**Java Script**

1. **Umieszczenie skryptu js**
2. **W sekcji <head>**

<script>

Kod js

</script>

1. **W sekcji <body>**

<script>

Kod js

</script>

1. **Zewnętrzny plik z rozszerzeniem .js**

<script src="obliczenia.js"></script>

Uwaga: w pliku obliczenia.js nie stosujemy już znaczników <script> </script>

1. **Wyprowadzanie informacji**
2. **Poprzez modyfikacje paragrafu**

<p id="p1"></p>  
  
<script>  
 document.getElementById("p1").innerHTML = 5 + 6;  
</script>

1. **Za pomocą funkcji document.write()**

<script>  
 document.write(5 + 6);  
</script>

Uwaga: Po załadowaniu strony użycie document.write (np. w funkcji) nadpisuje całą zawartość strony.

1. **Okno dialogowe**

<script>  
 window.alert(5 + 6);  
</script>

1. **Konsola**

<script>  
 console.log(5 + 6);  
</script>

1. **Wprowadzanie informacji:**
2. **Okno dialogowe prompt**

<script>  
 var zmienna = prompt("Wprowadz tutaj wartosc"); //wartość string  
</script>

1. **Pole input**

<input type="text" id="in1">

<script>  
 var zmienna = document.getElementById(‘in1’).value; //wartość string  
</script>

1. **Pobieranie zawartości paragrafu**

<p id="p1">Zawartość</p>

<script>  
 var zmienna = document.getElementById(‘p1’).innerHTML //wartość string  
</script>

1. **Z formularza**

<form name="myForm" action="skrypt.php" **onsubmit="return validateForm()"**   
 method="post">  
 Name: <input type="text" name="fname">  
 <input type="submit" value="Submit">  
</form>

<script>

function validateForm() {  
    var x = document.forms["myForm"]["fname"].value;  
    if (x == "") {  
        alert("Pole Name musi być wypełnione");  
        return false;  
    }

return true;  
 }

</script>

1. Zapis stałych typu string zarówno w ‘paragrafie’  jak i "cudzysłowiu"
2. **Typy danych**
3. **Tablicowy**

var cars = ["Saab", "Volvo", "BMW"];

var cars = new Array("Saab", "Volvo", "BMW");

cars[0] = "Opel";

var x = cars.length;

1. **Obiektowy**

var person = {  
    firstName: "John",  
    lastName : "Doe",  
    id       : 5566,  
    fullName : function() {  
        return this.firstName + " " + this.lastName;  
    }  
};

document.write( **person.firstName**);

document.write( **person.fullName()** );

function Person(firstName, lastName, age, eyeColor) {  
    this.firstName = firstName;    
    this.lastName = lastName;  
    this.age = age;  
    this.eyeColor = eyeColor;  
    this.changeName = function (name) {  
        this.lastName = name;  
    };  
}

var myFather = new Person("John", "Doe", 50, "blue");  
var myMother = new Person("Sally", "Rally", 48, "green");

1. **Funkcje**

function myFunction(a, b) {  
    return a \* b;               
}

var x = myFunction(4, 3);

1. **Zdarzenia**
2. **W znaczniku**

<button onclick="funkcja()">Kliknij</button>

1. **Wpis do właściwości odpowiadającej zdarzeniu**

<button id="Bt1">Kliknij</button>

<script>  
 document.getElementById("Bt1").onclick = displayDate;

function displayDate () {  
     //kod funkcji  
 }  
</script>

1. **Za pomocą EventListener**

<button id="Bt1">Kliknij</button>

<script>

document.getElementById("Bt1").addEventListener("click",  
 displayDate);

function displayDate () {  
     //kod funkcji  
 }  
</script>

element.addEventListener("mouseover", myFunction);  
element.addEventListener("click", mySecondFunction);  
element.addEventListener("mouseout", myThirdFunction);

1. **Przydatne metody**
2. **Liczbowe**

var x = 9.656;  
x.toFixed(0);           // returns 10  
x.toFixed(2);           // returns 9.66  
x.toFixed(4);           // returns 9.6560  
x.toFixed(6);           // returns 9.656000

* parseInt() method
* parseFloat() method

isNaN(123) //false  
isNaN(-1.23) //false  
isNaN(5-2) //false  
isNaN(0) //false  
isNaN('123') //false  
isNaN('Hello') //true  
isNaN('2005/12/12') //true  
isNaN('') //false

1. **Tekstowe (string)**

var str = "Please locate where 'locate' occurs!";  
var pos = **str.indexOf**("locate",15);

var str = "Apple, Banana, Kiwi";  
var res = **str.substring**(7, 13);

str = "Please visit Microsoft!";  
var n = str.replace("Microsoft", "W3Schools");

var text1 = "Hello World!";       // String  
var text2 = **text1.toUpperCase**();  // text2 is text1 converted to upper

var text1 = "Hello World!";       // String  
var text2 = **text1.toLowerCase**();  // text2 is text1 converted to lower

var str = "       Hello World!        ";  
alert(**str.trim()**);

