**Условные операторы php**

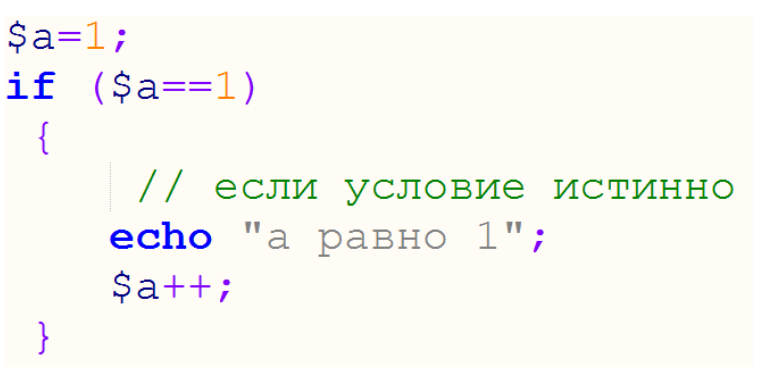
Условные операторы php представлены тремя основными конструкциями:

* оператор условия **if**,
* оператор переключения **switch**
* и **тернарный оператор**.

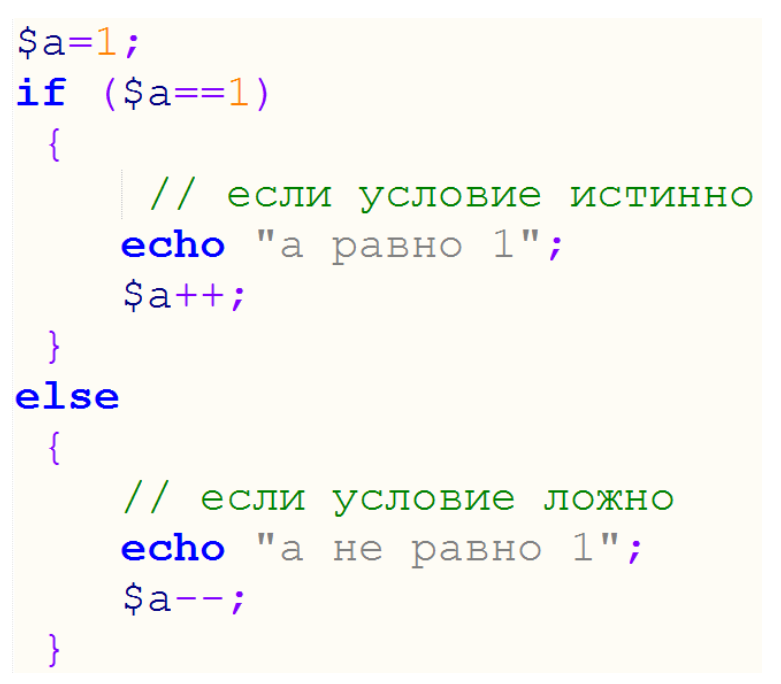
Рассмотрим подробнее каждый из них.

## PHP оператор if

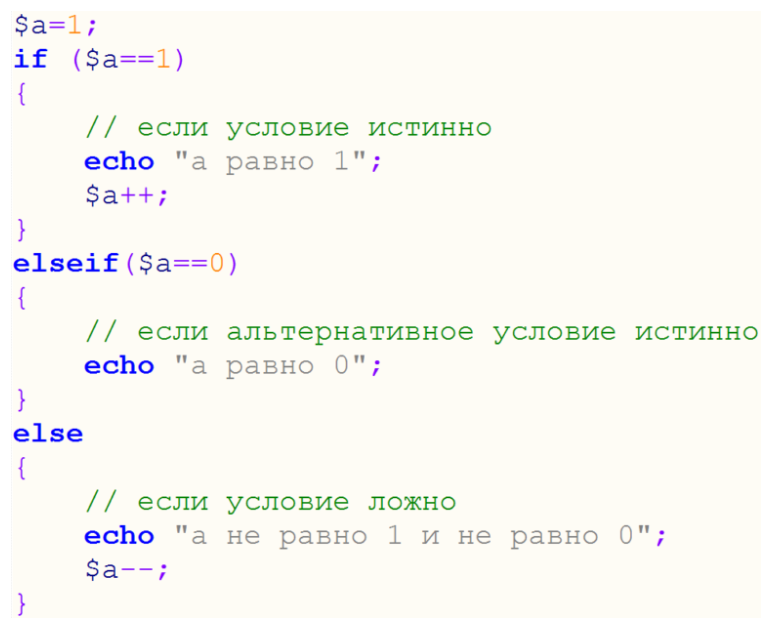
## Условный оператор if, укороченный вариант



