Krótki kurs JavaScript

Java Script jest językiem wbudowanym w przeglądarkę. Gdy ma się podstawy nabyte w innych językach programowania jest dość łatwy do opanowania.
JavaScript jest stosowany do powiększania możliwości i interaktywności stron WWW. Jednak trzeba pamiętać, że użytkownik ma możliwość wyłączenia obsługi tego języka. Należy tak projektować stronę aby nawet bez obsługi JavaScript pozostawała ona w pełni funkcjonalna. Skrypty powinno się dodawać w końcowym etapie projektowania strony WWW.

JavaScript nie ma możliwości bezpośredniego łączenia się z serwerem. Może jedynie wywołać skrypt napisany w innym języku, który jest utworzony do wykonywania operacji na serwerze jak odczyt danych czy operacje na bazie danych.

JavaScript nie ma też możliwości generowania grafiki, można jednak za jego pomocą manipulować istniejącymi plikami graficznymi.

Do pisania skryptów JavaScript nie wymagane jest posiadanie żadnego specjalistycznego oprogramowania, do tego celu można bezproblemowo posłużyć się zwykłym notatnikiem.

Przykładowe umieszczenie skryptu w pliku html:

```
<html>
<head>
<title>Pierwszy skrypt</title>
<script type=”text/javascript”>
//kod
</script>
</head>
<body>
<script type=”text/javascript>
alert(''Witaj, świecie'');
//pozostały kod
</script>
</body>
</html>
```

Umieszczanie skryptu w zewnętrznym pliku:

```
<script type="text/javascript" src="nazwa_pliku.js"></script>
```

Zmienne

```
<html><head>
<script type=''text/javascript''>
var sekundy = 60;
var minuty = 60;
var godziny = 24;
var sekundy_w _ciagu_dnia = sekundy * minuty * godziny;
```

```
</script>
</head>
<body>
<script type=''text/javascript''>
window.document.write(''doba ma '');
window.document.write(sekundy_w_ciagu_dnia);
window.document.write('' dziennie'');
</script>
</body>
</html>
```

Zmienne deklaruje się w następujący sposób: var sekundy;
Mimo, że wiele przeglądarek poprawnie zinterpretuje zapis: sekundy = 60; jednak powinno się używać pierwszego zapisu. Słowo var używamy wprowadzając zmienna poraz pierwszy.

Wprowadzając zmienną tekstowa musimy pamiętać o cudzysłowach:
var zmienna_tekstowa = "ciąg znaków";

Ciągi tekstowe można łączyć używając znaku "+"

```
var ciag1 = ''ala'';
var ciag2 = '' ma '';
var ciag3 = ''kota'';

var calosc = ciag1 + ciag2 + ciag3;
```

Javascript rozróżnia wielkość liter.
Zmienna nie może zawierać spacji.
Zmienne mogą przyjmować następujące wartości:
wartości liczbowe, łańcuchy znaków, wartości logiczne, wartość null.

Operatory Arytmetycze

Operator	Opis	Przykład
+	Dodawanie	x=3 x=x+4
-	Odejmowanie	x=4 x=6-x
*	Mnożenie	x=3 x=x*5
/	Dzielenie	10/5 9/2
%	Modulo (reszta z dzielenia)	4%3 12%8 8%2
++	Zwiększanie o 1	x=2 x++
--	Zmniejszanie o 1	x=4 x--

Operatory przypisania

Operator	Przykład	Równoważne z
=	x=y	
+=	x+=7	x=x+7
-=	x-=3	x=x-3
*=	x*=y	x=x*y
/=	x/=y	x=x/y
%=	x%=y	x=x%y

Funkcje

Funkcje służą do przetwarzania informacji. Mają one postać: funkcja().
Przykładem funkcji jest window.document.write('tekst'); Wyświetla ona zawartość nawiasów w oknie przeglądarki.

Swoje funkcje definiujemy na początku kodu strony - czyli w sekcji HEAD, a wywołujemy ją w dowolnym miejscu poniżej, jeżeli zajdzie taka potrzeba.

Aby zdefiniować funkcję po słowie function piszemy jej nazwę i w nawiasach argumenty, jakie chcemy jej przekazać. Kod wykonywany przez funkcję umieszczamy pomiędzy klamrą otwierającą { i zamykającą } która jest równoznaczna z końcem fuknkcji.

```
function funkcja() {
document.write("funkcja została wywołana")
}
```

Funkcja może zwracać wartość:

```
function funkcja() {
var a=1;
var b=2;
c = a + b;
return c;
}
```

Inne przykłady:
alert()
Wyświetla ona tekst w oknie dialogowym.
`alert(''wyświetlany tekst'');`
Po wywoałniu tej funkcji otworzy się okno z tekstem, które można zamknąc wciskając przycisk OK.
Jest ona przydatna do poinformowania użytkownika o błędzie, np. podczas analizy formularza.

prompt()
Wyświetla okno umożliwiające wprowadzenie informacji przez użytkownika.

```
var imie = prompt(''jak masz na imie'',''tu wpisz swoje imie'');
```
tu wpisz swoje imie pojawi się w polu w którym można wprowadzać dane
document.write(imie);

Instrukcje warunkowe i pętle

Wyrażenia if

```
składnia:
if(test)
{
wyrażenie1;
wyrażenie2;
wyrażenie3;
}
```

test umieszczony w nawiasach jest wyrażeniem logicznym. Wyrażenie może zwaracać jedynie prawdę lub fałsz.

```
if(a<2)
{
...
}
```

