

Раздаточный материал по предмету “Язык программирования PHP”. Лекция 3.

Пример нестрогой типизации переменных в PHP

```
<?php
$var = 1;
echo gettype($var). "<br />";
$var = "string";
echo gettype($var). "<br />";
$var = 0x88;
echo gettype($var). "<br />";
$var = true;
echo gettype($var). "<br />";
?>
```

```
<?php
var_dump((bool) "");           // bool(false)
var_dump((bool) 1);            // bool(true)
var_dump((bool) -2);           // bool(true)
var_dump((bool) "foo");        // bool(true)
var_dump((bool) 2.3e5);         // bool(true)
var_dump((bool) array(12));     // bool(true)
var_dump((bool) array());       // bool(false)
var_dump((bool) "false");       // bool(true)
?>
```

ТИП BOOLEAN

При преобразовании в логический тип, следующие значения рассматриваются как **FALSE**:

- Сам [булев FALSE](#)
- [целое 0](#) (ноль)
- [число с плавающей точкой 0.0](#) (ноль)
- пустая [строка](#) и [строка "0"](#)
- пустой [массив](#)
- [объект](#) без атрибутов (только в PHP 4)
- специальный тип [NULL](#) (включая неустановленные переменные)

Все остальные значения рассматриваются как **true** (включая любой [ресурс](#)).

Внимание

-1 считается **true**, как и любое ненулевое (отрицательное или положительное) число!

Особенности экранирования спецсимволов и вывода переменных при использовании одинарных кавычек.

```
<?php
echo 'это простая строка';

echo 'Также вы можете вставлять в строки
символ новой строки таким образом,
поскольку это нормально';

// Выведет: Однажды Арнольд сказал: "I'll be back"
echo 'Однажды Арнольд сказал: "I\'ll be back"';

// Выведет: Вы удалили C:\*.*?
echo 'Вы удалили C:\\*.*?';

// Выведет: Вы удалили C:\*.*?
echo 'Вы удалили C:\/*.*?';

// Выведет: Это не вставит: \n новую строку
echo 'Это не вставит: \n новую строку';

// Выведет: Переменные $expand также $either не подставляются
echo 'Переменные $expand также $either не подставляются';
?>
```

Несколько примеров строк

```
<?php
// Получение первого символа строки
$str = 'Это тест.';
$first = $str{0};

// Получение третьего символа строки
$third = $str{2};

// Получение последнего символа строки
$str = 'Это все еще тест.';
$last = $str{strlen($str)-1};

// Изменение последнего символа строки
$str = 'Посмотри на море';
$str{strlen($str)-1} = 'я';

?>
```

Порядок выполнения операторов

Ассоциативность	Оператор
неассоциативна	new
правая	[
неассоциативна	++ --
неассоциативна	! ~ - (int) (float) (string) (array) (object) @
левая	* / %
левая	+ - .
левая	<< >> Побитовые сдвиги влево/вправо
неассоциативна	< <= > >=
неассоциативна	== != === !==
левая	& Побитовое 'и'
левая	^ Исключающее или
левая	Побитовое 'или'
левая	&&
левая	
левая	? :
правая	= += -= *= /= %= &= = ^= <<= >>=
левая	and
левая	xor
левая	or
левая	,

Основные управляющие структуры

- **if**

```
if ($a==1)
    $b=2;
```

- **else**

```
if ($a==1)
    $b=2;
else
    $b=3;
```

- **elseif**

```
if ($a==1)
    $b=2;
elseif ($a==2)
    $b=3;
elseif ($a==3)
    $b=4;
else
    $b=5;
```

- **while**

```
$i=0;
while($i<3){
    $i++;
    echo $i."<br />";
}
```

- **do-while**

```
$i = 5;
do {
    echo $i."<br />";
    $i--;
} while ($i > 0);
```

- **for**

```
for($i=0;$i<=5;$i++){
    echo $i."<br />";
}
```

- **foreach**

Организация итерации по индексированным и ассоциированным массивам

```
foreach (array_expression as $value)
    statement
foreach (array_expression as $key => $value)
    statement
```

```
$arr = array("one", "two", "three");
foreach ($arr as $value) {
    echo "Значение: $value<br />\n";
}
```

```
$a = array(1, 2, 3, 17);

foreach ($a as $v) {
    echo "Текущее значение \$a: $v.\n";
}
```

```
/* foreach пример 2:
Выведет:
Текущее значение $a: 1.
Текущее значение $a: 2.
Текущее значение $a: 3.
Текущее значение $a: 17.
```

```

*/

$a = array(1, 2, 3, 17);

$i = 0; /* Только для иллюстрации */

foreach ($a as $v) {
    echo "\$a[$i] => $v.\n";
    $i++;
}
/*
    Выведет
    $a[0] => 1. $a[1] => 2. $a[2] => 3. $a[3] => 17.
*/

```

- **break**

```

while($i<3){
    if($i==2)
        break;
    $i++;
    echo $i."<br />";
}

```

- **continue**

```

while($i<3){
    if($i==2)
        continue;
    $i++;
    echo $i."<br />";
}

```

- **switch**

Более наглядное представление обработки разных вариантов действия

```

if ($i == 0) {
    echo "i равно 0";
} elseif ($i == 1) {
    echo "i равно 1";
} elseif ($i == 2) {
    echo "i равно 2";
}

```

```

switch ($i) {
case 0:
    echo "i равно 0";
    break;
case 1:
    echo "i равно 1";
    break;
case 2:
    echo "i равно 2";
    break;
default:
    echo "i равно чему-то другому";
}

```