Ćwiczenia – JavaScript

1. **Umieszczanie skryptu**

Kod JavaScript musi być zawarty pomiędzy znacznikami HTMLa <script> i </script>:

1.

<script>  
kod naszego skryptu  
</script>

2.

<html>  
<head>  
  
<script type="text/javascript">  
kod skryptu  
</script>  
  
<script type="text/javascript">  
kod skryptu  
/\* może być więcej rozgraniczonych znacznikami <script> kodów na jednej stronie \*/  
</script>  
  
</head>  
<body>  
  
<script type="text/javascript">  
tu - w dowolnym miejscy sekcji BODY - też można umieścić kod źródłowy skryptu  
</script>  
  
</body>  
</html>

**Ćwiczenie 1. - Umieszczanie skryptu w dokumencie HTML**

<html>

<body>

<script type="text/javascript">

<!--

alert("Hello World!");

//-->

</script>

</body>

</html>

**Cały skrypt jest objęty znakami komentarza (<!-- i -->), co gwarantuje, że kod nie zostanie wyświetlony na ekranie nawet przez stare przeglądarki nie obsługujące JS.**

**Kod źródłowy zamieszczony w oddzielnym pliku:**

<script type="text/javascript" src="nazwa\_pliku.js"></script>

1. **Komentowanie**

**Ćwiczenie 2. - Komentowanie**

<html>

<head>

<script type="text/javascript">

// Twoja przeglądarka nie obsługuje JavaScriptu.

/\* Aby zobaczyć stronę w pełnej funkcjonalności, zainstaluj inną przeglądarkę:

Internet Explorer, Netscape Navigator, Mozilla, Opera... \*/

<!-- ukrywamy kod przed nieznającymi nas przeglądarkami

kod skryptu

// koniec skryptu - koniec ukrywania -->

</script>

</head>

<body>

kod HTML strony

</body>

</html>

Dwa rodzaje komentarzy:

liniowy:

Zaczyna się od dwóch ukośników a kończy przy przejściu do następnej linii. Oznacza to, że przeglądarka zignoruje wszystko za znacznikiem // aż do końca linii, w której znacznik ten występuje.

blokowy:

Rozpoczyna się od sekwencji: /\* a kończy sekwencją: \*/. Oznacza to, że może on ciągnąć się przez wiele linii, przy czym niemożliwe jest jego zagnieżdżanie (czyli stosowanie jednego komentarza w innym).

1. **Obiekt Document**

Nasz skrypt będzie wypisywał na ekranie tekst. Posłużymy się do tego instrukcją document.write. Document jest to obiekt JavaScript, który reprezentuje aktualnie wyświetlaną stronę, natomiast write - to jego metoda, czyli funkcja wykonująca określone działania na obiekcie - w tym wypadku wypisuje tekst. Nasz tekst umieszczamy w nawiasach, jako argumenty wywołania metody. Ogólnie postępujemy następująco:

obiekt.metoda( argumenty metody )

**Ćwiczenie 3. Wykorzystanie jednej z metod obiektu document - write**

<html>

<head>

<script type="text/javascript">

document.write("<B> Hello World! </B>")

</script>

</head>

</html>

**Ćwiczenie 4. – wykorzystanie kilku własności obiektu Document**

Obiekt document służy do reprezentacji wczytanego do przeglądarki dokumentu HTML oraz zawiera szereg właściwości i metod pozwalających na jego modyfikację. Poprzez ten obiekt można otrzymać dostęp praktycznie do każdego elementu strony i za jego pomocą można tymi elementami manipulować. Poniżej zostały zaprezentowane wybrane właściwości i metody.

Document.Domain - Nazwa domenowa serwera, z którego pochodzi dokument.

Document.URL -Ciąg zawierający URL bieżącego dokumentu.

Document.lastModified - Zwiera datę i czas ostatniej modyfikacji dokumentu.

URL - Ciąg zawierający URL bieżącego dokumentu.

bgColor – Kolor tła dokumentu.

fgColor - Kolor tekstu dokumentu.

