



JĘZYK PROGRAMOWANIA
STRON HTML

J A V A S C R I P T

Wprowadzenie

Skryptowy język programowania

Wykonywany w kontekście przeglądarki internetowej

Dostęp do niemal wszystkich elementów składowych strony HTML

Możliwość reagowania na zdarzenia powstające w kontekście przeglądarki internetowej

Posiada kolekcję obiektów i działa na obiektach

Przykładowe zastosowania

Sprawdzanie poprawności formalnej formularza przed wysłaniem

Rozpowszechnianie niewielkich baz danych – np. lista adresowa

Dynamiczna zmiana wyglądu i zawartości stron w przeglądarce

Otwieranie, zamykanie i przesuwanie okien przeglądarki

Animacja na stronie przeglądarki

Wykonywanie obliczeń



**Kod JS jest dołączony do pliku
HTML i jest wykonywany:**

**w czasie wczytywania pliku
HTML do przeglądarki**

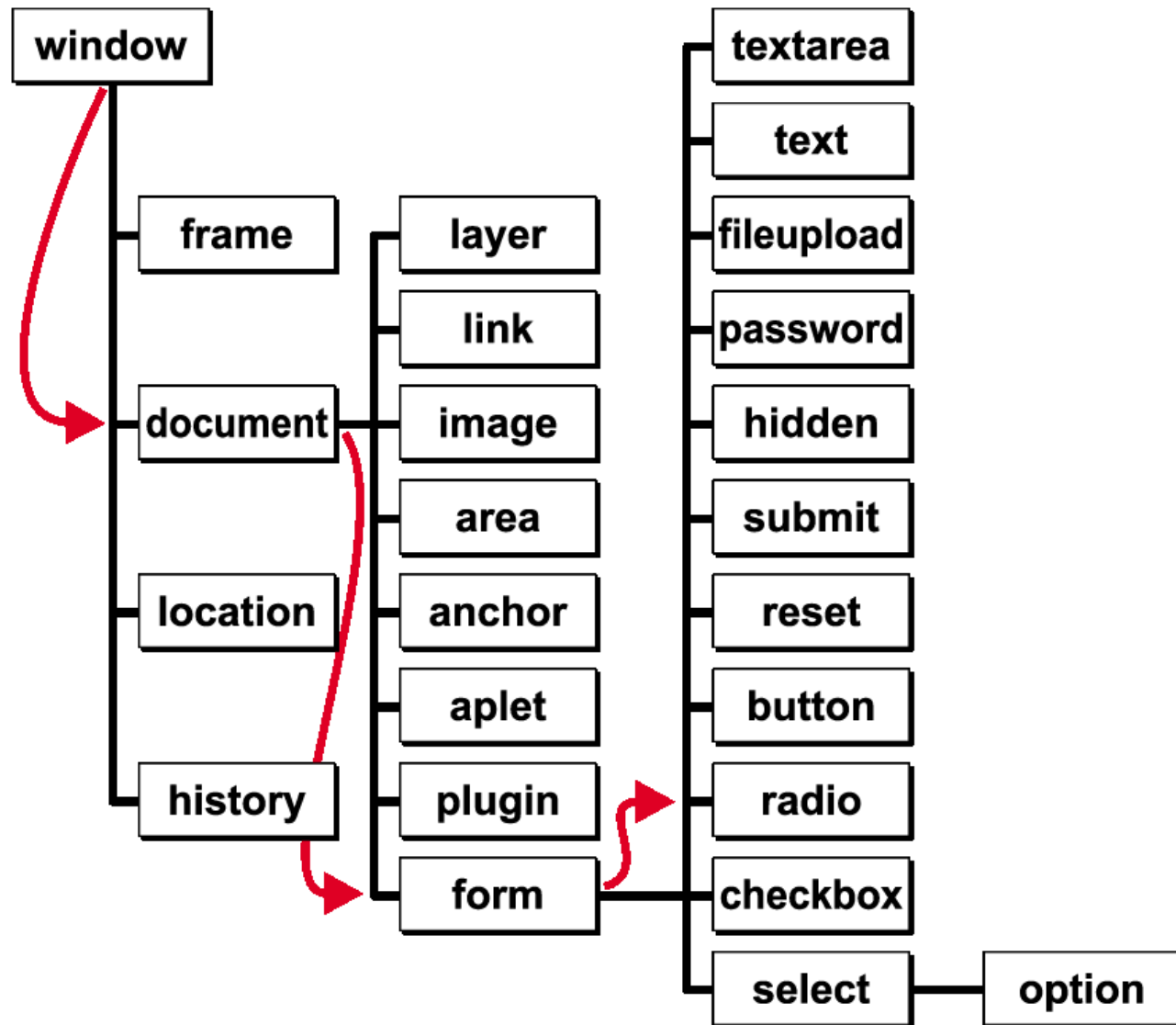
**w momencie wystąpienia
zdarzenia**

**może być uruchomiony w
określonych odstępach czasu**

**lub przetwarzany jako adres
URL**

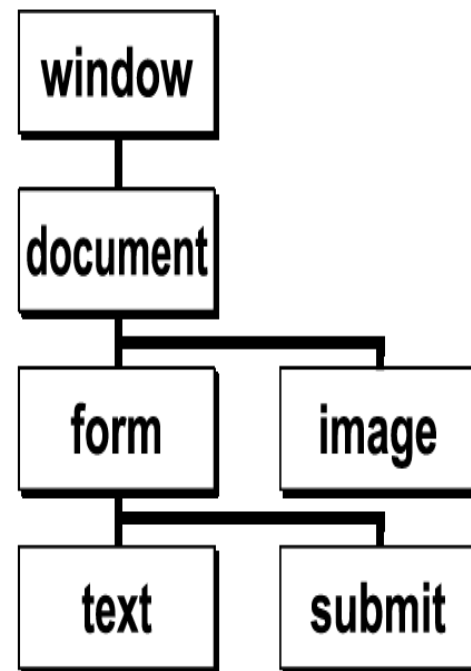
**Przetwarzanie
skryptu**

DOM Document Object Model



Tworzenie obiektów modelu DOM

```
<HTML>
<HEAD>
<TITLE>COŚ</TITLE>
</HEAD>
<BODY>
<FORM NAME=f1>
<INPUT TYPE=TEXT NAME=p1>
<INPUT TYPE=SUBMIT>
</FORM>
<IMG NAME=obraz SRC="o.gif">
</BODY>
</HTML>
```



window.document.f1.p1
window.document.obraz

Obiekt – właściwości

- Właściwości określają pewne atrybuty obiektów. Właściwości można odczytywać i/lub zmieniać. Niektóre właściwości są tylko do odczytu lub tylko do zapisu!!!
- Odczytanie wartości właściwości:
`nazwaObiektu.nazwaWłaściwości`
- Zmiana wartości właściwości
`nazwaObiektu.nazwaWłaściwości=wartość`

Obiekt – metody

