# Intro

## Start

### Include library

<!--

http://jquery.com/ - официальный сайт библеотеки JQuery

http://api.jquery.com/ - документация

-->

<!DOCTYPE html>

<html>

<head>

<title>Знакомство с библиотекой JQuery. Подключение удаленной библиотеки.</title>

<!-- Подключение удаленной библиотеки (CDN) -->

<script src="http://code.jquery.com/jquery-1.10.1.min.js"></script>

<!-- Подключение локальной библиотеки -->

<!-- <script src="jquery-1.10.2.js"></script> -->

<script>

jQuery(); // Главная функция для использования всех возможностей jQuery.

$(); // Псевдоним функции jQuery() с тем же набором функционала.

</script>

</head>

<body>

<p>JQuery - write less, do more.</p>

<p><a href="http://jquery.com/" target="\_blank">Official site</a></p>

<p><a href="http://api.jquery.com/" target="\_blank">Documenatation</a></p>

</body>

</html>

### Ready

<head>

<title>Метод ready().</title>

<script src="jquery-1.10.2.js"></script>

<script>

/\*

Событие ready происходит в момент готовности DOM-дерева, что происходит раньше начала загрузки "тяжелых" мультимедийных файлов.

Это оптимальный момент, когда можно приступить к установке обработчиков различных событий и выполнять другой подготовительный javascript-код. ВАЖНО!

Если подготовительные действия требуют наличия таких элементов как картинки,

правильным будет полагаться на событие load. Однако, имейте ввиду, что одновременное использование .ready() и <body onload="..."> приведет к ошибке.

Если вам необходимо использовать оба события, используйте метод библиотеки jQuery .load(), вместо <body onload="...">.

\*/

jQuery(document).ready(function () {

alert('Структура страницы сформирована, можно приступать!');

jQuery("p").text("Hello World (jQuery)");

});

// $() - псевдоним функции jQuery()

//$(document).ready(function () {

// alert('Структура страницы сформирована, можно приступать!');

// $("p").text("Hello World (jQuery)");

//});

</script>

</head>

<body>

<p></p>

</body>

</html>

## Selectors

### SelectAll

<title>Селектор \*</title>

<script src="jquery-1.10.2.js"></script>

<script>

/\*

$("\*") - выбор всех элементов на странице.

css("a","b") - метод jQuery, позволяющий добавлять CSS-стили елементам.

a - имя CSS-свойства.

b - значение.

\*/

$(document).ready(function () {

$("\*").css("font-style", "italic");

});

</script>

</head>

<body>

<p>Paragraph in body</p>

<div>

<p>Paragraph in div</p>

<ul>

<li>Item 1</li>

<li>Item 2</li>

<li>Item 3</li>

<li>Item 4</li>

</ul>

</div>

</body>

### Select Tag Name

<title>Селектор имя тега</title>

<script src="jquery-1.10.2.js"></script>

<script>

$(document).ready(function () {

// $("E") - соответствует всем элементам с именем тега E.

// Выбор всех параграфов на странице и установка СSS свойству color значения red.

$("p").css("color", "red");

});

</script>

</head>

<body>

<p>Paragraph in body</p>

<div>

<p>Paragraph in div</p>

<ul>

<li>Item 1</li>

<li>Item 2</li>

<li>Item 3</li>

<li>Item 4</li>

</ul>

</div>

</body>

### Multiple select

<title>Группировка</title>

<script src="jquery-1.10.2.js"></script>

<script>

$(document).ready(function () {

// $("E,G,H") - соответствует всем элементам с именами тегов E, G, H.

// Выбор тегов "p" и "ul" на странице.

$("p,ul").css("color", "red");

});

</script>

</head>

### Select inherit

<title>Выбор вложенных элементов (потомков)</title>

<script src="jquery-1.10.2.js"></script>

<script>

$(document).ready(function () {

// $("E F") - соответствует всем элементам с именем тега F, вложенным в элемент с именем тега E.

// Выбор всех параграфов, вложенным в тег "div".

$("div p").css("color", "red");

});

</script>

</head>

### Neighborhood select

<head>

<title>Выбор соседних элементов.</title>

<script src="jquery-1.10.2.js"></script>

<script>

$(document).ready(function () {

// $("E + F") - соответствует всем элементам с именем тега F, которым непосредственно предшествует элемент E на том же уровне вложенности.

// Выбор всех элементов "div", которым непосредственно предшествует элемент "p" на том же уровне вложенности.

$("p + div").css("color", "red");

});

</script>

</head>

<body>

### Child select

<title>Выбор прямых потомков.</title>

<script src="jquery-1.10.2.js"></script>

<script>

$(document).ready(function () {

// $("E > F") - соответствует всем элементам с именем тега F, являющимися прямыми потомками элементов с именем тега E.

// Выбор всех элементов "span", являющимися прямыми потомками элементов "div".

$("div > span").css("color", "red");

});

</script>

</head>

### Has selector

<title>"Has"</title>

<script src="jquery-1.10.2.js"></script>

<script>

$(document).ready(function () {

// $("E:has(F)) - соответствует всем элементам с именем тега E, имеющим хотя бы один вложенный элемент с именем тега F.

// Выбор всех элементов с именем тега div, имеющим хотя бы один вложенный элемент с именем тега ul.

$("div:has(ul)").css("background", "red");

});

### Class select

<title>Выборка по имени класса</title>

<script src="jquery-1.10.2.js"></script>

<script>

$(document).ready(function () {

// $("E.C) - соответствует всем элементам E с именем класса C. При отсутствии E эквивалентен селектору \*.C

// Выбор всех элементов "p" с именем класса "red".

