

ZAJĘCIA SPECJALISTYCZNE

JavaScript



TEMAT 1-5: Pętle (instrukcje iteracyjne).

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. **Instrukcja iteracyjna (pętla)** – to instrukcja, które umożliwia wielokrotne wykonywanie się pojedynczego zestawu instrukcji. Pomaga ona w realizowaniu powtarzających się zadań np. 10 razy wyświetlić coś na ekranie.
2. **Iteracja (powtórzenie)** – to jednokrotne wykonanie zestawu instrukcji wewnątrz instrukcji iteracyjnej (pętli).
3. **Pętla „while”** – oto jej ogólna postać:

```
while (warunek)
    instrukcja;
```

```
while (warunek)
{
    // instrukcje
}
```

Jej konstrukcja jest bardzo podobna do instrukcji warunkowej „if”. Instrukcja lub instrukcje wykonywane są, dopóki warunek jest spełniony. Warunek jest sprawdzany przed każdą iteracją, więc instrukcje wewnątrz tej pętli mogą nie wykonać się ani razu (tak jak w instrukcji „if”).

PRZYKŁADY:

```
var i=0; while (i++<100) document.write(i);
var j=0; while (++j>0) document.write(i);
```

4. **Pętla „do ... while”** – oto jej ogólna postać:

```
do
    instrukcja;
while (warunek);
```

```
do
{
    // instrukcje
}
while (warunek);
```

Instrukcja lub instrukcje wykonywane są także, dopóki warunek jest spełniony, ale warunek jest sprawdzany po każdej iteracji, więc instrukcje wewnątrz tej pętli muszą się wykonać co najmniej jeden raz.

PRZYKŁADY:

```
var i=0; do document.write(i) while (i++<100);
var j=0; do document.write(i) while (++j>0);
```

5. **Pętla „for”** – to rozbudowana wersja pętli „while”. Oto jej ogólna postać:

```
for (wyrazenie_i; warunek; wyrazenie_m)
{
    // instrukcje
}
```

Oto sposób działania tej pętli:

- ➔ najpierw wykonywane jest wyrażenie inicjalizujące zmienne iteracyjne („wyrażenie_i”) – zmienne te zliczają kolejne iteracje, a zwane są także iteratorami (zwykle występuje tam 1 zmienna);
- ➔ następnie wykonywane jest wyrażenie warunkowe („warunek”);
- ➔ gdy warunek jest spełniony, wykonane są: instrukcje wewnątrz pętli oraz wyrażenie modyfikujące iterator („wyrażenie_m”), a w przeciwnym razie następuje wyjście z pętli „for” do kolejnej instrukcji;
- ➔ kolejna iteracja rozpoczyna się od sprawdzenia warunku (wyrażenie inicjalizujące wykonywane jest tylko raz – na samym początku pętli).

PRZYKŁADY:

```
for (i=0; i<10; ++i) document.write(i + ' ');  
for (i=1; i<=10; ++i) document.write(i + ' ');  
for (i=10; i>=1; --i) document.write(i + ' ');
```

6. **Pętla „for ... in”** – służy do przeglądania wszystkich indeksów tablicy lub wszystkich właściwości obiektu. Będzie omówiona w dalszej części.
7. **Zagnieżdżanie pętli** – to umieszczanie jednej pętli wewnątrz innej (mogą być to 2 lub więcej dowolnych pętli). Zagnieżdżanie może spowodować długie wykonywanie się skryptu, np.

```
for (i=0; i<1000; ++i)  
    for (j=0; j<1000; ++j) document.write(i + '-' + j + ' ');
```

spowoduje aż milion (!) iteracji (to tak jakby odwołać do wszystkich elementów arkusza kalkulacyjnego o rozmiarach 1000x1000 komórek).

8. **Przerywanie bieżącej iteracji** – służy do tego instrukcja „continue”. Po napotkaniu takiej instrukcji następuje natychmiastowe rozpoczęcie wykonywania nowej iteracji pętli.

PRZYKŁAD:

```
for (i=0; i<10; ++i) if (i%3) continue; else document.write(i + ' ');
```

9. **Przerywanie pętli** – służy do tego instrukcja „break”. Po napotkaniu takiej instrukcji następuje natychmiastowe przeniesienie wykonywania programu do pierwszej instrukcji poza pętlą (ale pamiętać trzeba, że w pętli zagnieżdżonej powoduje ona wyjście tylko o 1 zagnieżdżenie do góry). PRZYKŁAD:

```
for (i=0; i<10; ++i) if (i==5) break; else document.write(i + ' ');
```

Zadania

1. Wyświetl wszystkie 3-cyfrowe liczby podzielne przez 31.
2. Wczytaj z klawiatury przedział [a, b]. Następnie wczytaj z klawiatury liczbę z tego przedziału. Do zabezpieczenia się przed nieprawidłowymi wpisami użyj odpowiedniej pętli. Wyświetl wszystkie liczby z podanego przedziału (jeśli a>b to malejąco, a w przeciwnym wypadku – rosnąco), rozdzielając je przecinkiem. Pogrub wszystkie liczby parzyste, podkreśl – podzielne przez 7, a wybraną liczbę pokoloruj na niebiesko, np. „[1, 20]: 1, **2**, 3, **4**, 5, **6**, 7, 8, 9, **10**, **11**, **12**, 13, **14**, 15, **16**, 17, **18**, 19, **20**”.
3. Wyświetl w tabeli HTML tabliczkę mnożenia używając dowolnej z poznanych pętli. Pierwszy rząd i pierwsza kolumna mają być nagłówkami tabeli.
4. Wykonaj tabliczkę mnożenia od 1 do 20 w pionie i od 1 do 10 w poziomie – użyj widoku kolumnowego, np.

1*1 = 1	1*2 = 2	1*3 = 3	...
2*1 = 2	2*2 = 4	2*3 = 6	...
...

Dostosuj czcionkę tak, żeby na Twoim monitorze widać było wszystkie 10 kolumn.

5. Wyświetl w tabeli HTML kalendarz na bieżący miesiąc – weź pod uwagę dni tygodnia (początek tygodnia w poniedziałek).