Jeśli zmienna a bedzie mniejsza od 2 zwrócona zostanie wartość *true*

Operatory porównania

Operator	**Opis**	**Przykład**
==	jest równe	2==3 wynik:fałsz
!=	nie jest równe	2!=3 wynik:prawda
>	jest większe	25>3 wynik:fałsz
<	jest mniejsze	2<3 wynik:prawda
>=	większe lub równe	25>=3 wynik:fałsz
<=	mniejsze lub równe	2<=3 wynik:prawda

Operatory logiczne

Operator	**Opis**	**Przykład**
&&	i	x=3 y=4 (x < 9 && y > 2) wynik:prawda
||	lub	x=3 y=4 (x==8 || y==6) wynik:fałsz
!	zaprzeczenie	x=3 y=4 !(x==y) wynik:prawda

Wyrażenia if-else

Gdy występuje więcej niż jeden warunek który chcemy sprawdzić możemy to zrobić kożystając z

konstrukcji:

```
if(test1)
{
...
} else if(test2)
{
...
} else {
...
}
```

Konstrukcja switch

W większości przypadków konstrukcję *if-else-if* możemy zastąpić instrukcją *switch*

```
switch (zmienna) {
case wartosc1:
kod wykonywany gdy zmienna=wartosc1
break
case wartosc2:
kod wykonywany gdy zmienna=wartosc2
break
case wartosc3:
kod wykonywany gdy zmienna=wartosc3
break
default:
kod wykonywany, gdy żadnej z powyższych wartości nie przyjmuje zmienna
}
```

Pętla while

Wyrażenia w pętli wykonuja się dopóki wynikiem testu jest wartość *true*

```
while (test) {
kod pętli
}
```

Pętla do-while
Podobna do powyżej z tą różnicą, że wykona się zawsze przynajmniej jeden raz

```
do {
kod wykonywany w pętli
} while (test)
```

Pętla for

Służy do wykonania kodu określoną ilość razy

```
for(inicjalizacja_zmiennej; warunek; zmiana_zmiennej) {
kod wykonywany w pętli
}
```

Obiekty
Obiekty są konstrukcjami programistycznymi posiadającymi tzw. właściwości, którymi mogą być zmienne lub inne obiekty.

Ogulnie do właściwości obiektu mamy dostęp stosując zapis:
```
nazwa_obiektu.nazwa_właściwości
```

np.:
```
kubek.pojemność = 1
kubek.kolor = czerwony
kubek.cena = 15
```

Aby kożystać z obiektu musimy najpierw go zdefiniować w funkcji zwanej konstruktorem:

```
function osoba(imie,nazwisko) {
this.nazwisko=nazwisko
this.imie=imie
}

function kubek(pojemnosc,kolor,cena,klient) {
this.pojemosc=pojemosc
this.kolor=kolor
this.cena=cena
this.klient=klient
}

klient = new osoba("Jan","Kowalski")
kubek = new kubek(1,"czerwony",20)
document.write("pojemosc: "+ kubek.pojemnosc +" kolor "+ kubek.kolor +" cena: "+
kubek.cena)
document.write("<br>Klientl: "+ kubek.klient.imie +" "+ kubek.klient.nazwisko)

}
```

Typy obiektów

- **array**
  Jest to tablica służąca do przechowywania zbioru wartoś**ci**
- **boolean**
  Służy do tworzenia i przechowywania wartości logicznych prawda lub fałsz.
- **Date**
  Do operacjach na dacie

  Math
  Do wykonywania skomplikowanych działań matematycznych

  Number

  String

Zdażenia
Skrypt sterowany zdażeniami uruchamiany jest dopiero wtedy, gdy użytkownik wykona określona czynność, lub nastąpi zmiana stanu strony. Na przykład najedzie kursorem na element graficzny.

Rodzaje zdażeń
onclick() wykonuje się gdy kliniemy na element

google

W powyższym przypadku w funkcji alert zamiast znaku cudzysłowu musimy użyć znaku apostrofu ('), ponieważ cudzysłów spowodowałby wcześniejsze zakończenie wprowadzania ciągu znaków i błąd. Jeżeli w ciagu musiwmy wprowadzić " musimy go poprzedzić backslashem - czyli \

onMouseOver, onMouseOut
odpowiednio wykonuja się po najechaniu kursorem na element i po opuszczeniu kursorem elementu

Inne najczęściej wykorzystywane rodzaje zdażeń:

zdarzenie	**opis**	**z obiektami**
onclick	zachodzi, gdy naciśniemy przycisk myszki na obiekcie	button, checkbox, Image, layer, link, radio, reset, submit
onmouseover	zachodzi, gdy najedziemy wskaźnikiem na obiekt	Image, layer, link
onmouseout	zachodzi, gdy opuszczamy wskaźnikiem obiekt	Image, layer, link
onmousedown	zachodzi, gdy trzymamy przyciśnięty przycisk myszki na obiekcie	Image, layer
onmouseup	zachodzi, gdy zwolnimy przycisk myszki na obiekcie	Image, layer
onblur	zachodzi, gdy opuszczamy obiekt (przejście na inny obiekt	password, select, text, textarea
onfocus	zachodzi, gdy wchodzimy na obiekt	password, select, text, textarea
onchange	zachodzi, gdy opuszczamy obiekt, którego zawartość została zmieniona	password, select, text, textarea
onselect	zachodzi, gdy zaznaczymy tekst wewnątrz pola tekstowego	password, text, textarea
onsubmit	zachodzi, gdy wysłamy formularz	form
onreset	zachodzi, gdy resetujemy formularz	form
onload	zachodzi w momencie załadowania dokumentu do przeglądarki	document, frame
onunload	zachodzi w momencie opuszczania strony	document, frame
onabort	zachodzi, gdy przerwane zostaje ładowanie strony	document, frame
onerror	zachodzi, gdy przeglądarka nie może załadować obrazu	Image

05.01.2008
Na podstawie książki Dave Thau - „JavaScript” i strony http://webmade.org/