$("p.red").css("color", "red");

});

### Id select

<title>Выборка по идентификатору</title>

<script src="jquery-1.10.2.js"></script>

<script>

$(document).ready(function () {

// $("E#I) - соответствует всем элементам E с именем идентификатора I.

// Выбор всех элементов "p" с идентификатором "red".

$("p#red").css("color", "red");

});

### Sibling select

<title>Выборка "братьев".</title>

<script src="jquery-1.10.2.js"></script>

<script>

$(document).ready(function () {

// $("E ~ F") - соответствует всем элементам с именем тега F, которым предшествует элемент E на том же уровне вложенности.

// Выбор всех элементов "p", которым предшествует элемент "span" на том же уровне вложенности.

$("span ~ p").css("color", "red");

});

</script>

### Select attribute1

<title>Выборка по атрибутам.</title>

<script src="jquery-1.10.2.js"></script>

<script>

$(document).ready(function () {

// $("E[A]") - соответствует элементам E с атрибутом A, имеющим любое значение.

// Выбор всех элементов "a", с атрибутом "target", имеющим любое значение.

$("a[target]").css("color", "red");

});

### Select attribute2

<title>Выборка по атрибутам.</title>

<script src="jquery-1.10.2.js"></script>

<script>

$(document).ready(function () {

// $("E[A=V]") - соответствует элементам E с атрибутом A, который равен V.

// Выбор всех элементов "a", с атрибутом "target", имеющим значение text.

// Метод val() - метод, который позволяет получить значение атрибута "value" первого элемента в наборе соответствующих селектору элементов или установить новое значение (переданное в качестве аргумента) для всех елементов, подходящих под выборку.

$("input[type=text]").val("admin");

});

### Select attribute3

<title>Выборка по атрибутам.</title>

<script src="jquery-1.10.2.js"></script>

<script>

$(document).ready(function () {

/\*

$("E[A^=V]") - соответствует элементам E с атрибутом A, который начинается с V.

$("E[A$=V]") - соответствует элементам E с атрибутом A, который заканчивается на V.

$("E[A\*=V]") - соответствует элементам E с атрибутом A, который содержит V.

\*/

$("a[href^='http://']").css("color", "red"); // Выбор всех элементов "a", с атрибутом "href", который начинается c "http://"

$("a[href$='.com']").css("color", "blue"); // Выбор всех элементов "a", с атрибутом "href", который заканчивается на ".com"

$("a[href\*='html']").css("color", "green"); // Выбор всех элементов "a", с атрибутом "href", который содержит "html"

});

</script>

<style>

span {

display: inline-block;

width: 80px;

}

</style>

</head>

<body>

<a href="http://edu.cbsystematics.com/video/ru">Видео обучение</a><br />

<a href="http://jquery.com">jQuery</a><br />

<a href="http://htmlbook.ru/">HTML-book</a>

</body>

</html>

## Samples

### Eq method

<title>Выборка элементов с помощью jQuery</title>

<script src="jquery-1.10.2.js"></script>

<script>

// eq() - метод, который позволяет выбрать нужный элемент из массива по индексу.

jQuery(document).ready(function () {

$('p').eq(0).text("$('p') - выборка по имени тега");

$('[name = "input\_background"]').css('background', 'yellow');

$('.bold\_paragraph').css('fontWeight', 'bold');

$('#red\_paragraph').css('color', 'red');

});

</script>

</head>

<body>

<p></p>

<input type="text" name="input\_background" />

<p class="bold\_paragraph"><u>$('.bold\_paragraph')</u> - выборка по имени класса.</p>

<p id="red\_paragraph"><u>$('#red\_paragraph')</u> - выборка по идентификатору</p>

</body>

</html

### Query selector method

<title>Выборка элементов с помощью методов querySelector() и querySelectorAll()</title>

<script>

window.addEventListener("load", function () {

// querySelector() - получение первого элемента на странице, который соответствует селектору.

// В данном случае возвращаем элемент p.

var paragraph = document.querySelector("p");

paragraph.style.fontSize = "30px";

// querySelectorAll() - возвращает массив элементов, которые соответствуют селектору.

// В данном случае возвращается массив элементов списка, находящихся в элементе ul с атрибутом class="list"

var list = document.querySelectorAll(".list > li");

for (var i = 0; i < list.length; i++) {

list[i].style.color = "blue";

}

});

</script>

</head>

<body>

<p>Метод querySelector()</p>

<p>Параграф без стилей</p>

<ul class="list">

<li>Item 1</li>

<li>Item 2</li>

<li>Item 3</li>

<li>Item 4</li>

</ul>

</body>

</html>

### On method

ad>

<title>Добавление классов к елементам с помощью jQuery</title>

<script src="jquery-1.10.2.js"></script>

<script>

// on() - Добавление обработчика событий текущему выбранному элементу. Он обеспечивает все функциональные возможности, необходимые для присоединения обработчиков событий.

// addClass() - Добавление указанного класса(ов) к каждому из набора соответствующих элементов.

// removeClass() - Удаляет классы в наборе соответствующих элементов.

$(document).ready(function () {

$('#first').on('click', firstButtonHandler);

$('#second').on('click', secondButtonHandler);

var div = $('#testDiv');

function firstButtonHandler() {

div.removeClass();

div.addClass("class1");

}

function secondButtonHandler() {

div.removeClass();

div.addClass("class2");

}

});

</script>

<style>

.class1 {

font-family: Tahoma;

color: Blue;

}

.class2 {

font-family: Courier New;

color: Red;

}

</style>

</head>

<body>

<input id="first" type="button" value="First" />

<input id="second" type="button" value="Second" />

<div id="testDiv">

Some text

</div>

</body>