

print-r()

Prints human-readable information about a variable.

what -name it is about
how (2, 1, 2, 2, 1)
and

var_dump()

Displays structured information about one or more expressions that includes its type and value.

-|-

is_array()

Returns ^{TRUE} if var is an array, FALSE otherwise.

Example:

```
<?php
```

```
$yes = array('this', 'is', 'an array');  
echo is_array($yes) ? 'Array' : 'not an Array';
```

```
echo "\n";
```

```
$no = 'this is a string';
```

```
echo is_array($no) ? 'Array' : 'not an Array';
```

```
?>
```

Output:

Array

not an Array

bool is_int()

Return true if var is an integer, false otherwise.

Example:

```
<?php
```

```
if (is_int(23))
```

```
{ echo "is integer\n"; }
```

```
else { echo "is not an integer\n"; }
```

```
var_dump(is_int(23));
```

```
var_dump(is_int("23"));
```

```
var_dump(is_int(23.5));
```

```
var_dump(is_int(true));
```

```
?>
```

Output:

is integer

bool(true)

bool(false)

bool(false)

bool(false)

extract()

Convert array name, value pair to variable, value pair

Example:

```
<?php
```

```
$var_array = array("color" => "blue",  
                  "size" => "medium",
```

```
extract($var_array; "shape" => "sphere");
```

```
echo "$color, $size, $shape\n";
```

```
$_POST = array("name" => "Jitendra",  
              "age" => "28");
```

```
extract($_POST);
```

```
echo "$name, $age";
```

```
?>
```

Output:

blue, medium, sphere

Jitendra, 28

implode()

to form

Join array elements with a string

Example:

```
<?php
```

```
$array = array('lastname', 'email', 'phone');
```

```
$comma-separated = implode(", ", $array);
```

```
echo $comma-separated;
```

```
?>
```

Output:

lastname,email,phone

explode()

array of

Return an/string, each of which is a substring of string.

Example:

```
<?php
```

```
$comma-separated = "lastname,email,phone";
```

```
$arr = explode($comma-separated);
```

```
$arr = explode(", ", $comma-separated);
```

```
echo "$arr[0] | $arr[1] | $arr[2]";
```

```
?>
```

Output:

lastname

email

phone


```
// Fetching object array
```

```
while($row = $result->fetch-object()) {  
    printf("%s (%s)\n", $row->Name, $row->Country);  
}
```

list()

Assign variable as if they were an array.

Example:

```
<?php
```

```
$info = array('coffee', 'brown', 'caffeine');
```

```
list($drink, $color, $power) = $info;
```

```
echo "$drink is $color and $power makes it special.\n";
```

```
list($drink, , $power) = $info;
```

```
echo "$drink has $power.\n";
```

```
list(, , $power) = $info;
```

```
echo "I need $power.\n";
```

```
?>
```

Output:

coffee is brown and caffeine makes it special.

coffee has caffeine

I need caffeine

Date/05-02-2012

```
$mysqli = new mysqli("localhost", "my_user",  
    "my_pass", "my_db");
```

```
// Check connection
```

```
if($mysqli->connect_error) {
```

```
    die('Connect Error (' . $mysqli->connect_errno . ')  
    ' . $mysqli->connect_error);
```

```
// select queries return a result set
```

```
if($result = $mysqli->query("SELECT Name FROM City LIMIT  
    printf("select returned %d rows.\n", $result->num_rows);
```

```
// free result set
```

```
$result->close();
```

```
} // close connection
```

```
$mysqli->close();
```

-5-