

2. Zmienne

Zmienna to obszar w pamięci komputera, w którym programista może umieścić jakieś wartości a następnie je odczytywać. Każdej zmiennej przypisujemy własną, unikalną nazwę, która jednoznacznie ją identyfikuje. Język PHP wymaga, aby zaczynała się ona od znaku dolara, a następnie od litery (ew. podkreślenia). Dalsza część nazwy może już zawierać cyfry. Stosując wielkie litery trzeba uważać, ponieważ dla interpretera są one rozróżnialne od małych, co ma istotne znaczenie. `$zmienna` i `$Zmienna` to dwie różne zmienne. Przykłady poprawnych nazw zmiennych:

`$a`, `$b`, `$foo`, `$_50`, `$_Foo`, `$moja_zmienna`, `$mojaZmienna3`

Przykłady nieprawidłowych nazw:

`$5a`, `$'a'`, `$`

2.1 Sprawdzanie wartości i typu zmiennej

Wartość i typ zmiennej możemy sprawdzić za pomocą funkcji `var_dump()`. W wyniku wykonania skryptu:

2.1 Ćwiczenie 2.1

```
<?php
```

```
$dlugosc = 1258;
```

```
var_dump($dlugosc);
```

```
?>
```

ujrzymy komunikat:

```
int(1258)
```

Oznacza on, że zmienna o nazwie `$dlugosc` przechowuje liczbę całkowitą (napis `int` na powyższym wydruku) o wartości 1258.

Typ zmiennej wynika z wykonanego podstawienia. Jeśli do zmiennej wstawimy liczbę całkowitą 67, to typem zmiennej będzie `int`, jeśli wstawimy napis 'Anna', to typem zmiennej będzie `string`. Po wstawieniu wartości zmiennopozycyjnej 56.8 zmienna będzie typu `float`. Zaś po wstawieniu wartości logicznej `false` zmienna będzie miała typ `bool`.

2.1 Ćwiczenie 2.2

```
<?php  
$wiek = 67;  
var_dump($wiek);
```

```
$imie = 'Anna';  
var_dump($imie);
```

```
$waga = 56.8;  
var_dump($waga);
```

```
$mezatka = false;  
var_dump($mezatka);
```

```
?>  
Powyższy skrypt wydrukuje komunikat:
```

```
int(67)  
string(4) "Anna"  
float(56.8)  
bool(false)
```

2.2 Podstawowe typy zmiennych

W PHP występują cztery typy skalarne:

- bool (typ logiczny)
- int (typ całkowity)
- float (typ zmiennopozycyjny)
- string (napisy)

dwa typy złożone:

- array (tablice)
- object (obiekty)

oraz dwa typy specjalne:

- resource (zasób)
- NULL (typ pusty)

Jak już powiedzieliśmy, typ zmiennej wynika z kontekstu.

2.2.1 Typ logiczny bool

Po wstawieniu do zmiennej wartości logicznej, typem zmiennej będzie bool:

```
<?php  
  
$widoczny = false;  
var_dump($widoczny);
```

```
?>
```

Powyższy skrypt wydrukuje komunikat:

```
bool(false)
```

2.2.2 Typ całkowity int

Po wstawieniu do zmiennej wartości całkowitej, typem zmiennej będzie int:

```
<?php
```

```
$liczba = 619;  
var_dump($liczba);
```

```
?>
```

Wynikiem wykonania jest komunikat:

```
int(619)
```

2.2.3 Typ zmiennopozycyjny float

Jeśli do zmiennej wstawimy wartość zmiennopozycyjną:

```
<?php
```

```
$wspolczynnik = 3.1415;  
var_dump($wspolczynnik);
```

```
?>
```

to typem zmiennej będzie float:

```
float(3.1415)
```

2.2.4 Napisy. Typ string

Jeśli natomiast do zmiennej wstawimy napis:

```
<?php
```

```
$marka = 'Wartburg';  
var_dump($marka);
```

```
?>
```

to typem zmiennej będzie string:

```
string(8) "Wartburg"
```

2.3 Typ zmiennej może ulegać zmianom

Do tej samej zmiennej możemy wstawiać wartości różnych typów:

```
$zm = 123;  
$zm = true;  
$zm = 'tak';  
$zm = 3.15;
```

2.4 Funkcje do badania wartości i typu zmiennych

W PHP występują trzy funkcje umożliwiające zbadanie typu i wartości zmiennej. Są nimi **var_dump()**, **var_export()** oraz **print_r()**.

Funkcje te różnią się kilkoma szczegółami.

Po pierwsze, w odniesieniu do obiektów, funkcja **var_dump()** drukuje informacje wyłącznie o publicznych składowych (funkcje **var_export()** oraz **print_r()** uwzględniają wszystkie składowe: publiczne, chronione i prywatne).

Po drugie wydruk generowany przez funkcję **var_export()** jest poprawnym kodem PHP.

Wreszcie funkcja **print_r()** nie drukuje informacji o typie zmiennej, o ile typem jest string, integer lub float.