**Синтаксис условного оператора If Else**



**Полный синтаксис условного оператора if elseif**

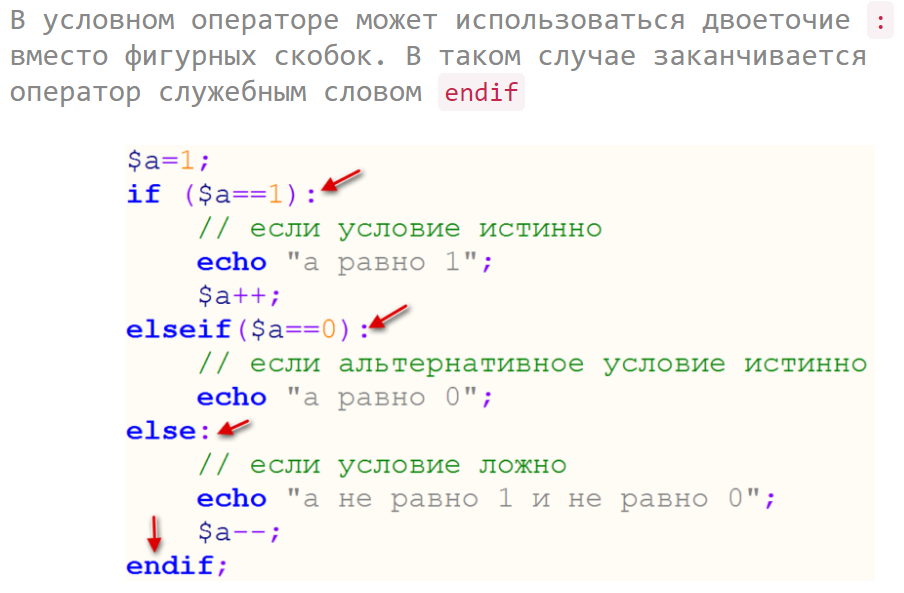


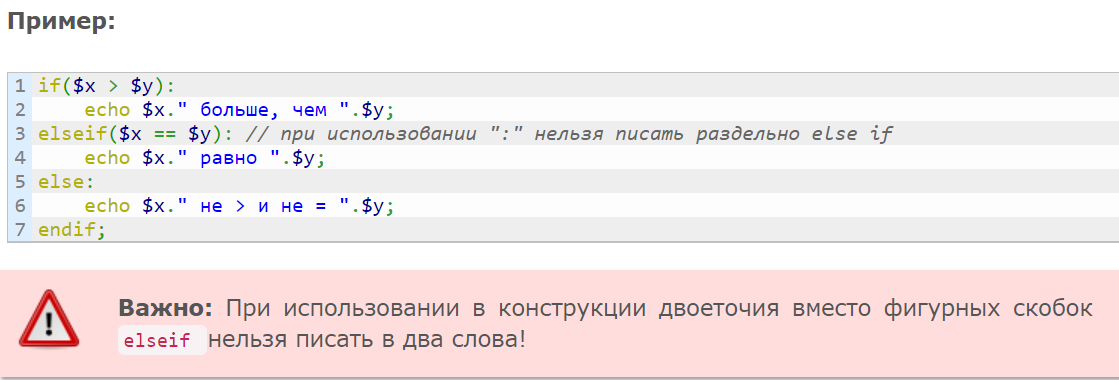
**Подытожим:**

Полный синтаксис:

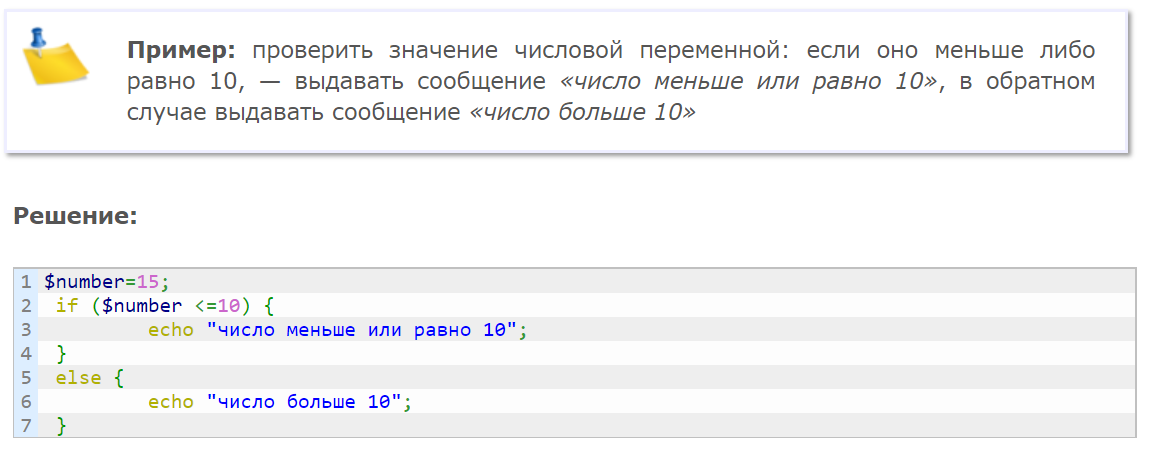


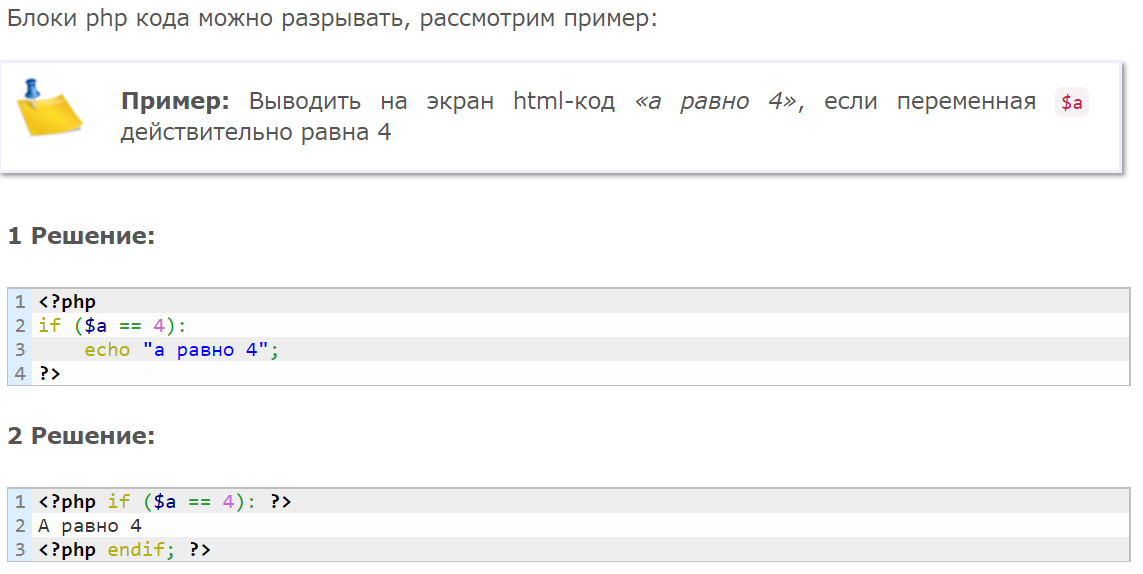
* Укороченный синтаксис может **не содержать части конструкции с else и не содержать дополнительного условия elseif**
* Вместо служебного слова elseif можно писать else if (отдельно)
* В одной конструкции if может быть несколько elseif. Будет выполнено первое встретившееся выражение elseif равное TRUE.
* В случае наличия альтернативного условия elseif конструкция else должна идти последней в синтаксисе.

