



JavaScript library

БИБЛИОТЕКА JQUERY

- Библиотека JavaScript, фокусирующаяся на взаимодействии JavaScript и HTML.
- Помогает легко получать доступ к любому элементу DOM, обращаться к атрибутам и содержимому элементов DOM, манипулировать ими.
- Удобный API для работы с AJAX.

- Библиотека jQuery – кроссбраузерная, предоставляет набор базовых функций, одинаково работающих во всех браузерах
- Поддержка языка JavaScript отличается от браузера к браузеру, появление библиотек упростило процесс разработки RIA-приложений (Rich Internet Application)
- <http://jquery.com>

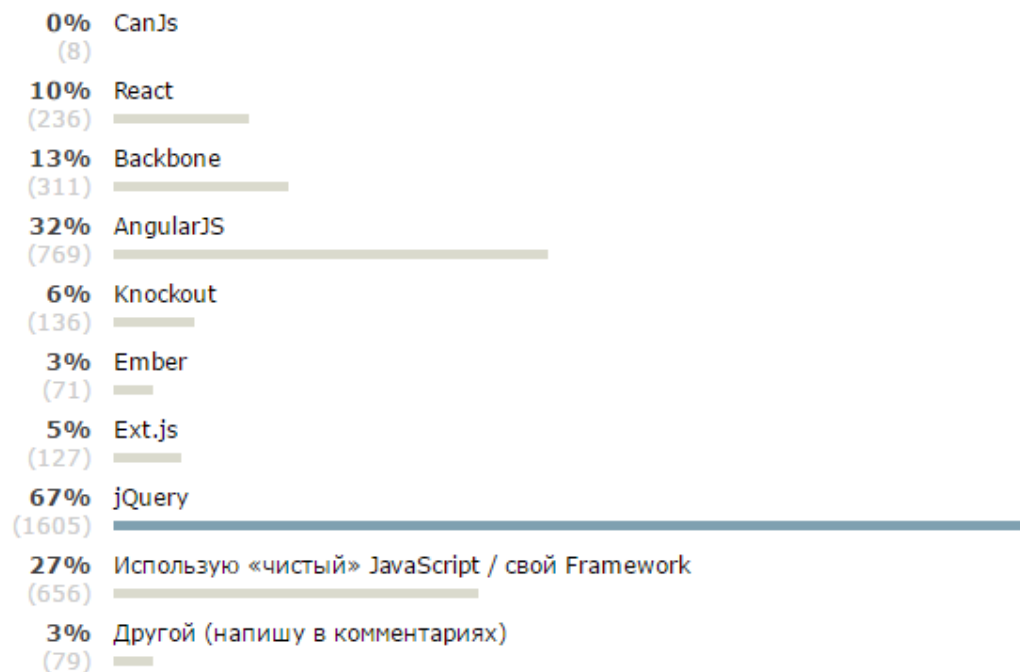
Из истории jQuery

- Библиотека была представлена общественности на компьютерной конференции «BarCamp» в Нью-Йорке в 2006 году
- Автор John Resig
<http://ejohn.org/about/>



Популярность jQuery

Какой JavaScript Framework используете вы?



Проголосовало 2404 человека. Воздержалось 967 человек.

Хабрахабр май 2015 (<https://habrahabr.ru/post/257825/>)



write less, do more.

PluginsContributeEventsSupportjQuery Foundation

DownloadAPI DocumentationBlogPluginsBrowser Support

Search



Lightweight Footprint

Only 32kB minified and gzipped. Can also be included as an AMD module



CSS3 Compliant

Supports CSS3 selectors to find elements as well as in style property manipulation



Cross-Browser

[Chrome](#), [Edge](#), [Firefox](#), [IE](#), [Safari](#), [Android](#), [iOS](#), and more



Download jQuery

v3.1.1

[View Source on GitHub →](#)
[How jQuery Works →](#)

What is jQuery?

jQuery is a fast, small, and feature-rich JavaScript library. It makes things like HTML document traversal and manipulation, event handling, animation, and Ajax much simpler with an easy-to-use API that works across a multitude of browsers. With a combination of versatility and extensibility, jQuery has changed the way that millions of people write JavaScript.

Corporate Members



Resources

- [jQuery Core API Documentation](#)
- [jQuery Learning Center](#)
- [jQuery Blog](#)
- [Contribute to jQuery](#)
- [About the jQuery Foundation](#)
- [Browse or Submit jQuery Bugs](#)



Learn jQuery in your browser!