Metoda definiuje akcje (czynności) jakie można wykonać na obiekcie. Wywołanie metody dla obiektu powoduje wykonanie akcji. Metodę można przyrównać do funkcji skojarzonej z obiektem.

- **`nazwaObiektu.nazwaMetody()`**
- **`NazwaObiektu.nazwaMetody(parametry)`**

Obiekt – zdarzenia

Każdy obiekt może reagować na pewne zdarzenia. Obsługa zdarzenia polega na wykonaniu zdefiniowanych dla danego obiektu i zdarzenia instrukcji języka.

- **<INPUT TYPE=BUTTON
onClick="window.alert('hej!')">**
- **<BODY onLoad="start()">**

Zdarzenia

- onAbort**
- onBlur**
- onChange**
- onClick**
- onDragDrop**
- onError**
- onFocus**
- onKeyDown**
- onKeyPress**
- onKeyUp**
- onLoad**

- onMouseDown**
- onMouseMove**
- onMouseOut**
- onMouseOver**
- onMouseUp**
- onMove**
- onReset**
- onResize**
- onSelect**
- onSubmit**
- onUnload**

Identyfikatory

Nazwa identyfikatora:

- musi zaczynać się od litery lub znaku podkreślenia _**
- może zawierać wyłącznie litery, cyfry i znak podkreślenia**
- Wielkość liter jest rozróżniana !!!**

Nazwy obiektów HTML nadawane są przez parametr NAME= dowolnego znacznika.

Umieszczanie kodu JS w HTML - I

Kod JS w pliku zewnętrznym: 

```
<SCRIPT LANGUAGE=JavaScript  
    TYPE="text/javascript"  
    SRC="urlPlikuZKodemJS">  
</SCRIPT>
```

UWAGA:

</SCRIPT> jest wymagany !!!

Umieszczanie kodu JS w HTML - II

Kod JS w pliku HTML:

```
<SCRIPT LANGUAGE=JavaScript  
    TYPE="text/javascript">
```

```
<!--
```

```
    ... instrukcje JS ...
```

```
//-->
```

```
</SCRIPT>
```

Umieszczanie kodu JS w HTML - III

Kod JS w obsłudze zdarzeń:

<ZNACZNIK ...

onZdarzenie="instrukcje JS" >

Umieszczanie kodu JS w HTML - IV

Kod JS jako adres URL:

...