<!DOCTYPE html>

<html>

<body>

<script type="text/javascript">

document.bgColor="black";

document.fgColor="yellow";

document.write("Podstawowe informacje o dokumencie:<br />");

document.write("Nazwa domenowa serwera: ");

document.write(document.domain+ "<br />");

document.write("URL: " + document.URL + "<br />");

document.write("Data ostatniej modyfikacji: ");

document.write(document.lastModified + "<br />");

</script>

</body>

</html>

1. **Zmienne**

Zmienne tworzone w języku JavaScript mogą przechowywać dane wszystkich dostępnych typów. Odwołanie do wartości danej następuje przez nazwę zmiennej. Zmienne definiuje się poleceniem var.

Jeżeli chcemy korzystać w skrypcie z długich tekstów, bądź wypisywać skomplikowane liczby, wygodnie jest przypisać je do zmiennej. W JavaScript nie musimy deklarować, jakiego typu jest zmienna.

**Ćwiczenie 5. – definiowanie zmiennych**

<!DOCTYPE html>

<html>

<body>

<script type="text/javascript">

var imie="Jan"

var nazwisko=" Kowalski"

var rokurodzenia=1990

document.write(imie + nazwisko+ " urodził się w " + rokurodzenia + " roku. ")

</script>

</body>

</html>

1. **Operatory**

Operatory i przypisania należą do jednych z ważniejszych elementów JavaScript. Operatory wykonują rozmaite typy operacji, a przypisania nadają wartości zmiennym.

**Ćwiczenie 6. - Operatory arytmetycze**

<!DOCTYPE html>

<html>

<body>

<script type="text/javascript">

var pierwszaLiczba = 12;

var drugaLiczba = 5;

var trzeciaLiczba = pierwszaLiczba - drugaLiczba;

document.write("Pierwsza Liczba = " + pierwszaLiczba)

document.write("<br/> Druga Liczba = " + drugaLiczba)

document.write("<br/> Wynikiem operacji " + pierwszaLiczba + " + 5 jest ");

document.write(pierwszaLiczba + 5);

document.write(

"<br />Wynikiem operacji " + pierwszaLiczba + " - " + drugaLiczba + " jest ");

document.write(trzeciaLiczba);

document.write("<br />Wynikiem operacji 2 \* 8 jest ");

document.write(2 \* 8);

</script>

</body>

</html>

**Ćwiczenie 7. - Operatory przypisania**

<!DOCTYPE html>

<html>

<body>

<p>Zakladajac ze x=10 i y=5, oblicz wartość wyrażenia x \* y.</p>

<button onclick="myFunction()">Try it</button>

<p id="demo"></p>

<script>

function myFunction()

{

var x=10;

var y=5;

x\*=y;

var demoP=document.getElementById("demo")

demoP.innerHTML="x=" + x;

}

</script>

</body>

</html

"demo": Jest skrótem do demonstracji, więc oznacza to, że używasz <p> do wykazania w jaki sposób można uzyskać odwołanie do elementu o id przez javascript

demo jest wykorzystane jako przykład - może być obojętnie co wpisane tam ("ala", "kot" co kolwiek) najważniejsze jest <p> - bo służy do wykazania w jaki sposób można uzyskać odwołanie do elementu o id przez javascript, a ostatnią składową jest: document.getElementById – jest metodą uzyskującą dostęp do pierwszego elementu z określonym id – w tym przypadku „demo”.

**Ćwiczenie 8. - Operatory porównania**

<!DOCTYPE html>

<html>

<body>

<p>Zakladajac ze x=5, to wartosc logiczna rownania x>8 jest.</p>

<button onclick="myFunction()">Sprobuj</button>

<p id="demo"></p>

<script>

function myFunction()

{

var x=5;

document.getElementById("demo").innerHTML=x>8;

}

</script>

</body>

</html>

**- Operatory logiczne**

1. **Funkcje**

**Ćwiczenie 9. – Funkcja**

Pisząc nasze skrypty mamy zawsze dostęp do zestawu funkcji standardowych dla JavaScript. Należy do nich przykładowo funkcja alert - wyświetlająca okienko dialogowe z informacją i przyciskiem OK, albo prompt - wyświetlającej okienko, w którym użytkownik może wpisać wartość zwracaną potem do programu.

<!DOCTYPE html>

<html>

<head>

<script>

alert("Witam na mojej stronie!")

document.write("Miło mi Cię gościć "+ prompt("Podaj imię:") +" na mojej stronie")

</script>

</head>

</html>