



JavaScript. DOM. jQuery

ДІАНА СІРЕНКО

diana@4enjoy.com

АНДРІЙ ЧАЙКА

chaika@4enjoy.com



Is it true?

Java == JavaScript



JavaScript

JavaScript is a **high-level, dynamic, untyped,** and **interpreted** programming language.

Alongside **HTML** and **CSS**, it is **one of the three essential technologies of World Wide Web** content production

Типи даних

```
var person;           // Value is undefined, type is undefined  
var person = undefined;  
var person = null; // Value is null, but type is still an object  
var is = true;  
var number = 5.25;  
var lastName = "Chaika";  
var something = ["Andrii", 23, "Olexandr", null, true];  
var x = {  
    firstName: "Andrii",  
    lastName: "Chaika",  
    age: 22  
};
```

Рядкові літерали

// Single quote inside double quotes

```
var answer = "It's the JavaScript course";
```

// Single quote inside double quotes

```
var answer = "It is the 'JavaScript' course";
```

// Double quotes inside single quotes

```
var answer = 'It is the "JavaScript" course';
```

Перетворення типів

```
1 + 2.03 + "Car" + 4 + 1.01;  
"3.03Car41.01"
```

```
"4" + 3;  
"43"
```

Перетворення типів

`parseInt("2", 10);` *//2*

`parseInt("2.34", 10);` *//2*

`parseInt("a");` *//NaN (Not a Number)*

`parseFloat("2.34");` *//2.34*

`parseInt("5", 10) + 2;` *//7*

Порівняння типів

`1 == 1; //true`

`1 == "1"; //true`

`1 === 1; //true`

`1 === "1"; //false`

`2 != "2"; //false`

`2 !== "2"; //true`

Порівняння типів

```
if(a) {  
    //true  
    //Будь-яке число не 0 (Наприклад: 5, -1.5)  
    //Будь-який не порожній рядок  
    //Будь-який масив або об'єкт  
}
```

```
if(!a) {  
    //false  
    //0  
    //null  
    //undefined  
    //"" (порожній рядок)  
}
```

Неінтуїтивне порівняння

***//УВАГА! Не порівнювати з null чи
з undefined через ==***

```
if(a == undefined) {  
    //a = 0;    //false  
    //a = 1;    //false  
    //a = null; //true  
    //a = "";   //false  
    //a = undefined //true  
}
```



JavaScript functions

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

```
var x = myFunction(4, 3);
```

JavaScript scope

In JavaScript, scope is the set of variables, objects, and functions you have access to.

```
var T = 54;  
function a(x) {  
    var K = 1;  
  
    function b(y) {  
        return x + y + T + K;  
    }  
  
    return b(5);  
}
```

```
a(1); //61  
a(10); //70
```

Console Log

```
console.log("Message", x);  
console.warn("Warning", x);  
console.error("Error", x);
```

Message Object {firstName: "Andrii", lastName: "Chaika", age: 22}

⚠ Warning Object {firstName: "Andrii", lastName: "Chaika", age: 22}

✖ ▶ Error Object {firstName: "Andrii", lastName: "Chaika", age: 22}

Масиви в JavaScript

`arr = [];` *//Порожній масив*

`arr = [1,2];` *//Масив з елементами*

`arr = [1, "A"];` *//Масив змішаний*

`arr = [[1, 2], [1]];` *//Масиви в середині масиву*

`arr.push(2);` *//Додає 2 в кінець масива*

`arr.length;` *//!Не функція - довжина масива*

`arr[2];` *//2-ий елемент масива або undefined якщо такого немає*



JavaScript forEach

```
function sum(arr) {  
    var result = 0;  
    arr.forEach(function(value){  
        result += value;  
    });  
    return result;  
}
```

```
sum([1,2,3]); //6
```

JavaScript forEach

```
function logNumber(number, i) {  
    console.log("Number", number, i);  
}
```

```
[10,20,30].forEach(logNumber);
```

```
var arr = [10,20,30];  
arr.forEach(logNumber);
```

Number	10	0
Number	20	1
Number	30	2



DOM

The **Document Object Model** (DOM) is a cross-platform and language-independent convention for representing and interacting with objects in **HTML**

DOM getElementById

```
<p id="p1">
```

This is a text.

This is a text.

This is a text.

```
</p>
```

```
var p1 = document.getElementById('p1');
```

```
p1.style.color = "red";
```

```
p1.style.fontSize = "20px";
```

DOM не дуже зручний

```
var p = document.createElement("p");
p.appendChild(document.createTextNode("more content"));
document.getElementById("container").appendChild(p);
```

До

```
<div id="container"></div>
```

Після

```
<div id="container">
  <p>more content</p>
</div>
```

jQuery

It is a cross-platform **JavaScript library**. jQuery's syntax is designed to make it easier to **navigate a document, select DOM elements, create animations, handle events.**

jQuery is the most popular JavaScript library in use today.