Typy danych

•Tekstowy:

"KOWALSKI" lub 'Matejko'

•Liczbowy

123 lub 3.1412

•Logiczny

true lub false

•Null

•Obiekt

Zmienne

**Zmienne nie mają określonego typu.
Zawsze są typu danej jaką aktualnie
przechowują.**

Deklaracja:

var Imie

var kwartal = 'IV'

Wykorzystanie:

A = x + b * 3

Wyrażenia i operatory

Operatory

- Arytmetyczne: + - / * ^ % ++ --
- Porównania == != > >= < <=
- Przypisania = += -= *= /=
- Bitowe & | ^ << >>
- Logiczne && || ! ?:
- Konkatenacji +

Konwersje typów

Znaków na liczby:

Funkcja `parseInt("sznur znaków")`

Funkcja `parseFloat("sznur znaków")`

Liczb na tekst:

`10 + "zł"` daje `"10zł"`

Dowolnego na logiczny:

`true: !=0 "niepusty" obiekt`

`false: 0 "" null`

Definicja funkcji

```
function nazwaFunkcji ( parametry )  
{  
    instrukcje ...  
    return wartość  
}
```

```
function zm ( ) { cos+=cos }
```

```
function dod ( a,b ) { return a + b }
```

Instrukcja if ... else

if (warunek)

{

instrukcje ...

}

else

{

instrukcje ...

}

Pętla for

```
for ( wyr1 ; wyr2 ; wyr3 )
```

```
{
```

```
    instrukcje ...
```

```
    break (przerwij wykonanie)
```

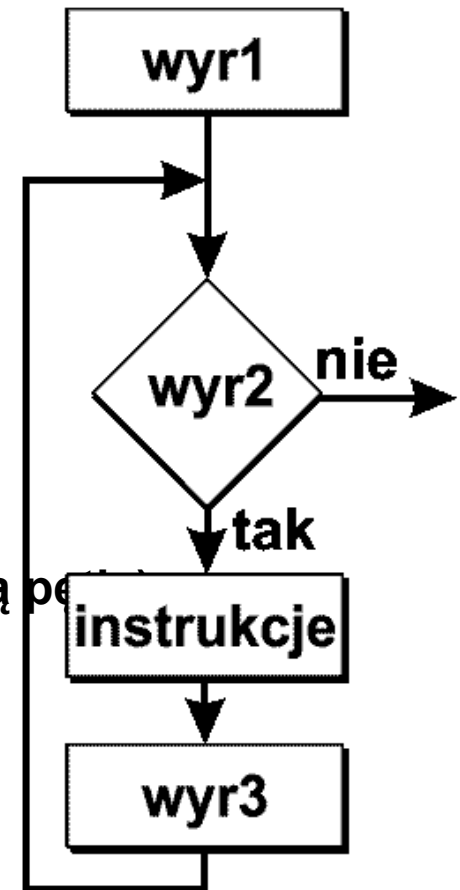
```
    continue (wymuś kolejną pętlę)
```

```
}
```

wyr1 - wartość początkowa

wyr2 – warunek

wyr3 - modyfikator



Pętle do, while

```
do
{
    instrukcje ...
    break
    continue
} while (warunek)
```

```
while (warunek)
{
    instrukcje ...
    break
    continue
}
```

Tablice

Deklaracja:

var nazwaT = new Array (ilość)

var nazwaT = new Array (el0,el1,...elN)

Wykorzystanie:

tab1[0] = "Kowalski"

nazwisko = tab[1]

Uwaga:

Elementy tablicy numerowane od 0 !!!

Tablice tylko jednowymiarowe.

Obiekt window

Wybrane właściwości:

**status, self, parent, name, opener,
document**

Wybrane metody:

open(URL, nazwaOkna, parametry)

close()

alert(tekst)

confirm(tekst) zwraca true lub false

prompt(tekst, wartpocz) zwraca wartość lub

null

Obiekt document

Wybrane właściwości:

bgColor, forms[], images[], title

Wybrane metody:

document.write(tekst)

document.writeln(tekst)

document.close()

Obiekt ciąg tekstowy

Właściwości obiektu string:

length

Metody obiektu string:

string.toUpperCase()

string.toLowerCase()

string.charAt(pozycja)

string.substring(n,m)

string.substr(n,ile)

string.indexOf(wzorzec,poz)

Dziękuję za uwagę.
Zapraszam do ćwiczeń.

A.Śmigielska