1. **Tablicowe (arrays)**

fruits.push("Kiwi");

var x = fruits.pop();

var x = fruits.shift();

cars.sort();

cars.reverse();

var points = [40, 100, 1, 5, 25, 10];  
points.sort(function(a, b){return a - b});

var points = [40, 100, 1, 5, 25, 10];  
points.sort(function(a, b){return b - a});

1. **Daty**

new Date()  
new Date(year, month, day, hours, minutes, seconds, milliseconds)  
new Date(milliseconds)  
new Date(date string)

|  |  |
| --- | --- |
| **Method** | **Description** |
| getFullYear() | Get the **year** as a four digit number (yyyy) |
| getMonth() | Get the **month** as a number (0-11) |
| getDate() | Get the **day** as a number (1-31) |
| getHours() | Get the **hour** (0-23) |
| getMinutes() | Get the **minute** (0-59) |
| getSeconds() | Get the **second** (0-59) |
| getMilliseconds() | Get the **millisecond** (0-999) |
| getTime() | Get the time (milliseconds since January 1, 1970) |
| getDay() | Get the weekday as a number (0-6) |

1. **Matematyczne**

Math.round(4.7);    // returns 5  
Math.round(4.4);    // returns 4

Math.pow(8, 2);      // returns 64

Math.sqrt(64);      // returns 8

Math.floor(x) returns the value of x rounded **down** to its nearest integer:

Math.floor(4.7);    // returns 4

1. **Generator liczb**

Math.floor(Math.random() \* 10);     // returns a random integer from 0 to 9

Math.floor(Math.random() \* 10) + 1;  // returns a random integer from 1 to 10

1. **Obiekty modelu DOM**
   1. **Metody pobierające obiekty ze strony HTML**

var **p1** = document.getElementById("p1"); // pojedynczy obiekt

**p1.innerHTML =** "Zmiana";

var p = document.getElementsByTagName("p"); // tablica obiektów

**p[0].innerHTML =** "Zmiana";

var ak = document.getElementsByClassName("akapity"); // tablica obiektów

**ak[0].innerHTML =** "Zmiana";

* 1. **Modyfikacja wyglądu elementów DOM CSS**

document.getElementById(*id*).style.*property*=*new style*

document.getElementById("p2").style.color = "blue";

document.getElementById("myP").style.fontFamily = "Impact,Charcoal,sans-serif";

document.getElementById("myP").style.fontSize = "16px";

document.getElementById("myDiv").style.border = "thick solid #0000FF";

* 1. **Animacja DOM**

var id = setInterval(frame, 5);  
  
function frame() {  
    if (/\* test for finished \*/) {  
        clearInterval(id);  
    } else {  
        /\* code to change the element style \*/    
    }  
}

1. **AJAX**
2. **Tekstowy**

<!DOCTYPE html>  
<html>  
<body>  
  
<div id="demo">  
  <h2>Let AJAX change this text</h2>  
  <button type="button" onclick="loadDoc()">Change Content</button>  
</div>  
  
</body>  
</html>

function loadDoc() {  
  var xhttp = new XMLHttpRequest();  
  xhttp.onreadystatechange = function() {  
    if (this.readyState == 4 && this.status == 200) {  
     document.getElementById("demo").innerHTML = this.responseText;  
    }  
  };  
  xhttp.open("GET", "ajax\_info.txt", true);  
  xhttp.send();  
}

1. **XML**

function loadDoc() {  
  var xhttp = new XMLHttpRequest();  
  xhttp.onreadystatechange = function() {  
    if (this.readyState == 4 && this.status == 200) {  
    myFunction(this);  
    }  
  };  
  xhttp.open("GET", "cd\_catalog.xml", true);  
  xhttp.send();  
}  
function myFunction(xml) {  
  var i;  
  var xmlDoc = **xml.responseXML;**  
  var table="<tr><th>Artist</th><th>Title</th></tr>";  
  var x = xmlDoc.getElementsByTagName("CD");  
  for (i = 0; i <x.length; i++) {   
    table += "<tr><td>" +  
    x[i].getElementsByTagName("ARTIST")[0].childNodes[0].nodeValue +  
    "</td><td>" +  
    x[i].getElementsByTagName("TITLE")[0].childNodes[0].nodeValue +  
    "</td></tr>";  
  }  
  document.getElementById("demo").innerHTML = table;  
}

1. **JQUERY**
2. **Dołączenie biblioteki**

<head>  
<script src="jquery-3.3.1.min.js"></script>  
</head>

<head>  
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js"></script>  
</head>

1. **Narzucony przepływ sterowania**

$(document).ready(function(){  
    $("p").click(function(){  
        $(this).hide();

Basic syntax is: **$(*selector*).*action*()**

$(this).hide() - hides the current element.

$("p").hide() - hides all <p> elements.

$(".test").hide() - hides all elements with class="test".

$("#test").hide() - hides the element with id="test".