Is it true?

jQuery == DOM



jQuery Селектори

1. Element selector

```
$("p");  
$("b");
```

2. Id selector

```
<p id="test_1">This is a paragraph.</p>  
$("#test_1");
```

3. Class selector

```
<p class="test_2">This is another paragraph.</p>  
$(".test_2");
```

jQuery Mixed Selectors

`<h4 class="test">This is a heading</h4>`

`<div class="test">Div text with class.</div>`

`<div>`

Text without class 1.

`Span text with class.`

Text without class 2.

`</div>`

`<div>`

Text without class 3.

`</div>`

This is a heading

Div text with class.

Text without class 1. Span text with class. Text without class 2.

Text without class 3.

`$("div.test").hide();`

jQuery Mixed Selectors

`<h4 class="test">This is a heading</h4>`

`<div class="test">Div text with class.</div>`

`<div>`

Text without class 1.

`Span text with class.`

Text without class 2.

`</div>`

`<div>`

Text without class 3.

`</div>`

This is a heading

Div text with class.

Text without class 1. Span text with class. Text without class 2.

Text without class 3.

`$("div.test").hide();`

This is a heading

Text without class 1. Span text with class. Text without class 2.

Text without class 3.

jQuery Mixed Selectors

`<h4 class="test">This is a heading</h4>`

`<div class="test">Div text with class.</div>`

`<div>`

Text without class 1.

`Span text with class.`

Text without class 2.

`</div>`

`<div>`

Text without class 3.

`</div>`

This is a heading

Div text with class.

Text without class 1. Span text with class. Text without class 2.

Text without class 3.

```
$("div.test").hide();
```

```
$("div").find(".test").hide();
```

jQuery Mixed Selectors

`<h4 class="test">This is a heading</h4>`

`<div class="test">Div text with class.</div>`

`<div>`

Text without class 1.

`Span text with class.`

Text without class 2.

`</div>`

`<div>`

Text without class 3.

`</div>`

This is a heading

Div text with class.

Text without class 1. Span text with class. Text without class 2.

Text without class 3.

`$("div.test").hide();`

`$("div").find(".test").hide();`

This is a heading

Div text with class.

Text without class 1. Text without class 2.

Text without class 3.

jQuery Events

```
<div class="background-class">  
  Title text<br>  
  <button>Click me</button>  
</div>
```

```
$("#button").click(function(){  
  $(".background-class").css("color", "green");  
  $(this).css("color", "#a11bff");  
});
```



jQuery Events

Mouse Events – **click**, mouseenter, mouseleave

Keyboard Events – keypress, keydown, keyup

Form Events – submit, change, **focus**

Window Events – **ready**, scroll, resize

http://www.w3schools.com/jquery/jquery_ref_events.asp

jQuery зміна HTML елемента

```
$("#container").html(  
    '<a href="https://google.com/">Link</a>'  
);
```

```
console.log($("#container").html());
```

```
//<a href="https://google.com/">Link</a>
```

До

```
<div id="container">  
    <p>something</p>  
</div>
```

Після

```
<div id="container">  
    <a href="https://google.com/">Link</a>  
</div>
```

jQuery зміна Тексту

```
$("#container").text("Hello, it's me");
```

```
console.log($("#container").text());  
//Hello, it's me
```

До

```
<div id="container">  
  <p>something</p>  
</div>
```

Після

```
<div id="container">  
  Hello, it's me  
</div>
```

jQuery зміна Атрибуту

```
$("#container").attr("id", "container2");
```

```
console.log($("#container2").attr("id"));  
//container2
```

До

```
<div id="container">  
  <p>something</p>  
</div>
```

Після

```
<div id="container2">  
  <p>something</p>  
</div>
```

jQuery зміна вмісту текстового поля

```
$("#text").val("New Value");
```

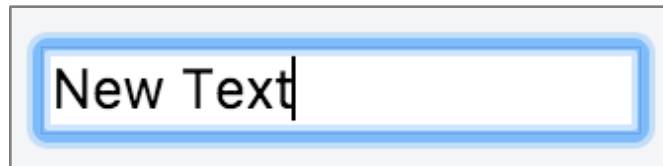
```
console.log($("#text").val());  
//New Value
```

До

```
<input id="text" value="Old Text" />
```

Після

```
<input id="text" value="New Text" />
```



jQuery Get/Set

```

$("#btn1").click(function(){
    $("#test1").text("JavaScript");
    alert($("#test1").text());
});
$("#btn2").click(function(){
    $("#test2").html("<b>is the best</b>");
    alert($("#test2").html());
});
$("#btn3").click(function(){
    $("#test3").val("language");
    alert($("#test3").val());
});
$("#btn4").click(function(){
    $("#test4").attr("value", "!!!");
    alert($("#test4").attr("value"));
});

```

JavaScript

is the best

Input field:

Input field:

Set Text

Set HTML

Set Value

Set Attr

jQuery CSS classes

```
$("#button").click(function(){
    $("#text1").addClass("important blue");
    $("#text2").removeClass("important");
});
```

```
.important {
    font-weight: bold;
    font-size: 45px;
}
.blue {
    color: blue;
}
```

```
<div id="text1">This is Text 1.</div>
```

```
<div id="text2" class="important">This is Text 2.</div>
```

This is Text 1.

This is Text 2.

This is Text 1.

