

Wprowadzenie do składni PHP

Wprowadzenie do składni PHP

Składnia PHP definiuje zasady, według których piszemy kod w tym języku. PHP łączy się z HTML, co pozwala na dynamiczne generowanie treści stron internetowych. Aby kod PHP działał poprawnie, musi być zapisany w odpowiednich blokach .

Przykład podstawowego bloku PHP:

```
<?php
    echo "Witaj w PHP!";
?>
```

Podstawowe reguły składni

Instrukcje i średniki

Każda instrukcja PHP powinna kończyć się średnikiem (;).

Przykład:

```
<?php
    $imie = "Jan";    // Poprawne
    echo $imie;       // Poprawne
?>
```

Błąd: brak średnika

```
<?php
    $imie = "Jan"    // Błąd: brak średnika
    echo $imie;
?>
```

Białe znaki

PHP ignoruje spacje, tabulatory i nowe linie poza cudzysłowami. Można formatować kod dla lepszej czytelności.

Przykład:

```
<?php
$liczba = 10;
$tekst = "PHP";
```

```
echo $liczba;  
echo $tekst;  
?>
```

Zmienne w PHP

Definicja zmiennych

Zmienne w PHP zaczynają się od znaku \$, po którym następuje nazwa zmiennej.

Przykład:

```
<?php  
    $nazwa = "PHP";  
    $liczba = 123;  
    $logiczna = true;  
  
    echo $nazwa;  
    echo $liczba;  
?>
```

Zasady nazewnictwa zmiennych

Muszą zaczynać się od litery lub znaku _.

Mogą zawierać litery, cyfry i podkreślenia.

Nie mogą zawierać spacji ani znaków specjalnych.

Poprawne:

```
<?php  
    $imie = "Anna";  
    $_wiek = 30;  
    $liczba123 = 50;  
?>
```

Błędne:

```
<?php  
    $1zmienna = 100;    // Błąd: zaczyna się od cyfry  
    $moj zmienna = 50; // Błąd: spacja w nazwie  
?>
```

Typy danych w PHP

PHP obsługuje różne typy danych:

Podstawowe typy danych

String (ciąg znaków):

```
$tekst = "To jest PHP";
```

Integer (liczba całkowita):

```
$liczba = 100;
```

Float (liczba zmiennoprzecinkowa):

```
$liczba_plywajaca = 3.14;
```

Boolean (wartość logiczna):

```
$prawda = true;  
$falsz = false;
```

Tablice

Tablice pozwalają przechowywać wiele wartości w jednej zmiennej.

Przykład tablicy:

```
<?php  
$owoce = array("Jabłko", "Banan", "Gruszka");  
echo $owoce[0]; // Wyświetli: Jabłko  
?>
```

Obiekty

PHP obsługuje programowanie obiektowe:

```
<?php  
class Samochod {  
    public $marka;  
    function ustaw_marke($nazwa) {  
        $this->marka = $nazwa;  
    }  
}  
  
$auto = new Samochod();  
$auto->ustaw_marke("Toyota");  
echo $auto->marka;  
?>
```

NULL

Typ NULL oznacza brak wartości:

```
<?php
    $zmienna = null;
?>
```

Operatory w PHP

PHP oferuje wiele operatorów do wykonywania działań:

Operatory arytmetyczne

- (+) - (dodawanie)
- (-) - (odejmowanie)
- (*) - (mnożenie)
- (/) - (dzielenie)
- (%) - (modulo – reszta z dzielenia)

Przykład:

```
<?php
    $a = 10;
    $b = 3;

    echo $a + $b;    // 13
    echo $a % $b;    // 1
?>
```

Operatory przypisania

- = (przypisanie)
- +=, -=, *=, /=, %= (przypisanie z operacją)

Przykład:

```
<?php
    $liczba = 5;
    $liczba += 10;    // równoważne $liczba = $liczba + 10;
    echo $liczba;    // 15
?>
```

Operatory porównania

- == (równość)
- === (równość wartości i typu)
- != (różne)
- !== (różne lub inny typ)
- <, >, <=, >=

Przykład:

```
<?php
    $a = 5;
    $b = "5";

    var_dump($a == $b);    // true
    var_dump($a === $b);  // false
?>
```

Operatory logiczne

&& (AND)

|| (OR)

! (NOT)

Przykład:

```
<?php
    $x = true;
    $y = false;

    var_dump($x && $y); // false
    var_dump($x || $y); // true
?>
```

Instrukcje warunkowe

PHP obsługuje klasyczne konstrukcje warunkowe:

If, else, elseif

```
<?php
$wiek = 20;

if ($wiek < 18) {
    echo "Niepełnoletni";
} elseif ($wiek >= 18 && $wiek < 65) {
    echo "Dorosły";
} else {
    echo "Senior";
}
?>
```

Operator trójargumentowy (ternary)

```
<?php
$punktacja = 85;
```

```
echo ($punktacja >= 50) ? "Zaliczony" : "Niezaliczony";  
?>
```

Pętle w PHP

Pętla for

```
<?php  
for ($i = 1; $i <= 5; $i++) {  
    echo "Iteracja: " . $i . "<br>";  
}  
?>
```

Pętla while

```
<?php  
$i = 1;  
while ($i <= 5) {  
    echo $i . "<br>";  
    $i++;  
}  
?>
```

Pętla foreach (dla tablic)

```
<?php  
$owoce = ["Jabłko", "Banan", "Gruszka"];  
  
foreach ($owoce as $owoc) {  
    echo $owoc . "<br>";  
}  
?>
```

Zadania do wykonania

1. Zadeklaruj zmienną \$nazwa i przypisz do niej wartość „PHP”. Następnie wyświetl ją za pomocą echo.
2. Stwórz zmienną \$liczba1 = 15 i \$liczba2 = 5. Wykonaj na nich wszystkie operacje arytmetyczne i wyświetl wyniki.
3. Napisz warunek, który sprawdzi, czy liczba 10 jest większa od 5, i wyświetli odpowiedni komunikat.
4. Stwórz tablicę kolory zawierającą wartości: „czerwony”, „zielony”, „niebieski”. Wyświetl jej zawartość pętlą foreach.
5. Użyj var_dump(), aby sprawdzić typ zmiennej true.
6. Napisz pętlę for, która wyświetli liczby od 1 do 10.
7. Użyj operatora trójargumentowego, aby sprawdzić, czy zmienna \$wiek jest większa od 18.

8. Stwórz skrypt, który wyświetli „Weekend”, jeśli dzień tygodnia to sobota lub niedziela, w przeciwnym razie „Dzień roboczy”.
9. Napisz pętlę while, która wyświetli liczby parzyste od 2 do 20.
10. Stwórz prostą funkcję w PHP, która przyjmuje dwa argumenty i zwraca ich sumę.