

Declaration

explicit	<code><?php</code>	<code><?php</code>
location	<pre>\$printer[0] = " Laserjet 2001"; \$printer[1] = " LMP 172 4fa st";</pre>	<pre><?php \$printer = array('R&B PRINTERS' => " Laserjet 2002", 'Ja - ckson Printers' => " J20 01L - ase rje t",); print_r(\$ printer); //r esult array (0 => " Laserjet 2001", 1 => " LMP 172 4fa st",)</pre>

implicit

location	<pre>\$printer[] = " Laserjet 2001"; \$printer[] = " LMP 172 4fa st";</pre>	<pre>\$printer[] = " Laserjet 2001"; \$printer[] = " LMP 172 4fa st";</pre>
-----------------	---	---

Associative Arrays

```
<?php
    $printer = array( 'R&B
                    PRINTERS' => " Laserjet 2002",
                    'Ja -
                    ckson Printers' => " J20 01L -
                    ase rje t",
                    );
    print_r($ printer);

    //r esult
    array (
        0 => " Laserjet 2001",
        1 => " LMP 172 4fa st",
    )
```

with **associative arrays**, you can reference the items in an array by name rather than by number.

Multidimensional Arrays

```
<?php
    $products = array(
        'pens' => array(
            'ball' =>
                "Ball Point",
            'hi lite' =>
                " Hig hli ght ers ",
            'ma rker' =>
                " Mar ker s"),
        'misc' => array(
            'tape' => " -
                Sticky Tape",
            'glue' => " -
                Adh esi ves ",
            'clips' => " -
                Pap erc lip s"
        )
    );
```

Assignment using the array Keyword

The for..each as loop

```
<?php
    $paper = array( " Cop ier ",
    " Ink jet ", " Las er", " Pho -
    to");
    $j = 0;
    for eac h($ paper as $item)
    {
        echo "$j: $item< br> ";
        ++$j;
    }
    ?>
```

Functions

Functions

is_array()

```
echo (is_ar ray ($p rod ucts)) ? "
y";
```

count()

```
echo count( $pr odu cts);
```

sort()

```
sort($products);
sort($products, SORT_NUMERIC);
sort($products, SORT_S TRING);
```

shuffle()

```
shuffl e($ cards);
```

explode()

```
<?php
$temp = explode(' ', "This is a se
    print_r($ temp);
    ?>
```

extract()

```
extrac t($ _GET);
```

compact()

```
<?php
$planet = " Gal lif rey ";
$system = " Gri dlo ck";
$const ell ation = " Kas ter -
bor ous ";
$contact = compact 'planet',
'system', 'const ell ati on';
```

use compact, the inverse of extract, to create an array from variables and their values.

```
`<?php
$p1 = array("China", "Russia",
"Austria", "Turkey");
$p2 = array
(
'china' => "Beijing",
'Russia' => "Moscow" ,
'Austria' => "Vienna",
'Turkey' => "Ankara"
);
```



By **shirM**
cheatography.com/shirM/

Published 7th May, 2023.
Last updated 7th May, 2023.
Page 1 of 2.

Sponsored by **Readable.com**
Measure your website readability!
<https://readable.com>