Support from our corporate members makes it possible for the jQuery Foundation to continue our work on our JavaScript libraries and pushing the open web forward with events and participation in the standards process. [View](#)

Подключение

- `<head>`
`<script type="text/javascript"`
`src="jquery-X.X.X.min.js"></script>`
`</head>`
- `<head>`
`<script type="text/javascript"`
`src="http://code.jquery.com/jquery-`
`latest.js"></script>`
`</head>`

API



- jQuery Core
- Selectors
- Attributes
- Traversing
- Manipulation
- CSS
- Events
- Effects
- Ajax
- Utilities

<http://api.jquery.com/>

Назначение jQuery

"WRITE LESS, DO MORE"

Сравнение jQuery и JS

Javascript

```
document  
.getElementById ("div-3")  
.innerHTML = "Новый текст";
```

jQuery

```
$("#div-3")  
.html("Новый текст");
```

Суть jQuery

1. Выбираем элементы HTML документа
2. Проводим операции над выбранными

```
<head>
```

```
<script type="text/javascript" src="jquery.js"></script>
```

— подключаем jQuery

```
<script type="text/javascript">
```

Событие "ready" (функция будет выполнена, когда DOM будет готов)

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

```
$(".button").click(function(){  
    $("#panel").slideDown("slow");  
});
```

```
});
```

```
</script>
```

```
</head>
```

Эту часть можно вынести во внешний файл

К чему Вы хотите привязать Вашу функцию? Это может быть class, ID, selector (DIV, H1, P...)

Эта функция будет вызвана по событию "click" на элементе с классом "button"

Что же будет происходить с #panel? Элемент будет медленно опускаться вниз

Чем мы будем оперировать? Элементом с ID = panel

```
$("#panel")
```

Для кватирования могут использоваться как одинарные так и двойные кавычки: `(".class") == ('.class')`

Vanilla JS DOM Methods

There are 6 main ways of querying DOM:

- `getElementById`
- `getElementsByTagName`
- `getElementsByName`
- `getElementsByClassName` (except IE<9)
- `querySelector` (except IE<8 and IE8 in compat mode)
- `querySelectorAll`

Навигация по дереву JS

```
<body>
  <ol id=ordlist>
    <li>Первый</li>
    <li>Второй</li>
    <li>Третий</li>
  </ol>
  <script>
    )
  </script>
</body>
```

Навигация по дереву jQuery

<body>

<ol id="ordlist">

Первый

Второй

Третий

<script>

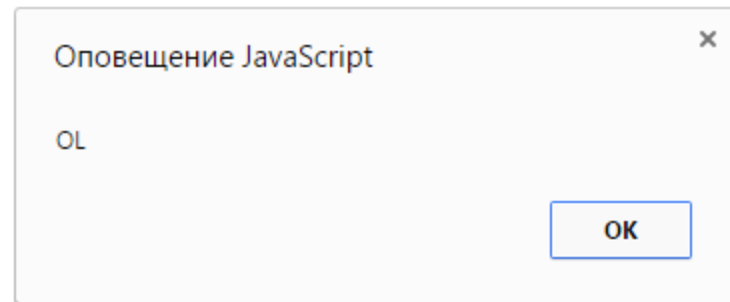
alert(document.getElementById('ordlist').tagName

alert(\$('#ordlist').prop("tagName"));

</script>

</body>

1. Первый
2. Второй
3. Третий
4. Четвертый





jQuery Tutorial

[jQuery HOME](#)[jQuery Intro](#)[jQuery Get Started](#)[jQuery Syntax](#)[jQuery Selectors](#)[jQuery Events](#)

jQuery Effects

[jQuery Hide/Show](#)[jQuery Fade](#)[jQuery Slide](#)[jQuery Animate](#)[jQuery stop\(\)](#)[jQuery Callback](#)[jQuery Chaining](#)

jQuery HTML

[jQuery Get](#)[jQuery Set](#)[jQuery Add](#)[jQuery Remove](#)[jQuery CSS Classes](#)

"Try it Yourself" Examples in Each Chapter

With our online editor, you can edit the code, and click on a button to view the result.