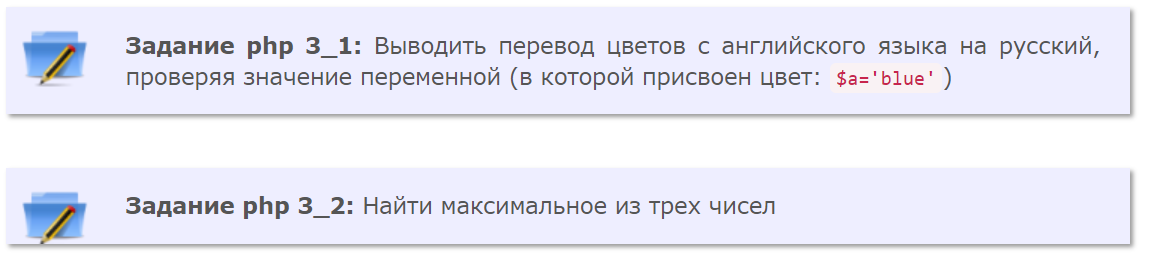








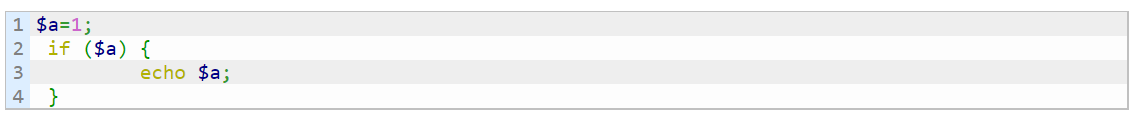




**ОПЕРАЦИИ СРАВНЕНИЯ И ПРАВИЛО ЛЖИ**

В конструкции if в скобках должно находиться логическое выражение или переменная, которые рассматриваются с точки зрения алгебры логики, возвращающая значения либо true либо false

Т.е. в качестве условия может выступать единственная переменная. Рассмотрим пример:

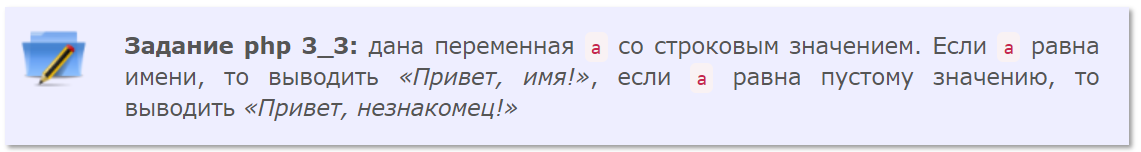


В примере транслятор языка php будет рассматривать переменную в скобках на правило лжи:

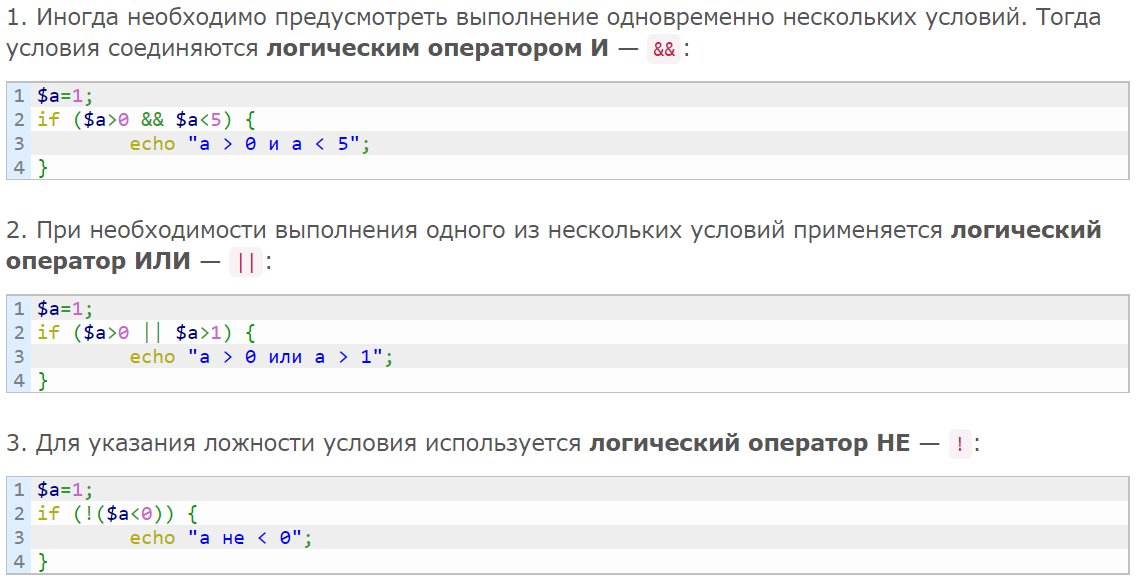
**Правило ЛЖИ или что считается ложным:**

* логическое **False**
* целый ноль (**0**)
* действительный ноль (**0.0**)
* **пустая строка** и строка **«0»**
* массив без элементов
* объект без переменных
* специальный тип **NULL**

Таким образом, в рассмотренном примере переменная $a равна единице, соответственно условие будет истинным и оператор echo $a; выведет на экран значение переменной.

