

jQuery

write less, do more

Что такое jQuery?

jQuery библиотека JavaScript, фокусируется на взаимодействии JavaScript и HTML.

Библиотека jQuery помогает легко получать доступ к любому элементу DOM, обращаться к атрибутам и содержимому элементов DOM, манипулировать ими. Также библиотека jQuery предоставляет удобный API по работе с Ajax.

Цели и задачи jQuery

Write less, do more.

jQuery – нужен для того, чтобы облегчить работу веб-программиста. С помощью этой библиотеки проще повысить функциональность страниц с минимальным выполнением монотонной работы.

История



jQuery был создан Джоном Реззином в 2004 году

Библиотека была представлена общественности на компьютерной конференции «BarCamp» в Нью-Йорке в 2006 году.

ВОЗМОЖНОСТИ

- Кросс-браузерный выбор DOM объектов
- Перемещение и модификация DOM
- События (Events)
- Работа с CSS
- Визуальные эффекты и анимация
- AJAX-дополнения
- Расширяемость плагинами включая jQuery UI

Подключение

jQuery, как правило, включается в веб-страницу как один внешний JavaScript-файл:

```
<head>  
  <script src="jquery-3.2.1.min.js">  
  </script>  
</head>
```

<https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js>

Вся работа с jQuery ведётся с помощью функции \$. Если \$ уже использоваться для своих нужд, то можно использовать её синоним jQuery

ИСПОЛЬЗОВАНИЕ

`$(selector).action()`

\$ - знак доступа к jQuery

(*selector*) - найти HTML элемент по селектору

action() - действие над этим элементом

Селекторы

- id

. - class

`$(".test").hide()` - спрятать элемент с `class="test"`.

Селекторы

Пример	Результат
<code>\$("p")</code>	Будут выбраны все элементы <code>p</code> , которые находятся на странице.
<code>\$(".par")</code>	Будут выбраны все элементы на странице с <code>class="par"</code> .
<code>\$("#par")</code>	Будет выбран первый элемент на странице с <code>id="par"</code> .
<code>\$(this)</code>	Позволяет выбрать текущий HTML элемент. Щелкните на <code>S(this)</code> слева, чтобы посмотреть пример использования данного селектора в онлайн редакторе.
<code>\$("p.par")</code>	Будут выбраны все элементы <code>p</code> на странице с <code>class="par"</code> .
<code>\$("p#par")</code>	Будут выбраны все элементы <code>p</code> на странице с <code>id="par"</code> .
<code>\$(".par,.header,#heat")</code>	Будут выбраны все элементы на странице со значениями атрибутов <code>class="par"</code> , <code>class="header"</code> и <code>id='heat'</code> .
<code>\$("[src]")</code>	Будут выбраны все элементы на странице, имеющие атрибут <code>src</code> .
<code>\$("[src='значение']")</code>	Будут выбраны все элементы со значениям атрибута <code>src="значение"</code> .
<code>\$("[src\$='.gif']")</code>	Будут выбраны все элементы со значениями атрибута <code>src</code> заканчивающимися на <code>.gif</code> .
<code>\$("div#wrap .par1")</code>	Будут выбраны все элементы с <code>class=par1</code> , которые находятся внутри элементов <code>div</code> с <code>id=wrap</code> .
<code>\$(":input")</code>	Будут выбраны все <code>input</code> элементы на странице.
<code>\$(":тип")</code>	Будут выбраны все <code>input</code> элементы с <code><input type='тип' /></code> . Например <code>:button</code> выберет все <code><input type="button" /></code> элементы, <code>:text</code> выберет все <code><input type="text" /></code> элементы и т.д.

События

```
$( "p" ).click( function () {  
    // action goes here!!  
} );
```

Mouse Events	Keyboard Events	Form Events	Document/Window Events
click	keypress	submit	load
dblclick	keydown	change	resize
mouseenter	keyup	focus	scroll
mouseleave		blur	unload

DOM иерархия



- **parent**
- **ancestor**
- **child**
- **descendant**
- **sibling**