Example

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

[Try it Yourself »](#)

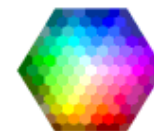
Click on the "Try it Yourself" button to see how it works.

jQuery Examples

Learn by examples! At W3Schools you will find a lot of jQuery examples to edit and test yourself.

- [jQuery Examples](#)

COLOR
PICKER



LEARN
MORE

Tabs
Dropdowns



find packages

sign up or log in 

packages with keyword 'jquery-plugin'

**es-cookies**

Module-friendly fork of [js-cookie](ht...

2.1.5 published 5 hours ago by zeekay

**restintag**

HTTP requests made easy through HT...

0.2.3 published 6 hours ago by khaledela..

**ukey1-jquery-sdk**

Ukey1 SDK plugin for jQuery

1.0.0 published 8 hours ago by asaritech

**jquery-gmap**

jQuery Wrapper for Google Maps API v...

2.1.1 published 14 hours ago by graubnla

**jquery-validation**

Client-side form validation made easy

1.16.0 published 3 months ago by staabm

**jquery-datetimepicker**

jQuery Plugin DatePicker it is Da...

2.5.4 published 10 months ago by xdan

**urijs**

URI.js is a Javascript library for worki...

1.18.9 published 18 hours ago by rodne...

**jquery-goup**

A simple jQuery plugin that let users g...

1.1.0 published 10 months ago by ryuk87

**gijgo-modular**

Gijgo is a set of free open source javas...

1.2.0 published yesterday by atatanasov

**gijgo-combined**

Gijgo is a set of free open source javas...

1.2.0 published yesterday by atatanasov

**gijgo**

Gijgo is a set of free open source javas...

1.2.0 published yesterday by atatanasov

**jquery-grid**

jQuery Grid by Gijgo.com is a plug-in f...

1.2.0 published yesterday by atatanasov

**jquery-load-json**

jQuery plugin that enables developer...

**jquery-confirm**

A jQuery plugin that provides great se...

**tooltipster**

A flexible and extensible jQuery plugi...

<https://www.npmjs.com/browse/keyword/jquery-plugin>

[Plugins](#)[Contribute](#)[Events](#)[Support](#)[jQuery Foundation](#)

The jQuery Plugin Registry

Popular Tags

- [ui](#) (542)
- [jquery](#) (482)
- [form](#) (285)
- [animation](#) (273)
- [input](#) (252)

The jQuery Plugin Registry is in read-only mode. New plugin releases will not be processed.

We recommend moving to npm, using `"jquery-plugin"` as the keyword in your package.json. The npm blog has [instructions for publishing your plugin to npm](#).

<https://plugins.jquery.com/>

- Ajax
 - Глобальные Ajax события
 - Вспомогательные методы
 - Низкоуровневый интерфейс
 - Сокращенные методы
- Атрибуты
- Объекты обратного вызова
- Ядро
- CSS
- Данные
- Объекты отложенного выполнения
- Устаревшие методы
 - Версия 1.3

Русская документация по API jQuery

jQuery это быстрая, небольшая и богатая возможностями JavaScript библиотека. Она позволяет очень просто делать такие вещи как: обход элементов или манипуляция элементами HTML документа или AJAX запросы. Не думая о кросс-браузерности - это и многое другое берет на себя jQuery.

.add()

Перемещения > Обход

Добавляет заданные элементы в уже существующий набор элементов.

.addBack()

Перемещения > Обход

Функция объединяет предыдущую выборку элементов с текущей. Это часто необходимо в тех случаях, когда Вам нужно осуществить над этими элементами общее действие.

.addClass()

Атрибуты | Манипуляция DOM > Class атрибут | CSS

Добавляет класс(ы) выбранным элементам страницы. Если из этих элементов некоторые уже принадлежат каким-то классам, то новый (новые) класс добавится к

JQUERY CORE

Функция jQuery(...)