This is Text 2.

jQuery animation

```
function updateNode(node, fn) {
  node.fadeOut(250, function() {
    fn();
    node.fadeIn(250);
  });
}
```

Назва товару		Додати
Печиво	<div>-</div> <div>1</div> <div>+</div>	<div>Куплено</div> <div>×</div>
Сир	<div>1</div>	<div>Не куплено</div>
Апельсин	<div>-</div> <div>4</div> <div>+</div>	<div>Куплено</div> <div>×</div>

jQuery Створення елементів

```
var p = $("<p>more content</p>");  
$("#container").append(p);
```

```
var p = document.createElement("p");  
p.appendChild(document.createTextNode("more content"));  
document.getElementById("container").appendChild(p);
```

До

```
<div id="container"></div>
```

Після

```
<div id="container">  
  <p>more content</p>  
</div>
```

jQuery Add/Remove

```
var LIST = $('.list-of-items');
var ITEM_TEMPLATE = $('.one-item').html();

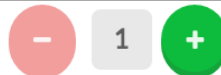
function addItem(title) {
    var node = $(ITEM_TEMPLATE); //Create new HTML node

    node.find(".title").text(title); //Set product title

    //Delete Action
    node.find(".delete-button").click(function(){
        node.remove();
    });

    LIST.append(node); //Add to the end of the list
}
```

Печиво



Куплено



jQuery each

```
$(".items").each(function(i, item){
    item = $(item); //to have access to all jQuery functions

    console.log(item.find('.title').text());
});
```

Залишилося	
Печиво 1	апельсини 4
Куплено	
Сир 1	

JavaScript: підключення

```
<html>
  <head>
    ...
  </head>
  <body>

    ...

    <script type="text/javascript" src="https://code.jquery.com/jquery-2.1.4.min.js"> </script>
    <script type="text/javascript" src="js/main.js"> </script>

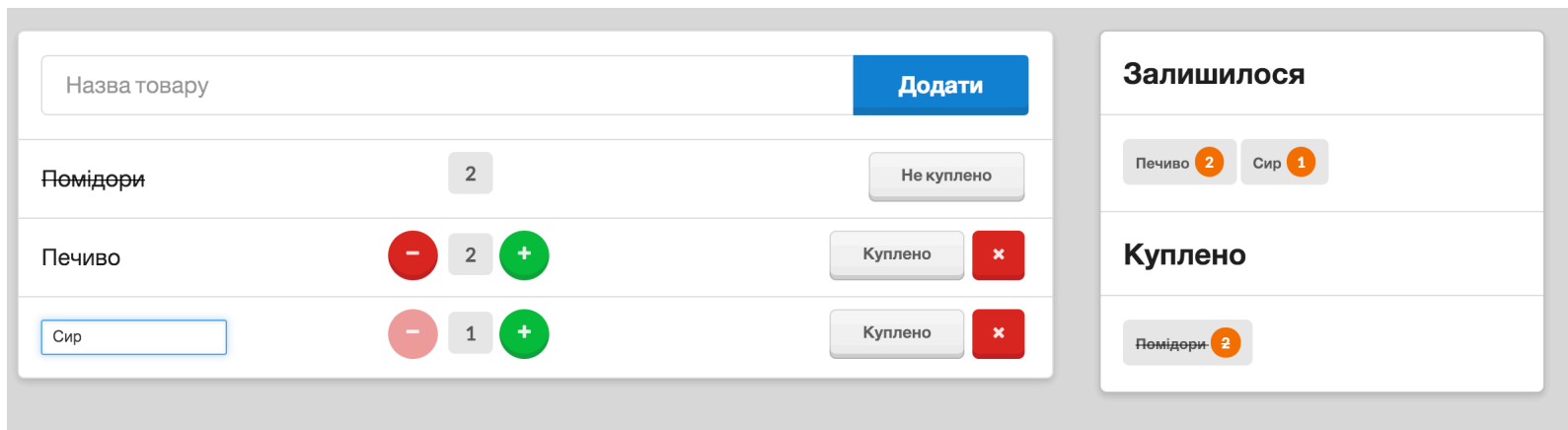
  </body>
</html>
```

jQuery Ремапка

```
//You should wrap all your jQuery code  
$(function()) {  
    //This function will be called  
    // after the DOM was loaded and ready  
  
    //Here we can place all jQuery Code  
    $('button').click(function() { ... });  
});
```


Домашнє завдання

Завдання: Оживити кошик покупок



The screenshot shows a shopping cart application interface. On the left, there is a table with columns for item name, quantity, and status. The items are: Помідори (2, Не куплено), Печиво (2, Куплено), and Сир (1, Куплено). Each item has a minus button, a quantity input, and a plus button. A 'Додати' button is at the top right of the table. On the right, there are two sections: 'Залишилося' (Remaining) showing Печиво (2) and Сир (1), and 'Куплено' (Purchased) showing Помідори (2).

Назва товару	Додати
Помідори 2	Не куплено
Печиво - 2 +	Куплено ✕
Сир - 1 +	Куплено ✕

Залишилося
Печиво 2 Сир 1

Куплено
Помідори 2

<http://js-buylist.oop.in.ua/>