

Pozostałe operatory

Dodał Administrator
sobota, 13 marzec 2010 23:15

Indeks tablicy

Operator indeksowania tablicy składa się ze znaków nawiasu kwadratowego i umożliwia dostęp do poszczególnych komórek tablicy. Schematyczna postać (dla tablicy jednowymiarowej) wygląda następująco:

`nazwa_tablicy[indeks]` **Wywołanie funkcji**

Operator wywołania funkcji składa się z dwóch znaków nawiasu okrągłego i umożliwia wywołanie funkcji o zadanej nazwie. Jeżeli funkcja przyjmuje argumenty, należy je umieścić w nawiasie. Schematycznie wygląda to następująco:

`nazwa_funkcji(argumenty funkcji)` **Utworzenie obiektu**

Operator `new` służy do tworzenia obiektów. Jego schematyczne wywołanie ma postać:

`new konstruktor_obiektu();`

przykładowo:

`new Date();` **Usunięcie elementu**

Operator `delete` usuwa element tablicy o zadanym indeksie bądź składową obiektu. Przykładowo: jeśli utworzona zostanie tablica `tab`:

```
var tab = new Array("ind1", "ind2", "ind3");
```

to wykonanie instrukcji:

```
delete tab[1];
```

spowoduje usunięcie elementu tablicy o indeksie 1, numeracja pozostałych komórek pozostanie jednak bez zmian. Po wykonaniu tej operacji tablica `tab` będzie zawierała dwa elementy o indeksach 0 i 2. Operator `delete` dostępny jest w językach JavaScript od wersji 1.3 i JScript od wersji 3.0. **Sprawdzenie typu (operator `typeof`)**

Operator `typeof` pozwala na ustalenie typu zmiennej. Wynikiem działania operatora `typeof` na zmiennej może być jedna z następujących wartości:

-

`number` — dla wartości liczbowych,

-

`string` — dla łańcuchów znaków,

-

`object` — dla obiektów,

-

`function` — dla funkcji,

-

Pozostałe operatory

Dodał Administrator
sobota, 13 marzec 2010 23:15

boolean — dla wartości typu boolowskiego,

-

undefined — jeśli typ nie może być ustalony.

Przykład:

```
var x = 100;
```

```
document.write(typeof x)
```

Operator typeof dostępny jest w językach JavaScript od wersji 1.1 i JScript od wersji 1.0. Uwaga! Zapis typeof null daje w wyniku ciąg object. **Operator warunkowy**

Operator warunkowy ma postać:

```
warunek ? wartość1 : wartość2
```

Oznacza ona: jeżeli warunek jest prawdziwy, podstaw za wartość wyrażenia wartość1, w przeciwnym razie podstaw za wartość wyrażenia wartość2. Operator ten można traktować jak skróconą postać instrukcji warunkowej if. Zostanie on bliżej omówiony w dalszej części kursu. **Rozdzielanie wyrażeń**

Operator , (przecinek) pozwala na oddzielenie od siebie wyrażeń, które mają być przetwarzane od strony lewej do prawej. Najczęściej wykorzystywany jest przy deklaracjach wielu zmiennych w jednej linii programu, np.:

```
var liczba1 = 1, liczba2, liczba3 = 3;
```

oraz w pętli typu for, gdzie umożliwia umieszczenie kilku instrukcji w wyrażeniach początkowych, warunkowych i modyfikujących, np.:

```
for (i = 0, j = 100; i < 100, j > 0; i++, j--){/*instrukcje wnętrza pętli*/}
```