CODE ACADEMY

ОБУЧЕНИЕ ПО ПРОГРАМИРАНЕ

Числови типове. Оператори. Приоритети

CODE ACADEMY

Type Juggling

Трябва да се отбележи за PHP, че той осигурява автоматично преобразуване на типовете данни. Така че, ако присвоите цяло число на променлива, типът на тази променлива автоматично ще бъде integer. След това, ако присвоите текстов низ на същата променлива, типът ще се промени на string. Това се нарича Type Juggling.

Тази особеност се променя след PHP 7, където е въведено т.н. type declaration за аргументи на функции, връщани стойности, и class properties.

CODE ACADEMY

PHP Integers

- 2, 256, -256, 10358, -179567 всички са integers (цели числа)
- Едно цяло число трябва да има поне една цифра
- Цяло число НЕ трябва да има десетична запетая
- Едно цяло число може да бъде положително или отрицателно
- Целите числа могат да бъдат зададени в три формата: десетичен, шестнадесетичен или осмичен

CODE ACADEMY

PHP Integers
(Примерен код)

Numeric Data Types

```php
<?php
$x = 5985;
var_dump(is_int($x)); // prints TRUE

$x = 59.85;
var_dump(is_int($x)); // prints FALSE
```

CODE ACADEMY

PHP Floats

float е число с десетична запетая или число в експоненциална форма (пр.: 2.0, 256.4, 10.358, 7.64E+5, 5.56E-5)

Типът данни float обикновено може да съхранява стойност до 1.7976931348623E+308 (в зависимост от платформата) и има максимална точност от 14 цифри.

CODE ACADEMY

PHP Floats
(примерен код)

В следващия пример **$x** е float. Функцията в PHP **var_dump()** ще принтира TRUE като резултат от проверката **is_float()**:

Numeric Data Types

```php
<?php
$x = 10.365;
var_dump(is_float($x)); // prints TRUE
?>
```

CODE ACADEMY

Оператори в PHP

Операторите се използват за извършване на операции с променливи и стойности.

PHP разделя операторите на следните групи:

- Аритметични оператори
- Оператори за присвояване
- Оператори за сравнение
- Оператори за увеличаване/намаляване
- Логически оператори
- Стрингови оператори
- Оператори за масиви
- Оператори за условно присвояване

CODE ACADEMY

Аритметични оператори

PHP аритметичните оператори се използват с числови стойности за извършване на общи аритметични операции, като събиране, изваждане, умножение и т.н.

Оператор	Действие	Пример	Резултат
+	Събиране	$x + $y	Сумата на $x и $y
-	Изваждане	$x - $y	Разликата м/у $x и $y
*	Умножение	$x * $y	Произведението на $x по $y
/	Делене	$x / $y	Частното на $x и $y
%	Модул	$x % $y	Остатък от $x, делено на $y
**	Степенуване	$x ** $y	Резултат от повдигане на $x на $y-та степен

CODE ACADEMY

Оператори за присвояване

PHP операторите за присвояване се използват с числови стойности за записване на стойност в променлива.

Основният оператор за присвояване в PHP е "=". Това означава, че на левият операнд се задава стойността на израза за присвояване отдясно.

Присвояване	Същото като	Описание
x = y	x = y	Левият операнд присвоява стойността на израза отдясно
x += y	x = x + y	Събиране
x -= y	x = x - y	Изваждане
x *= y	x = x * y	Умножение
x /= y	x = x / y	Делене
x %= y	x = x % y	Модул

CODE ACADEMY

Оператори за сравнение

PHP операторите за сравнение се използват за сравняване на две стойности (число или стринг):

Оператор	Име	Пример	Резултат
==	Равенство	$x == $y	Връща true, ако $x е равно на $y
===	Идентичност	$x === $y	Връща true, ако $x е равно на $y и те са от един и същи тип
!=	Не равно	$x != $y	Връща true, ако $x не е равно на $y
<>	Не равно	$x <> $y	Връща true, ако $x не е равно на $y
!==	Не идентично	$x !== $y	Връща true, ако $x не е равно на $y или не са от един и същи тип
>	По-голямо	$x > $y	Връща true, ако $x е по-голямо от $y
<	По-малко	$x < $y	Връща true, ако $x е по-малко от $y
>=	По-голямо или равно от	$x >= $y	Връща true, ако $x е по-голямо или равно на $y
<=	По-малко или равно от	$x <= $y	Връща true, ако $x е по-малко или равно на $y
<=>	Spaceship	$x <=> $y	Връща цяло число, по-малко, равно на или по-голямо от нула, в зависимост от това дали $x е по-малко, равно или по-голямо от $y. Въведено в PHP 7.

CODE ACADEMY

Оператори за нарастване / намаляване

Операторите за увеличаване на PHP се използват за увеличаване на стойността на променлива.

PHP операторите за намаляване се използват за намаляване на стойността на променлива.

Оператор	Име	Описание
++$x	Pre-increment	Увеличава $x с единица, след което връща $x
$x++	Post-increment	Връща $x, след което увеличава $x с единица
--$x	Pre-decrement	Намалява $x с единица, след което връща $x
$x--	Post-decrement	Връща $x, след което намалява $x с единица

CODE ACADEMY

Логически оператори

Логическите оператори в PHP се използват за комбиниране на условни изрази.