- Сокращенный вариант вызова `$(...)`
- Функция для доступа к возможностям библиотеки
- Функция вызывается всеми методами jQuery
- Функция возвращает список объектов, над которым определены все доступные jQuery-функции
- Позволяет работать с любыми объектами (существующими на странице и созданными динамически) так, как будто это одни и те же элементы, уже существующие на странице.
 - `$(html)`
 - `$(elements)`
 - `$(expression[, context])`
 - `$(callback)`

Выбор элемента по id

- `<head>`
`<script type="text/javascript" src="jquery-1.9.1.min.js"></script>`
`<script>`
`function change() {`
 `$('#paragraph').toggleClass('red');`
`}`
`</script>`
`<style>`
`.red{background-color:#ff0000}`
`</style>`
`</head>`
- `<body>`
`<p id=paragraph onclick=change()>Текст</p>`
`</body>`

`$(html)`

- Позволяет создать html-элементы «на лету» из «чистого» HTML.
- Например, можно создать элемент `div`, содержащий абзац с текстом «Append to» и добавить его к элементу с `id="body"`

```
my_div = $("<div><p>Append to</p></div>");  
my_div.appendTo("#body");
```
- Или

```
 $("<div><p>Append to</p></div>").appendTo("#body");
```
- Элемент до отработки скрипта:

```
<div id="body"></div>
```
- Элемент после отработки скрипта:

```
<div id="body"><div><p>Append to</p></div></div>
```

`$(elements)`

- Позволяет добавить функциональность jQuery к уже существующим элементам страницы (к элементам из объектной модели документа)

`$(elements)`

- `<html><head>Подключение
jQuery</head><body>`
- `<form id="myForm">
<input type="text" value="HTML">
</form>`
- `<script type="text/javascript">
$(document.body).css("background-color",
"yellow");
$(myForm.elements).hide();
</script>`
- `</body></html>`

`$(expression[, context])`

- Первый параметр – выражение, которое позволит jQuery найти элемент на странице
Второй параметр определяет область поиска (по умолчанию поиск выполняется на всей странице)
- `$("script");` поиск всех элементов script
- `$("ul > li");` поиск li, входящих в ul
- `$("input:checkbox", document.myForm);`
поиск checkbox, входящих в первую форму
- `$("#myForm");` элементы с id=myForm
- `$("#myForm > p.red");`
- ...



`$(callback)` или `$(document).ready()`

- JavaScript часто использует событие `window.onload`
- Событие вызывается после того, как документ полностью загрузился в окно браузера
- Вызываемый в обработчике этого события код не сработает, пока не загрузятся все изображения и т.п. на странице
- jQuery позволяет выполнить код в тот момент, когда доступна объектная модель документа (полностью загружен исходный код страницы, но, возможно, не загружено мультимедийное содержимое)
- Для этого используется метод `.ready()`:

`$(document).ready()`

- `$(document).ready`
(
 `function()`
 `{ /* ... */ }`
);
- Или `$(fn)`
`$(`
 `function()`
 `{ /* ... */ }`
);
- В качестве аргумента в `.ready()` передается функция, которая будет выполнять необходимые действия

`$(document).ready()`

- Подключение jQuery
- ```
<script>
$(document).ready(
 function()
 {
 $("a").append(" (ссылка на сайт
университета)");
 }
);
</script>
```
- ```
</head>
<body>
<ul>
<li><a href="www.isu.ru">ИГУ</a>
<li><a href="www.istu.edu">ИРНИТУ</a>
</ul>
</body>
```

`$(document).ready()`

```
<style> .red {color:red} </style>
<script>
$(document).ready(init);
function init() {
  $("p.red").css("background-color", "#00ff00");
  alert('Прочие действия');
}
</script>
</head>
<body>
<ul>
<li><p class=red>НГТУ</p>
<li><p>НГУ</p>
<li><p class=red> <a href="www.isu.ru">ИГУ</a></p>
<li><p><a href="www.istu.edu">ИРНИТУ</a></p>
</ul>
</body>
```

Effects (основные)

- `show( )`
Отображает скрытые элементы
- `show( speed, callback )`
Отображает скрытые элементы с анимацией
- `hide( )`
Скрывает элементы
- `hide( speed, callback )`
Скрывает элементы с анимацией
- `toggle( )`
Переключает соответствующие элементы
- `toggle( switch )`
Переключает соответствующие элементы с указанием нового значения
- `toggle( speed, callback )`
Переключает соответствующие элементы с анимацией

Effects (скольжения)

- `slideDown( speed, callback )`
Раскрывает элемент с эффектом скольжения
- `slideUp( speed, callback )`
Закрывает элемент с эффектом скольжения.
- `slideToggle( speed, callback )`
Переключает видимость элементов с эффектом скольжения

Effects (растворения)

