

Тернарный оператор ¶

Ну всё-таки предлагаю начать с типа данных `null` [КЛИК](#)

Специальное значение `null` представляет собой переменную без значения. `null` - это единственно возможное значение типа `null`.

Переменная считается `null`, если:

- ей была присвоена константа `null`.
- ей ещё не было присвоено никакого значения.
- она была удалена с помощью [unset\(\)](#).

Давайте создадим переменную без значения и выведем её на экран

```
$price;
```

```
var_dump($price);
```

Также мы можем объявить эту переменную и присвоить ей явно `null`

```
$price = null;
```

```
var_dump($price);
```

Тернарный оператор ¶

Давайте возьмём условие из предыдущей нашей темы

```
$light = "green";

if ($light == "green"){
    echo "go";
}else {
    echo "stop";
}
```

Мы можем записать эту запись покороче

```
echo $light == "green" ? "go" : "stop" ;
```

Мы сначала проверяем истина у нас или ложь

если у нас условия true то срабатывает инструкция после вопросительного знака
если условия у нас False то срабатывает инструкция после двоеточия

Также мы можем сделать 3 условия

```
$code = 3;
echo $code == 0 ? 0 : ($code == 1 ? 1 : 2);
echo true ? 0 : (true ? 1 : 2);
```

php8 Без скобок ничего не будет работать

Также результат мы можем присвоить переменной

```
$code = 3;
$a = $code == 0 ? 0 : ($code == 1 ? 1 : 2);
echo $a;
```

Также мы можем использовать более короткую запись

```
<?php error_reporting(-1);
```

```
$price = 100;
```

```
echo $price ? $price : "цена не установлена";
```