

ZAJĘCIA SPECJALISTYCZNE

JavaScript



TEMAT 1-3: Więcej o operatorach.

Autor dokumentu: Wojciech Galiński

wtorek, 5 lutego 2013 r.

312[01]/T,SP/MENIS/2004.06.14

ŹRÓDŁA WIEDZY: <http://pl.wikipedia.org/>, <http://webmaster.helion.pl/index.php/kurs-javascript>,
<http://krook.org/jsdom/>, <http://www.dynamicdrive.com/>, <http://www.w3schools.com>.

Zagadnienia obowiązkowe

1. Operatory – wyróżniamy:

➔ PODSTAWOWE OPERATORY – wyróżniamy:

- ✓ arytmetyczne – były omówione w poprzednim temacie;
- ✓ relacyjne – równe „==”, różne „!=”, identyczne „===” (równe i tego samego typu), nie identyczne „!==”, większe „>”, mniejsze „<”, większe lub równe „>=”, mniejsze lub równe „<=”);

PRZYKŁAD:

```
document.write( '0==false to ' + (0==false) +  
                ' ale 0===false to ' + (0===false) );
```

- ✓ logiczne – iloczyn logiczny „&&” (część wspólna wyrażeń), alternatywa logiczna „||” (suma wyrażeń), negacja logiczna „!” (negacja wyrażenia);

PRZYKŁAD: `document.write(!1 || 5!=3 && 10==5+5);`

- ✓ bitowe – iloczyn bitowy „&”, suma bitowa „|”, negacja bitowa „~”, alternatywa wyłączająca bitów „^”, przesunięcie bitów w lewo „<<”, przesunięcie bitów w prawo „>>”, przesunięcie bitów w prawo z wypełnieniem zerami „>>>”;

PRZYKŁAD:

```
document.write( (~1) + ' ' + (2&3) + ' ' + (2|3) + ' ' + (2^3) + ' ' +  
                (3<<1) + ' ' + (-3>>1) + ' ' + (-3>>>1) );
```

- ✓ przypisania – dostępne operatory: „=”, „+=”, „*=”, „/=”, „%=”, „<<=”, „>>=”, „>>>=”, „&=”, „|=”, „^=”. Wyrażenie typu: **x [operator] y** <=> **x = x [operator] y**, np. `x+=3`; `<=> x=x+3`;

➔ POZOSTAŁE OPERATORY – wyróżniamy:

- ✓ indeks tablicy – operator: **[]** (zostanie opisany w temacie o tablicach);
- ✓ wywołanie funkcji – operator: **()** (zostanie opisany w temacie o funkcjach);
- ✓ utworzenie obiektu – operator: **new** (zostanie opisany w temacie o obiektach);
- ✓ usunięcie obiektu – operator: **delete** (zostanie opisany w temacie o obiektach);
- ✓ operator warunkowy – operator: **?:** (zostanie opisane w temacie o warunkach);
- ✓ rozdzielanie wyrażeń – operator: **,** (służy do rozdzielania instrukcji tam, gdzie dozwolona jest tylko jedna instrukcja, m. in. przy deklaracjach wielu zmiennych lub w pętli „for”, np. `var x=1, y=2;`);
- ✓ sprawdzenie typu obiektu – operator: **typeof** nazwa_obiektu; (typ zmiennej ustalany jest na podstawie ostatnio przypisanej wartości, a operator ten zwraca aktualny typ zmiennej, czyli jedną z następujących wartości: number, string, object, function, boolean, undefined).

2. **Priorytety operatorów** – to kolejność wykonywania złożonych działań (1 to najwyższy priorytet). Priorytety operatorów przedstawione są w poniższej tabeli.

Priorytet	Operatory	Symbole operatorów
1	indeks tablicy, wywołanie funkcji	<code>[]</code> , <code>()</code>
2	inkrementacja i dekrementacja, ustalenie znaku, negacja bitowa i logiczna, utworzenie obiektu, ustalenie typu zmiennej, usunięcie składowej	<code>++</code> , <code>--</code> , <code>+</code> , <code>-</code> , <code>~</code> , <code>!</code> , <code>new</code> , <code>typeof</code> , <code>delete</code>
3	mnożenie, dzielenie, reszta z dzielenia	<code>*</code> , <code>/</code> , <code>%</code>
4	dodawanie, odejmowanie	<code>+</code> , <code>-</code>
5	przesunięcie bitowe, w lewo, w prawo, w prawo z wypełnieniem zerami	<code><<</code> , <code>>></code> , <code>>>></code>
6	mniejsze, większe, mniejsze lub równe, większe lub równe, porównanie typów	<code><</code> , <code>></code> , <code><=</code> , <code>>=</code>
7	równe, różne	<code>==</code> , <code>!=</code>
8	iloczyn bitowy	<code>&</code>
9	bitowa różnica symetryczna	<code>^</code>
10	suma bitowa	<code> </code>
11	iloczyn logiczny	<code>&&</code>
12	suma logiczna	<code> </code>
13	warunkowy	<code>?:</code>
14	operatory przypisania	<code>=</code> , <code>+=</code> , <code>-=</code> , <code>*=</code> , <code>/=</code> , <code>%=</code> , <code>&=</code> , <code>^=</code> , <code> =</code> , <code><<=</code> , <code>>>=</code> , <code>>>>=</code>
15	rozdzielanie wyrażeń	<code>,</code>

3. **Odczytywanie wartości z kontrolki „input text”** – oto przykład (więcej na temat dostępu do elementów dokumentu HTML, będziemy mówić podczas omawiania JavaScript DOM):

```
<script type="text/javascript">
  function info()
  {
    dane = document.getElementById('liczba');
    alert('Dane wejściowe: ' + dane.value);
    wynik = document.getElementById('wynik');
    wynik.innerHTML = 'Wynik: ' + (dane.value * dane.value);
  }
</script>
<input type="text" id="liczba" value="5" /> <span id="wynik"></span>
<input type="button" value="Naciśnij mnie" onclick="info()" />
```

Zadania

- Wczytaj 2 wartości za pomocą funkcji „**prompt()**”. Porównaj te wartości wykorzystując operatory relacji. Wyświetl te porównania oraz wartości tych porównań, np. „5==5 to true, 3==8 to false”.
- Zapytaj użytkownika, „czy świeci słońce” i „czy jest mróz”. Wyświetl informację o tym, czy:
 - ➔ obydwa zdarzenia miały miejsce;
 - ➔ co najmniej 1 z tych zdarzeń miało miejsce;
 - ➔ żadne z tych zdarzeń nie miało miejsca.
- Wczytaj liczbę całkowitą za pomocą funkcji „**prompt()**”. Wykorzystaj operatory bitowe w celu zmiany znaku oraz do pomnożenia tej liczby przez 8, np. 5 → -5, 5 → 40.
- Za pomocą jednej instrukcji utwórz 3 zmienne przechowujące różne rodzaje wartości i wyświetl typy tych zmiennych.
- Sprawdź i wyjaśnij wyniki wyrażeń: 2+2*2, (2+2)*2, (3*4)/(20-3*5).
- Za pomocą 2 kontrolki „input text” wczytaj 2 liczby i oblicz dla nich wyniki podstawowych operacji arytmetycznych. Oto przykładowy wynik zapisany w znacznikach „div”:
 5 + 2 = 7, 5 - 2 = 3, 5 * 2 = 10, 5 / 2 = 2.5, 5 % 2 = 1.