

5- Условные конструкции (if-else). Логические операторы

Условные конструкции позволяют проверить некое выражение и в зависимости от его результата выполнить необходимый код. В уроке мы познакомимся с конструкцией «if - else» и изучим все тонкости данной конструкции.

Логические операции используются для проверки данных. Их часто применяют в условных операторах, которые мы будем изучать в дальнейшем. Каждая логическая операция возвращает либо 0 (**false**), либо 1 (**true**).

В логических операциях есть следующие операнды:

`$a == $b` - равенство (если a равно b);

`$a != $b` - неравенство (если a **не** равно b);

`$a > $b` - больше (если a больше b);

`$a < $b` - меньше (если a меньше b);

`$a >= $b` - больше или равно (если a больше или равно b);

`$a <= $b` - меньше или равно (если a меньше или равно b);

`!$a` - ложь (если a равно false);

`$a` - истинна (если a равно true).

Если понадобится проверить несколько условий в одной логической операции, то мы можем воспользоваться логическим "и" или же логическим "или":

// Если и первое, и второе условие верные,

// то результат будет true

`$a != $b && $a > $b`

// Если или первое, или второе условие верные,

// то результат будет true

`$a < $b || $a == $b`

Оператор эквивалентности

В **PHP** существует специальный оператор эквивалентности, который позволяет проверить на соответствие значений и на соответствие типов данных.

Для проверки на эквивалентность используйте тройной символ равенства - `===`. В результате он проверит значение переменной, а также сравнит тип данных. Если оба значения будут совпадать, то результат будет **true**, иначе – **false**.

Условные конструкции

Условные операторы очень важны, ведь они позволяют выполнять разный код в зависимости от исхода условия. В языке PHP существует три конструкции позволяющих осуществлять проверку. Сейчас рассмотрим две из них.

Конструкция `if - else`

За счёт `if else` можно проверить одно или несколько условий и в случае их успешной проверки будет выполнен один, иначе другой.

Рассмотрим пример:

```
$a = 2;
```

```
$b = 10;
```

```
if ($a == $b) { // Если $a будет равным $b, тогда будет выполнен код
```

```
    // Здесь код который будет выполнен
```

```
    // Если все одна строка кода, то фигурные скобки {}
```

```
    // можно не ставить
```

```
    echo "$a == $b";
```

```
} else if ($a <= $b) { // Если $a будет меньшим или равным $b
```

```
    // Если предыдущее условие не будет выполнено,
```

```
    // а здесь условие окажется верным,
```

```
    // то будет выполнен этот код
```

```
    echo "$a <= $b";
```

```
} else {
```

```
    // Этот код сработает, если другие условия не будут выполнены
```

```
    echo "$a > $b";
```

```
}
```

Вы можете прописать структуру лишь с одним условием `if`, а можете дописать в неё сколько-угодно вложенных условий `else if`.

Внутри каждого из условий можно прописывать другие конструкции if else, что будут проверять новые выражения.

Если необходимо проверить несколько условий в одном операторе, то можно воспользоваться логическим «и» или же логическим «или»:

```
if ($a != $b && $a > $b) {  
    // Код будет выполнен, если и первое, и второе условие  
    // окажутся верными  
}
```

```
if ($a < $b || $a == $b) {  
    // Код будет выполнен, если или первое, или второе условие  
    // окажется верным  
}
```

Тернарная операция

Это сокращенная форма if else. Записывается следующим образом:

```
$x = 90;  
$a = 8;  
$res = $x < $a ? ($x + $a) : ($x - $a);  
echo $res;
```

Мы проверяем, если "x" будет меньше за "a", то мы в переменную res устанавливаем их сумму, иначе - их разницу.

Исходной код Выполнить задание

```
<?php  
$a = 5;  
$str = "Hello";
```

```
$isWeatherGood = false;
```

```
if($str == "Hello" && !$isWeatherGood)
```

```
    echo 'First';
```

```
else if($a == 5) {
```

```
    $res = "Second";
```

```
    echo $res;
```

```
    if($str == "Hello") {
```

```
        echo "<br>Yes";
```

```
    }
```

```
}
```

```
else if($a > 50)
```

```
    echo '$a > 50';
```

```
else if($a <= 45)
```

```
    echo '$a <= 45';
```

```
else
```

```
    echo 'Else';
```

```
?>
```

Задание1

1.

У вас есть переменная:

```
$a = 23;
```

Создайте условие if else, которое:

будет выводить число "10", если переменная равна 5;

будет выводить число "15", если переменная равна 10;

будет выводить число "20", если переменная равна 30;

если все проверки не подойдут, то необходимо вывести предложение: «Что-то не то!».

2. Логическая операция

У вас есть две переменные:

$x = 4.2$;

$y = 5.6$;

Создайте несколько логических операций, которые будут проверять:

является ли x равным числу 15;

является ли y большим или равным за 6;

равные ли между собой переменные x и y ;

Все результаты проверок выведите на экран.

3. Создайте переменную «a» со значением 34.

Создайте проверку, которая будет выполняться при условии, что число будет меньше 50, но не равно 37. При этом не меньше или равно 32.

Допишите в условие проверку, что результат будет также со значением «true» в случае когда переменная «a» будет со значением 0 или 15.

4. У вас есть две переменные:

$x = 7$;

$y = "7"$;

Проверьте их на равенство лишь значений, а также на равенство и значений, и типов данных.

Выполните проверку и посмотрите на отличия.

5. Из двух чисел с разной четностью вывести на экран нечетное число. a, b - данные числа

6. Создайте две переменных со значениями 5 и -3.

Создайте переменную, в которую установите значение в зависимости от результата тернарного оператора:

установку деления переменных в случае если делитель не равен числу 0;

установку число 1 в случае если делитель равен числу 0.