Оператор	Име	Пример	Резултат
and	And	$x and $y	Вярно, ако $x и $y са верни
or	Or	$x or $y	Вярно, ако $x или $y е вярно
xor	Xor	$x xor $y	Вярно, ако $x или $y е вярно, но не и двете
&&	And	$x && $y	Вярно, ако $x и $y са верни
||	Or	$x || $y	Вярно, ако $x или $y е вярно
!	Not	!$x	Вярно, ако $x не е вярно

CODE ACADEMY

Стринг оператори

PHP има два оператора, които са специално проектирани за работа със стрингове:

Оператор	Име	Пример	Резултат
.	Конкатениране	$txt1 . $txt2	Конкатенира $txt1 и $txt2
.=	Конкатениране с присвояване	$txt1 .= $txt2	Добавя $txt2 към $txt1

CODE ACADEMY

Оператори за масиви

PHP операторите за масиви се използват за сравняване на масиви. Не се опитвайте да ги запомните, сега само ги споменавам, а в темата за масиви ще разгледаме подробно.

Оператор	Име	Пример	Резултат
+	Union	$x + $y	Обединение на $x и $y
==	Equality	$x == $y	Връща true, ако $x и $y имат еднакви двойки ключ/стойност
===	Identity	$x === $y	Връща true, ако $x и $y имат едни и същи двойки ключ/стойност в същия ред и от едни и същи типове
!=	Inequality	$x != $y	Връща true, ако $x не е равно на $y
<>	Inequality	$x <> $y	Връща true, ако $x не е равно на $y
!==	Non-identity	$x !== $y	Връща true, ако $x не е идентичен на $y

CODE ACADEMY

Оператори за условно присвояване

PHP операторите за условно присвояване се използват за задаване на стойност в зависимост от условията:

Оператор	Име	Пример	Резултат
?:	Ternary	$x = *expr1* ? *expr2* : *expr3*	Връща стойността на $x. Стойността на $x е *expr2*, ако *expr1* = TRUE. Стойността на $x е *expr3*, ако *expr1* = FALSE
??	Null coalescing	$x = *expr1* ?? *expr2*	Връща стойността на $x. Стойността на $x е *expr1*, ако *expr1* съществува и не е NULL. Ако *expr1* не съществува или е NULL, стойността на $x е *expr2*. Въведено в PHP 7

CODE ACADEMY

Приоритети

Приоритетът на операторите и асоциативността са две важни концепции, които трябва да се разбират при работа с PHP оператори.

Приоритетът на операторите се отнася до реда, в който PHP оценява операторите в израз. Подобно е на реда на операциите в математиката. Например, ако един израз съдържа както оператори за събиране, така и за умножение, PHP ще оцени първо умножението, а след това събирането. Това е така, защото умножението има по-висок приоритет от събирането.

CODE ACADEMY

Асоциативност

Асоциативността, от друга страна, определя реда, в който PHP оценява оператори със същия приоритет в израз. Може да бъде както отляво надясно, така и отдясно наляво. Например, операторите за събиране и изваждане имат еднакъв приоритет и се оценяват отляво надясно. Това означава, че в израза 5 - 3 + 2 PHP първо ще оцени 5 - 3, след което ще добави 2 към резултата.

CODE ACADEMY

Обобщение

За да обобщим, приоритетът на операторите и асоциативността са правила, които определят реда, в който PHP оценява операторите в израз. Разбирането на тези концепции е важно за писането на правилен и ефективен PHP код.

CODE ACADEMY

Задачи за упражнение

1. Напишете PHP скрипт, който изчислява площта и обиколката на кръг. Декларирайте променлива за радиуса и й присвоете стойност. Изчислете площта и обиколката с помощта на подходящите формули и отпечатайте резултатите. Документацията на php.net е ваш приятел!
2. Декларирайте две променливи и им присвоете числови стойности. Използвайте аритметичните оператори, за да извършвате събиране, изваждане, умножение и деление на двете променливи. Разпечатайте резултатите.
3. Напишете програма на PHP, която преобразува температура от Целзий във Фаренхайт. Декларирайте променлива за температурата в Целзий и й присвоете стойност. Изчислете температурата във Фаренхайт по формулата (Целзий * 9/5) + 32 и отпечатайте резултата.

CODE ACADEMY

Задачи за упражнение

4. Декларирайте две променливи и им присвоете цели числа. Използвайте операторите за сравнение, за да определите дали една променлива е по-голяма, по-малка или равна на другата. Разпечатайте резултатите с var_dump или print_r
5. Напишете програма на PHP, която изчислява сумата на първите 10 четни числа. Използвайте цикъл и оператора на модула, за да определите дали дадено число е четно. Разпечатайте резултата.
6. Декларирайте три променливи и им присвоете числови стойности. Използвайте операторите за присвояване, за да промените стойностите на променливите. Разпечатайте резултатите след всяка задача с функцията print_r
7. Напишете скрипт на PHP, която изчислява корените на квадратно уравнение. Декларирайте променливи за коефициентите a, b и c и им присвоете числови стойности. Използвайте подходящата формула за изчисляване на корените и отпечатайте резултатите.