- `fadeIn( speed, callback )`
Изменяет прозрачность элемента
- `fadeOut( speed, callback )`
Изменяет прозрачность элемента до 0
- `fadeTo( speed, opacity, callback )`
Изменяет прозрачность элемента до указанного значения

Effects (анимации)

- Замена технологии Flash
- `animate( params, duration, easing, callback )`
Создает настраиваемую анимацию для визуальных эффектов
 - `params` – свойства, которые участвуют в анимации в виде пар {ключ: значение}
Например: {height: “show”} или {opacity: 50, width: 100, height: 200}.
 - `duration` – продолжительность в миллисекундах.
 - `easing` – замедление анимации (замедляется к концу, “easein” или ускоряется, “easeout”)
 - `callback` – функция, которая будет вызвана после завершения анимации

- `<html>`
`<head>Подключение jQuery </head>`
- `<body>`
`<div id=mydiv style="background-color:magenta; height:300; width:300"></div>`
- `<script>`
`$("#mydiv")`
`.animate({height: "hide"}, "slow")`
`.animate({height: "show"}, "slow");`
`</script>`
`</body>`
`</html>`

Interactions

- Draggable
- Droppable
- Resizable
- Selectable
- Sortable

Widgets

- Accordion
- Autocomplete
- Button
- Checkboxradio
- Controlgroup
- Datepicker
- Dialog
- Menu
- Progressbar

jQuery UI Demos

jQuery UI offers a combination of interaction, effects, widgets, utilities, and themes designed to work well together or on their own. Play with the demos, view the source, [build a theme](#), read the [API documentation](#) and start using jQuery UI today.

Interactions

Interactions add basic mouse-based behaviors to any element. You can create sortable lists, resizable elements, drag & drop behaviors and more with just a few lines of code. Interactions also make great building blocks for more complex widgets and applications.

- [Draggable](#)
- [Droppable](#)
- [Resizable](#)
- [Selectable](#)
- [Sortable](#)

Widgets

Widgets are full-featured UI controls that bring the richness of desktop applications to the Web. All widgets provide a solid core with plenty of extension points for customizing behavior, as well as full theming support.

- [Accordion](#)

<http://jqueryui.com/demos/>

Примеры ex1.html

- ```
<script type="text/javascript">
$(document).ready(
function()
{
 $(".btn-slide").click(function()
 {
 $("#panel").slideToggle("slow");
 });
});
</script>
```
- После щелчка на ссылке панель с id=«panel» выдвигается/прячется

# Примеры ex2.html

---

- ```
<script type="text/javascript">
$(document).ready(function()
{
    $(".pane .delete").click(function(){
        $(this).parents(".pane").animate({ opacity:
'hide' }, "slow");
    });
});
</script>
```
- После щелчка на изображении `<img class=«delete»>`, находится родительский элемент `<div class=«pane»>` и его прозрачность будет медленно изменяться от `opacity= 1.0` до `opacity=hide`

Примеры ex3.html

- ```
<script type="text/javascript">
$(document).ready(function(){
 $(".run").click(function(){
 $("#box").animate({opacity: "0.1", left: "+=400"}, 1200)
 .animate({opacity: "0.4", top: "+=160", height: "20", width: "20"}, "slow")
 .animate({opacity: "1", left: "0", height: "100", width: "100"}, "slow")
 .animate({top: "0"}, "fast")
 .slideUp()
 .slideDown("slow")
 return false;
 });});
</script>
```

# Примеры ex4.html

---

- ```
<script type="text/javascript">
$(document).ready(function(){
    $("h2").append('<em></em>')
    $(".thumbs a").click(function(){
        var largePath = $(this).attr("href");
        var largeAlt = $(this).attr("title");
        $("#largeimg").attr({ src: largePath, alt: largeAlt
    });
    $("h2 em").html(" (" + largeAlt + ")"); return false;
    });
});
</script>
```


Примеры ex5.html

- Добавляется тег `<em>` в заголовки `<h2>`. По щелчку на изображения с `<p class=thumbs>` выполняются следующие действия:
 - значение атрибута «href» сохраняется в переменной «largePath»
 - значение атрибута «title» сохраняется в переменной «largeAlt»
 - в элементе `<img id=«largeImg»>` заменяется значение атрибута «src» и «alt» значениями из переменных «largePath» и «largeAlt»
 - элементу присваивается «h2 em» значение из «largeAlt»