

ZAJĘCIA SPECJALISTYCZNE

JavaScript



TEMAT 1-7: Funkcje użytkownika.

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. **Funkcja** – to wydzielony blok kodu, działający niezależnie od reszty skryptu, przeznaczony do wykonywania konkretnego zadania.
2. **Budowa funkcji** – oto jej ogólne postaci (najprostsza i najbardziej rozbudowana):

```
function nazwa_funkcji()  
{  
    // instrukcje  
}  
  
function nazwa_funkcji(argument_1, ..., argument_N)  
{  
    // instrukcje  
    return wartosc;  
}
```

Oto znaczenie poszczególnych składowych:

- ➔ „NAZWA_FUNKCJI” – to nazwa poprzez którą wywołujemy (uruchamiamy) funkcję (musi spełniać zasady nazwy języka JavaScript);
 - ➔ „ARGUMENTY” – to lista danych wejściowych dla funkcji (wartości, która funkcja będzie przetwarzać). Lista ta rozdzielana jest przecinkami;
 - ➔ // INSTRUKCJE – w miejsce tego komentarza wstawiamy treść funkcji;
 - ➔ RETURN WARTOSC; – ten wiersz umożliwia zwrócenie np. wyniku na zewnątrz funkcji.
3. **Rodzaje argumentów funkcji** – do funkcji przekazywane być mogą tylko kopie argumentów.
 4. **Zwracanie wartości przez funkcję** – nie jest obowiązkowe (słowo kluczowe „return” może nie występować). Typ wartości zwracany przez funkcję zależy od wartości, która stoi po słowie kluczowym „return”.

PRZYKŁADY:

<pre>function nazwa_programu() { document.write('Nazwa programu'); }</pre>	<pre>function suma(wartosc1, wartosc2) { document.write(wartosc1 + wartosc2); }</pre>
<pre>function nazwa_programu() { return 'Nazwa programu'; }</pre>	<pre>function suma(wartosc1, wartosc2) { return wartosc1 + wartosc2; }</pre>

5. **Zmienne globalne** – dotychczas używaliśmy zwykle zmiennych globalnych (bez słowa kluczowego „var”). Takie zmienne widoczne są w całym skrypcie – nawet wewnątrz funkcji.
6. **Zmienne lokalne** – definiujemy je wewnątrz funkcji poprzedzając słowem kluczowym „var”. Takie zmienne dostępne są tylko wewnątrz funkcji, w której są zdefiniowane.
7. **Przesłanianie zmiennych** – jeśli wewnątrz jakiejś funkcji zdefiniujemy zmienną lokalną, a istnieje zmienna globalna o tej samej nazwie, to zmienna lokalna przysłoni widzialność zmiennej globalnej wewnątrz funkcji, w której została zdefiniowana.

Oznacza to, że poprzez nazwę zmiennej będziemy mieli dostęp tylko do zmiennej lokalnej.

```
function funkcja()
{
    var k = 10;
    document.write(k);
}
k = 5;
document.write(k);
funkcja();
document.write(k);
```

Zadania

1. Zaprojektuj funkcję wyświetlającą wszystkie litery alfabetu łacińskiego. Dodaj parametr informujący, czy mają być to duże, czy małe litery. Dodaj kolejny parametr kontrolujący, czy wstawiać polskie litery.
2. Zaprojektuj funkcję wyświetlającą wśród liczb od 1 do 1E6 takie potęgi, w których $a^b = b^a$.
3. Zaprojektuj funkcję wyświetlającą liczby pierwsze nie większe od 10000.
4. Wykonaj funkcję generującą kod HTML poniższej tabeli:

Nazwisko	Imię	Stawka za godzinę	Liczba godzin	Kwota brutto	Podatek (19%)	Do wypłaty
Nowak	Jan	50,0 zł	23,800	1190,00 zł	190,00 zł	1000,00 zł
Kowalska	Ewa	26,75 zł	40,000	1070,00 zł	170,84 zł	899,16 zł

W tym celu zdefiniować 2 funkcje „zestawienie” oraz „dodaj_osobe” (z parametrami: nazwisko, imię, stawka za godzinę, liczba godzin).

5. Utwórz funkcję generującą tabliczkę mnożenia od 1 do N w tabeli HTML, gdzie N jest argumentem funkcji.
6. Utwórz funkcję, której argumentem jest tekst będące imieniem i nazwiskiem jakiejś osoby. Funkcja ma zamienić pierwszą literę imienia i nazwiska na dużą, a resztę liter – na małe.