

The PHP logo, consisting of the lowercase letters 'php' in a white, italicized sans-serif font, set against a solid blue rectangular background.

php

The Pentalog logo, featuring the word 'Pentalog' in a blue sans-serif font, followed by a circular icon containing a stylized blue lightning bolt.

Pentalog

PHP Training



Episode 1

by Andrei Kojusko, 2018

About me

Andrei Kojusko

15+ years with PHP

akojusko@pentalog.com

Skype: p5_akojusko

About PHP

- PHP 1, 2 - basic scripts
- PHP 3, 4 - programming language
- PHP 5.0 - OOP-centered
- PHP 5.4 - full OOP
- **PHP 6 - unicode**
- PHP 7 - strict typing
- PHP 7.2 - now

- Frameworks: Symfony, Zend, etc.

Lesson contents

- Using PHP from CLI
- Variables and types
- Operators
- Control structures
- Functions

Using PHP from CLI

- IDE/text editor
- `php filename.php`
- `<?php`
- CLI config
- no request variables (`$_GET`, `$_POST`, `$_FILES`, etc.)
- `\ namespace`

Variables

```
$name = 'Andrei';
```

```
$greeting = "Hello $name !"; // => 'Hello Andrei !'
```

```
$year = 1999;
```

```
$price = 1.25;
```

```
$finalPrice = round($price * 1.2); // => 2
```

```
$date = new \DateTime(); //pointer type!
```

```
$connection = mysqli_connect("127.0.0.1", "my_user", "my_password", "my_db");
```

Reserved variables = superglobals

- `$_SERVER`
- `$_GET`
- `$_POST`
- `$_SESSION`
- `$_ENV`

and other

Types

- Boolean (true, false)
- Integer (1, 2, -5)
- Float (-1.34, 3.1415, 2.0)
- String ("wassup")
- Array ([1, 2, 'test' => 3, 'more test', new \DateTime()])
- Object (new \DateTime())
- Callable (function (\$name) { echo 'Hello' . \$name;})
- Null (null)
- Resource

- Type casting: (int) '33' === 33

Operators

`2 + 2 // => 4`

`2 - 2 // => 0`

`2 / 2 // => 1`

`2 * 2 // => 4`

`5 % 2 // => 1`

`'a' . 'b' // => 'ab'`

`1 == 1 // => true`

`1 == '1' // => true`

`1 === '1' // => false`

`2 > 1 // => true`

`1 < 2 // => true`

`2 >= 2 // => true`

`2 <= 2 // => true`

Control structures - if/else

```
if ($a > $b) {  
    echo "a is bigger than b";  
}  
elseif ($a == $b) {  
    echo "a is equal to b";  
}  
else {  
    echo "a is smaller than b";  
}
```

Order is important

Control structures - loops

```
$arr = [1, 2, 3, 4];
```

```
foreach ($arr as &$amp;value) {
```

```
    $value = $value * 2;
```

```
}
```

```
// [2, 4, 6, 8]
```

```
foreach ($arr as $key => $value) {
```

```
    $arr[$key] = $value * 2;
```

```
}
```

```
// [4, 8, 12, 16]
```

```
for ($i = 1; $i <= 3; $i++) {
```

```
    echo $i;
```

```
}
```

```
// 1 2 3
```

```
$arr = [1, 2, 3, 4];
```

```
$len = \count($arr);
```

```
for ($i = 0; $i <= $len; $i++) {
```

```
    echo $arr[$i];
```

```
}
```

```
// 1 2 3 4
```

Control structures - other

```
switch ($i) {  
    case 0:  
        echo "i equals 0";  
        break;  
  
    case 1:  
        echo "i equals 1";  
        break;  
  
    case 2:  
        echo "i equals 2";  
        break;  
  
    default:  
        echo "i has other value";  
}
```

```
$res = $a > 0 ? 'positive' : 'negative'
```

```
$res = $a ?? $b ?? $c
```

```
while (list($key, $value) = each($arr)) {  
    if (!($key % 2)) { // skip even members  
        continue;  
    }  
  
    do_something_odd($value);  
}
```

Language basics - links

<http://php.net/manual/en/language.types.php>

<http://php.net/manual/en/language.variables.php>

<http://php.net/manual/en/reserved.variables.php>

<http://php.net/manual/en/language.operators.php>

<http://php.net/manual/en/language.control-structures.php>

Functions - arguments

```
function makecoffee($type = "cappuccino")  
{  
    return 'Making a cup of ' . $type . "\n";  
}
```

```
echo makecoffee();  
echo makecoffee("espresso");
```

```
// Making a cup of cappuccino.  
// Making a cup of espresso.
```

```
function add_some_extra(&$string)  
{  
    $string .= 'and something extra.';  
}
```

```
$str = 'This is a string, ';
```

```
add_some_extra($str);
```

```
echo $str;
```

```
// This is a string, and something extra.
```

Functions - important

- count
- implode
- explode
- strlen
- substr
- strtolower
- strtoupper
- ucfirst

Functions - links

<http://php.net/manual/en/functions.arguments.php>

<http://php.net/manual/en/functions.returning-values.php>

<http://php.net/manual/en/functions.anonymous.php>

<http://php.net/manual/en/book.strings.php>

<http://php.net/manual/en/book.array.php>

Scope

```
$a = 1; /* global scope */
```

```
function test($a)
{
    echo $a; /* reference to local scope variable */
    $b = 5;
}
```

```
test(2);
```

```
echo $a;
echo $b;
```

```
// 2
```

```
// 1
```

```
// notice - $b is undefined
```

Script arguments

```
> php script.php Andrei Kojusko
```

```
<?php
```

```
var_dump($_SERVER['argv']); // $argv
```

```
/*
```

```
array(3) {
```

```
  [0] => string(10) "script.php"
```

```
  [1] => string(6) "Andrei"
```

```
  [2] => string(7) "Kojusko"
```

```
}
```

```
*/
```

```
$_SERVER['argc'] = count($_SERVER['argv'])
```

Homework

1. `>php hello.php m Smith`
Hello Mr Smith!
2. `>php hello.php f Smith`
Hello Ms Smith!
3. `>php hello.php M smith F jackson f smith`
Hello Mr Smith and Ms Jackson and Ms Smith!
4. `>php hello.php OR >php hello.php f`
Missing arguments.
5. `>php hello.php z Smith`
Unknown title.