche" => "92345 qkm"
    )
);
```

Zugriff auf die Inhalte

```
echo "<p><strong>Angaben zu Portugal:</strong><br>";
echo "Hauptstadt: " . $land["Portugal"]["Hauptstadt"] . "<br>";
echo "Sprache: " . $land["Portugal"]["Sprache"] . "<br>";
echo "Währung: " . $land["Portugal"]["Waehrung"] . "<br>";
echo "Fläche: " . $land["Portugal"]["Flaeche"] . "</p>";
```

Erweitern der Inhalte

```
// (manuelle) Erweiterung des Array durch einen neuen Eintrag
$land["Ungarn"]["Hauptstadt"] = "Budapest";
$land["Ungarn"]["Sprache"] = "Ungarisch";
$land["Ungarn"]["Waehrung"] = "Forint";
$land["Ungarn"]["Flaeche"] = "93036 qkm";
// oder
$land["Ungarn"] = array("Hauptstadt" => "Budapest",
    "Sprache" => "Ungarisch",
    "Waehrung" => "Forint",
    "Flaeche" => "93036 qkm"
);
```

ÜBUNG

- Geben Sie die Inhalte des Arrays als Tabelle aus.
- Recherchieren Sie die Funktion ksort() und fügen Sie einen button hinzu und sortieren Sie.

Spanien	Madrid	Spanisch	Euro	504645 qkm
England	London	Englisch	Pfund Sterling	130395 qkm
Portugal	Lissabon	Portugiesisch	Euro	92345 qkm
Ungarn	Budapest	Ungarisch	Forint	93036 qkm

³ PHP 7.0 – Dynamische Webseiten erstellen, Herdt Verlag, 2016, S. 83.

6.5.3 Mischform

```

$staedte = array(
    "Japan" => array("Tokio", "Japanisch", "Yen"),
    "Niederlande" => array("Amsterdam", "Niederländisch", "Euro"),
    "Polen" => array("Warschau", "Polnisch", "Złoty"),
    "Indien" => array("Neu Delhi", "Indisch", "Rupie"),
    "Island" => array("Reykjavik", "Isländisch", "Krone"),
    "Italien" => array("Rom", "Italienisch", "Euro"),
    "Frankreich" => array("Paris", "Französisch", "Euro"),
    "Spanien" => array("Madrid", "Spanisch", "Euro"),
    "England" => array("London", "Englisch", "Pfund Sterling")
);

// Auslesen des gesamten Arrays mit foreach
foreach ($staedte as $key => $ausgabe) {
    list($hauptstadt, $sprache, $waehrung) = $ausgabe;
    echo "<tr>";
    echo "<td>" . $key . "</td>";
    echo "<td>" . $hauptstadt . "</td>";
    echo "<td>" . $sprache . "</td>";
    echo "<td>" . $waehrung . "</td>";
    echo "</tr>";
}
?>^


```

`list()` dient dazu, die einzelnen Werte des inneren Arrays aufzunehmen und sie den Variablen zuzuweisen, die als Parameter in `list()` angegeben sind.

- In den runden Klammern der Funktion `list()` werden die Variablennamen angegeben, in denen die einzelnen Array-Werte gespeichert werden sollen. Diese können frei gewählt werden.
- Über dem Zuweisungsoperator `=` wird der Funktion `list()` die Array-Variable zugewiesen, im Beispiel `$wert`. Dies wiederholt sich beim `foreach()` bei jedem einzelnen Schleifendurchlauf.
- Nun werden die Werte der in `list()` angegebenen Variablen mit den einzelnen Einträgen des Arrays versehen. Die erste Variable erhält den Wert des ersten Eintrags aus dem Array `$wert`, die zweite Variable den des zweiten Eintrags aus dem Array usw.
- `list()` benötigt numerisch indizierte Arrays, die mit dem Schlüssel 0 beginnen und die nicht mit abweichenden Schlüsseln versehen worden sind.

6.6 Übungen zu Arrays

Übung 1: Mit eindimensionalen Arrays arbeiten

Level		Zeit	ca. 10 min
Übungsinhalte	<ul style="list-style-type: none"> ✓ Arrays ✓ Array-Elemente hinzufügen ✓ Array-Elemente löschen ✓ Nutzen der <code>foreach()</code>-Schleife ✓ HTML-Ausgabe 		
Übungsdatei	--		
Ergebnisdatei	<i>kennzeichen.php</i>		

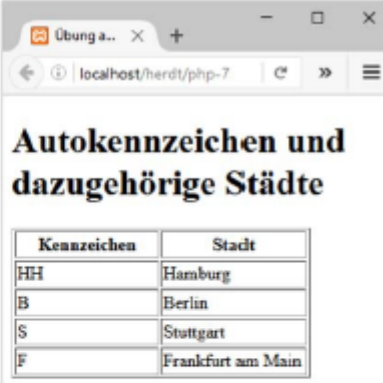
- Erstellen Sie eine neue PHP-Datei unter dem Namen *kennzeichen.php*. Definieren Sie eine assoziative Array-Variable, die Sie mit der folgenden Liste von Autokennzeichen und den dazugehörigen Städten füllen:

HH Hamburg
B Berlin
S Stuttgart

- Ergänzen Sie die Liste nach der Erstellung mit den Elementen:

F Frankfurt
HB Bremen


- Löschen Sie mithilfe des Befehls `unset()` das Element *Bremen* aus dem Array und definieren Sie den Eintrag *Frankfurt* neu – er soll nun *Frankfurt am Main* heißen.
- Listen Sie mithilfe von `foreach` alle Elemente der Tabelle auf und vergeben Sie zusätzlich Überschriften für die Tabelle.



Autokennzeichen und dazugehörige Städte	
Kennzeichen	Stadt
HH	Hamburg
B	Berlin
S	Stuttgart
F	Frankfurt am Main

Beispiel-Ergebnisdatei „*kennzeichen.php*“

Übung 2: Mit mehrdimensionalen Arrays arbeiten

Level		Zeit	ca. 10 min
Übungsinhalte	<ul style="list-style-type: none"> ✓ Mehrdimensionale Arrays ✓ Array-Elemente hinzufügen ✓ Array-Elemente löschen ✓ Nutzen der <code>foreach()</code>-Schleife ✓ HTML-Ausgabe 		
Übungsdatei	--		
Ergebnisdatei	uebung_mehrdimensional.php		

- Erstellen Sie eine neue Datei unter dem Namen `uebung_mehrdimensional.php`. Erstellen Sie ein mehrdimensionales indiziertes Array mit folgenden Inhalten und geben Sie die Daten anschließend in Tabellenform (mit Überschrift) auf dem Bildschirm aus:

Beginn	Disziplin	Ort	Bemerkung
09:30 Uhr	Diskuswurf	Nebenplatz	Jugendmeisterschaften
10:00 Uhr	5-km-Lauf	Stadion - Laufbahn	Offener Lauf
11:00 Uhr	Halbmarathon	Waldgebiet	Teilnahme ab 18 Jahren
12:00 Uhr	Stabhochsprung	Stadion - Stabhochsprunganlage	Nur Frauen



Beginn	Disziplin	Ort	Bemerkung
09.30 Uhr	Diskuswurf	Nebenplatz	Jugendmeisterschaften
10.00 Uhr	5 km-Lauf	Stadion - Laufbahn	Offener Lauf
11.00 Uhr	Halbmarathon	Waldgebiet	Teilnahme ab 18 Jahren
12.00 Uhr	Stabhochsprung	Stadion - Stabhochsprunganlage	Nur Frauen

Beispiel-Ergebnisdatei „uebung_mehrdimensional.php“