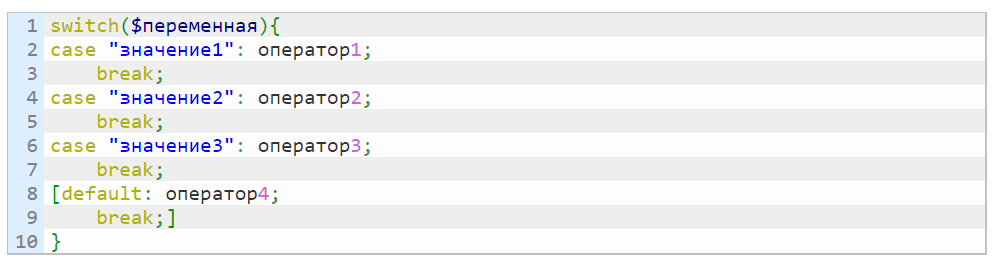


### **ЛОГИЧЕСКИЕ КОНСТРУКЦИИ**И ИЛИ **И**НЕ **В УСЛОВНОМ ОПЕРАТОРЕ**

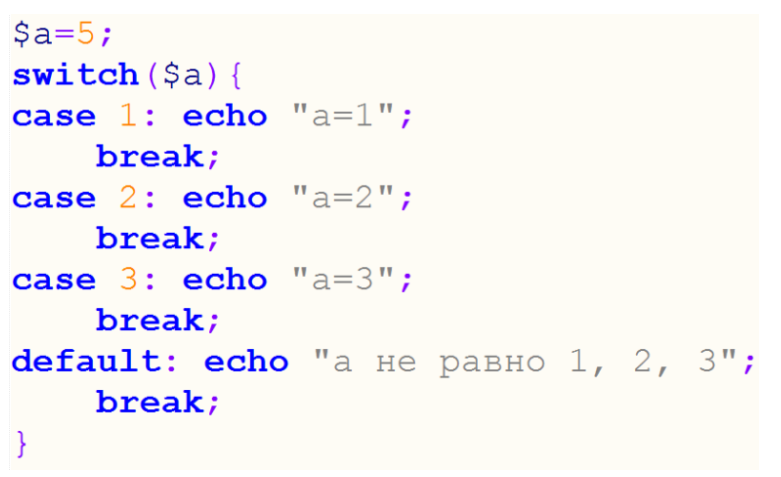


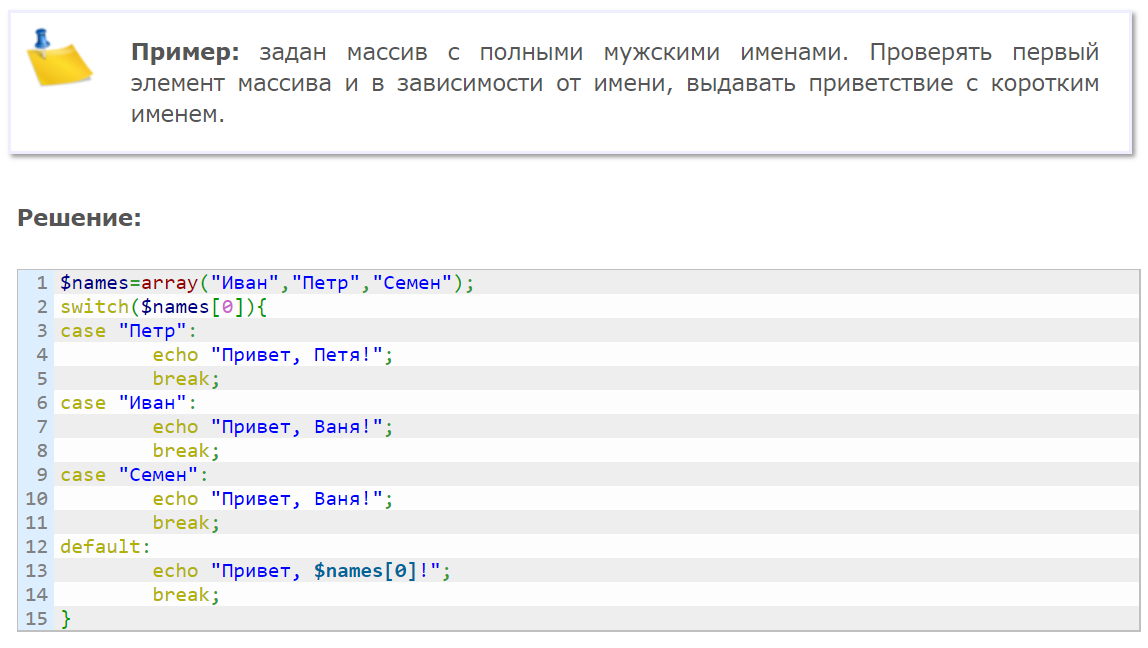
## Оператор Switch PHP

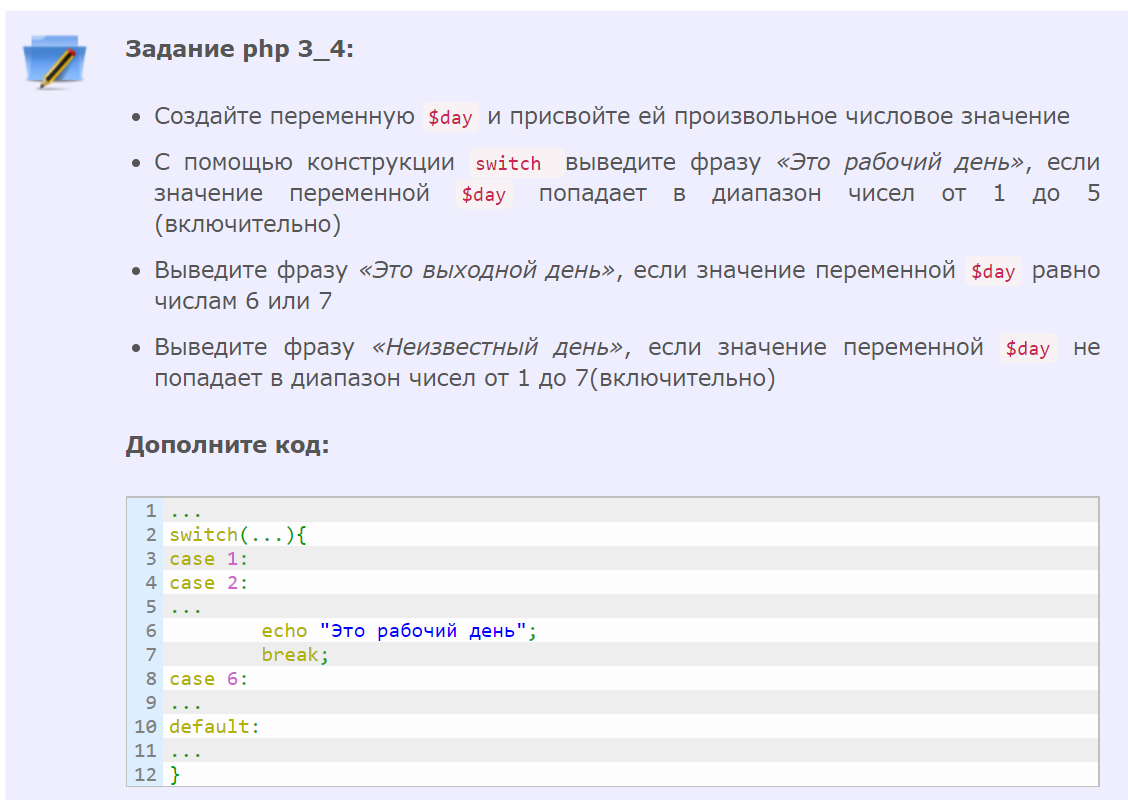
Оператор switch или «переключатель» заменяет несколько подряд идущих конструкций if. При этом он сравнивает одну переменную с множеством значений. Таким образом, это наиболее удобное средство **для организации мультиветвления**.



* Оператор может проверять как строковые значения (тогда они указываются в кавычках), так и числовые (без кавычек).
* Оператор break в конструкции обязателен. Он осуществляет выход из конструкции при истинном условии и выполнении соответствующего условию оператора. Без break будут выполняться операторы всех case независимо от их истинности.







## Тернарный оператор PHP

Тернарный оператор, т.е. с тремя операндами, имеет достаточно простой синтаксис, в котором слева от знака ? записывается условие, а справа — два оператора разделенные знаком :, слева от знака выполняется оператор при истинности условия, а справа от знака : выполняется оператор при ложном условии.



