

Podstawy struktury PHP

Podstawy struktury PHP

PHP jest językiem skryptowym wykonywanym po stronie serwera, co oznacza, że kod PHP nie jest widoczny dla użytkownika w przeglądarce – zostaje przetworzony na serwerze, a do użytkownika wysyłany jest wynik w postaci HTML.

Każdy plik PHP musi mieć rozszerzenie .php i może zawierać zarówno kod PHP, jak i HTML.

Przykładowa struktura pliku PHP:

```
<!DOCTYPE html>
<html lang="pl">
<head>
    <meta charset="UTF-8">
    <title>Strona PHP</title>
</head>
<body>

<h1>Witaj na stronie</h1>

<?php
    echo "To jest kod PHP!";
?>

</body>
</html>
```

W powyższym kodzie:

HTML tworzy strukturę strony,

PHP jest osadzony wewnątrz znaczników i generuje dynamiczną treść.

Znaczniki PHP

PHP może być osadzane w kodzie HTML przy użyciu różnych znaczników:

Podstawowy znacznik PHP (zalecany):

```
<?php
    echo "To jest PHP!";
?>
```

Krótsza wersja (short tag, wymaga aktywacji w php.ini):

```
<?=  
    "To jest PHP!";  
?>
```

Krótsza forma jest skrótem dla .

Stary styl znaczników (niezalecany):

```
<?  
    echo "To jest PHP!";  
?>
```

Znaczniki skryptowe (deprecated, rzadko używane):

```
<script language="php">  
    echo "To jest PHP!";  
</script>
```

Rekomendowany jest pierwszy sposób, ponieważ jest obsługiwany przez wszystkie wersje PHP.

Podstawowe instrukcje PHP

Wyświetlanie danych

PHP pozwala na wyświetlanie treści w przeglądarce za pomocą:

echo – najczęściej stosowane:

```
echo "Witaj w PHP!";
```

print – działa podobnie jak echo, ale zwraca wartość:

```
print "To jest PHP!";
```

print_r() – stosowane do wyświetlania tablic:

```
$tablica = array("Jabłko", "Banan", "Wiśnia");  
print_r($tablica);
```

var_dump() – służy do debugowania, wyświetla szczegóły typu danych:

```
$x = 5;  
var_dump($x);
```

Komentarze w PHP

Komentarze pozwalają na dokumentowanie kodu:

Jednoliniowy komentarz:

```
// To jest komentarz jednoliniowy
```

Inna forma komentarza jednoliniowego:

```
# To także komentarz jednoliniowy
```

Wieloliniowy komentarz:

```
/*  
To jest komentarz  
wieloliniowy.  
*/
```

Podstawowe zasady składni PHP

Zmienne

PHP pozwala na tworzenie zmiennych, które przechowują dane. Każda zmienna w PHP zaczyna się od znaku \$:

```
$imie = "Jan";  
$wiek = 25;
```

Nazwy zmiennych:

Mogą zawierać litery, cyfry i podkreślenia.

Muszą zaczynać się od litery lub _.

Nie mogą zaczynać się od cyfry.

Nie mogą zawierać spacji i znaków specjalnych.

Typy danych

PHP obsługuje różne typy danych:

```
$tekst = "To jest tekst";    // String (ciąg znaków)  
$liczba = 25;                // Integer (liczba całkowita)  
$zmiennoprzecinkowa = 3.14; // Float (liczba zmiennoprzecinkowa)  
$logiczna = true;            // Boolean (wartość logiczna)  
$tablica = array("Jabłko", "Banan"); // Tablica  
$obiekt = new stdClass();    // Obiekt  
$brak = NULL;                // Wartość NULL
```

Stałe

Stałe w PHP to zmienne, których wartość nie może zostać zmieniona po zdefiniowaniu:

```
define("PI", 3.14159);  
echo PI;
```

Podsumowanie

PHP wykorzystuje znaczniki , aby osadzać kod wewnątrz HTML.

Instrukcje PHP muszą kończyć się średnikiem ;.

echo i print służą do wyświetlania danych.

Zmienne zaczynają się od \$, a stałe są definiowane przez define().

PHP obsługuje różne typy danych i operatory.

Pliki PHP muszą mieć rozszerzenie .php i działać na serwerze.

Zadania

1. Stwórz plik struktura.php i osadź w nim kod PHP oraz HTML.
2. Zadeklaruj zmienne imie, wiek i wyświetl je na stronie.
3. Stwórz tablicę owoce, dodaj do niej kilka owoców i wyświetl je w pętli foreach.
4. Stwórz skrypt, który wyświetli różne komunikaty w zależności od wartości zmiennej \$wiek.
5. Użyj print_r() do wyświetlenia tablicy zawierającej kilka nazw miast.
6. Wykorzystaj var_dump() do sprawdzenia typu zmiennej liczba, której wartość wynosi 20.
7. Stwórz stałą PI o wartości 3.14 i wyświetl jej wartość.
8. Użyj operatorów +=, -=, *=, /= do zmiany wartości zmiennej liczba i wyświetl jej wartość po każdej operacji.
9. Utwórz skrypt PHP, który dynamicznie wygeneruje stronę HTML zawierającą nagłówek i akapit.
10. Napisz kod, który sprawdzi, czy zmienna \$x istnieje, i jeśli tak, to wyświetli jej wartość.