

METODA GETELEMENTBYID W JAVASCRIPT

Prowadząca: Karina Czarkowska

ZAKRES MATERIAŁU WYKŁADU

1. Rola metody **getElementById** i jej zastosowanie
2. Wywołanie metody
3. Przykłady zastosowań na stronach www
4. Słów kilka o właściwości **innerHTML**

ROLA METODY GETELEMENTBYID

Metoda **getElementById** jest metodą obiektu document i pozwala na dostęp do wybranego elementu witryny, zdefiniowanego dzięki unikatowemu atrybutowi **id**.

Wykorzystanie metody **getElementById** sprawdza się podczas obsługi elementów **input** formularzy, a także innych elementów strony www.

```
<form action="get">  
    <input type="text" name="pole" id="liczba1">  
</form>
```

ROLA METODY GETELEMENTBYID

```
<form>
    <input type="text" name="pole" id="liczba1">
</form>
```

Standard HTML dla znacznika **input** pozwala na utworzenie atrybutów **name** oraz **id** służących do identyfikacji tego znacznika.

Pierwszy atrybut **name** wykorzystywany jest m.in. przy przesyłaniu danych z formularzy na serwer.

Drugi atrybut **id** jest wykorzystywany przez skrypty JS uruchamiane przez przeglądarkę.

WYWOŁANIE METODY GETELEMENTBYID

Wywołanie w postaci:

document. getElementById("dzialanie")

daje dostęp do elementu o atrybucie **id** równym *dzialanie*.

Chcąc sprawdzić (zwrócić) wartość, jaką obecnie ma ten element, wystarczy odwołać się do pola **value** naszego obiektu document:

document.getElementById("dzialanie").value

WYWOŁANIE METODY GETELEMENTBYID

Jeśli wartość elementu o atrybucie **id** *dzialanie* będzie zawierać np. nazwę lub skrót nazwy działania arytmetycznego, to po dołączeniu instrukcji warunkowej oraz metody **alert()**, możemy w prosty sposób sprawdzić, czy działanie zostało wybrane prawidłowo:

```
if (document.getElementById("dzialanie").value=="do") {  
    alert("Wybierz działanie");  
    return;  
}
```

WYWOŁANIE METODY GETELEMENTBYID

Wykorzystanie tej metody sprawdzi się dobrze w zadaniu omawianym na naszych zajęciach - w kalkulatorze działań arytmetycznych.

Przeanalizujmy kod skryptu

PRZYKŁAD 1

Kalkulator działań matematycznych - dodawanie

```
<form>
  <input type="text" id="pierwsza">
  <input type="button" id="dodawanie" value="+">
  <input type="text" id="druga">
  <input type="button" id="przycisk" name="guzik" value="Oblicz"
        onclick="licz()">
  <input type="text" id="wynik"><br><br>
</form>
```

PRZYKŁAD 1

Kalkulator działań matematycznych - dodawanie

```
<script language="JavaScript">
function licz(){
var wynik=0;
if (document.getElementById("pierwsza").value==""){
alert("Wprowadź pierwszą wartość");
return;
}
if (document.getElementById("druga").value==""){
alert("Wrowadź drugą wartość");
return;
}
```

PRZYKŁAD 1

Kalkulator działań matematycznych - dodawanie

```
if (document.getElementById("dodawanie").value=="+"{  
wynik =  
parseFloat(document.getElementById("pierwsza").value)+parseFloat(  
document.getElementById("druga").value);  
}  
document.getElementById("wynik").value=wynik;  
}  
</script>
```

parseFloat ()- funkcja wbudowana, przetwarza argument w postaci łańcucha znaków zwracając liczbę zmiennoprzecinkową.

PRZYKŁAD 2

Obliczanie pola i obwodu prostokąta

```
<form>
```

```
Podaj bok a: <input type="text" id="a"> <br><br>
```

```
Podaj bok b: <input type="text" id="b"> <br><br>
```

```
<input type="button" id="pole" name="guzik" value="Pole"  
onclick="licz1()>
```

```
<input type="button" id="obwod" name="guzik" value="Obwód"  
onclick="licz2()>
```

```
<h2>Wynik:</h2>
```

```
<p id="p1">Tutaj pojawi się wynik obliczeń pola prostokąta</p>
```

```
<p id="p2">Tutaj pojawi się wynik obliczeń obwodu prostokąta</p>
```

```
</form>
```

PRZYKŁAD 2

Obliczanie pola i obwodu prostokąta

```
<script language="JavaScript">
function licz1(){
if (document.getElementById("a").value==""){
alert("Wprowadź wartość boku a");
return;
}
if (document.getElementById("b").value==""){
alert("Wprowadź wartość boku b");
return;
}
```

PRZYKŁAD 2

Obliczanie pola i obwodu prostokąta

```
else {  
    pole =  
    document.getElementById("a").value*document.getElementById("b")  
    .value;  
    }  
    document.getElementById("p1").innerHTML = pole+ '  
m2';  
}
```

Przykład

PRZYKŁAD 3

Dostęp do elementów formularza za pomocą metody getElementById().

```
<script language="JavaScript">
    function checkElement() {
        alert("Wartość wpisana w pole to:
        "+document.getElementById("Elem1").value);
    }
</script>

<form>
    <input type="text" id="Elem1" >
    <input type="button" value="Sprawdź" onClick="checkElement()"
>
</form>
```

PRZYKŁAD 3

Dostęp do elementów formularza za pomocą metody getElementById().

<input type="text" value="napis"/>	<input type="button" value="Sprawdź"/>
<p>Komunikat z bieżącej strony:</p> <p>Wartość wpisana w pole to: napis</p> <p><input type="button" value="OK"/></p>	

SŁÓW KILKA O WŁAŚCIWOŚCI INNERHTML

Właściwość innerHTML służy do pobierania i ustawiania zbioru zawartych w danym elemencie znaczników razem z ich treścią.

Inaczej mówiąc, innerHtml obsługuje obiekty będące elementami html. Za jej pomocą można odczytać lub zapisać to co znajduje się wewnątrz danego elementu, np.

HTML - <p id="p1"></p>

JS - **p1.innerHTML=(parseInt(a)+parseInt(b)+parseInt(c))/3**

PRZYKŁAD 4

Wypisanie wyniku w treści strony

Zadanie obliczania średniej z 3 liczb

ŽRÓDŁA

- http://www.crimsteam.site90.net/crimsteam/dom/dom_document_getelementbyid.html
- <http://webmaster.helion.pl/index.php/kursjs-obsluga-zdarzen-i-elementow-strony/kursjs-dostep-do-elementow-strony/29-kurs-javascript/376-kursjs-dostep-do-elementow-strony?tmpl=component&print=1&page=>
- <http://www.programowaniepc.pl/javascript-getelementbyid.php>
- http://kursjs.pl/kurs/hierarchia/hierarchia_tworzenie_i_usuwanie.php
- <https://developer.mozilla.org/pl/docs/Web/JavaScript/Referencje/Obiekty/parseFloat>