



Что же такое jQuery?

- jQuery очень легкая библиотека Javascript (некоторые называют ее фреймворком), которая избавляет от головной боли при написании Javascript кода. У нее много очень мощных возможностей, как например: отслеживание DOM, добавление красивых эффектов и анимаций к элементам, супер простые Ajax техники и методы. На главной странице сайта jQuery наиболее точное, на мой взгляд, описание:
- jQuery быстрая и лаконичная библиотека, которая упрощает обработку событий, анимацию и взаимодействие с Ajax для более быстрой веб разработки. jQuery разработан для того, чтобы изменить методы написания JavaScript кода





Преимущества jQuery?

- существенно уменьшается количество кода (необходимого для работы скрипта) по сравнению с JavaScript, что в свою очередь означает меньше временных затрат и более читабельный код. Далее в статье будут рассмотрены некоторые примеры.+
- намного проще понять код (в отличии от JavaScript).
- очень удобная документация
- использование Ajax становится намного проще
- огромное количество плагинов, с помощью которых можно сделать практически все, что угодно.





Как начать?

- Сперва необходимо подключить файл с библиотекой jquery.

после загрузки необходимо загрузить этот файл к себе на хостинг, и в шапке доку

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

```
<script src="https://ajax.googleapis.com/ajax/libs/jquery/2.2.4/jquery.min.js"></script>
```





Готов ли документ?

- Для выполнения нашего первого jQuery скрипта, нам необходимо поместить весь наш скрипт в функцию. Эта функция будет выполнена при полной готовности DOM (когда "документ будет готов" - дословный перевод с англ

```
$(document).ready(function() {  
    //Code here  
});
```

```
$(function(){  
    //Code here  
});
```





Как получить элемент ?

- Для того чтобы понимать как работает селектор Вам все-же необходимы базовые знания CSS, т.к. именно от принципов CSS отталкивается селектор jQuery

```
1 | $('#sidebar'); // выбор элемента с id = sidebar
2 | $('.post'); // выбор элементов с class = post
3 | $('div#sidebar'); // выбор элемента div с id = sidebar
4 | $('div.post'); // выбор элементов div с class = post
```

```
$('#div, span'); // выбор всех div и span элементов
```





Как получить элемент ?



```
$('span + img');           // выбор всех img элементов перед которыми  
                             // идут span элементы  
$('span ~ img');           // выбор всех img элементов после  
                             // первого элемента span  
$('#banner').prev();        // выбор предыдущего элемента от найденного  
$('#banner').next();        // выбор следующего элемента от найденного
```





Фильтры

```
$('div:first'); // выбираем первый div в доме
$('div:last'); // выбираем последний div в доме
$('div:not(.red)'); // выбираем div'ы у которых нету класса red
$('div:even'); // выбираем четные div'ы
$('div:odd'); // выбираем нечетные div'ы
$('div:eq(N)'); // выбираем div идущим под номером N в DOME
$('div:gt(N)'); // выбираем div'ы, индекс которых больше чем N в DOME
$('div:lt(N)'); // выбираем div'ы, индекс которых меньше чем N в DOME
$(':header'); // выбо заголовков h1, h2, h3 и т.д.
$('div:animated'); // выбор элементов с активной анимацией
```





Фильтры

```
$( ":text" );           // выбор всех input элементов с типом =text  
$( ":radio" );          // выбор всех input элементов с типом =radio  
                        // и так далее  
$( "input:enabled" );   // выбор всех включенных элементов input  
$( "input:checked" );   // выбор всех отмеченных чекбоксов
```





События

- **change** — изменение значения элемента (значение, при потере фокуса, элемента отличается от изначального, при получении фокуса)
- **click** — клик по элементу (порядок событий — mousedown, mouseup, click)
- **dblclick** — двойной клик по элементу
- **resize** — изменение размеров элементов
- **scroll** — скроллинг элемента
- **select** — выбор текста (актуален только для `input[type=text]` и `textarea`)





События

- **submit** — отправка формы
- **focus** — фокус на элементе – актуально для `input[type=text]`, но в современных браузерах работает и с другими элементами
- **blur** — фокус ушел с элемента – актуально для `input[type=text]` — срабатывает при клике по другому элементу на странице или по событию клавиатуры (к примеру переключение по `tab`'у)
- **focusin** — фокус на элементе, данное событие срабатывает на предке элемента, для которого произошло событие `focus`





События

focusout — фокус ушел с элемента, данное событие срабатывает на предке элемента, для которого произошло событие blur

keydown — нажатие клавиши на клавиатуре

keypress — нажатие клавиши на клавиатуре (порядок событий — keydown, keypress, keyup)

keyup — отжатие клавиши на клавиатуре

load — загрузка элемента (img)

unload — выгрузка элемента (window)

mousedown — нажатие клавиши мыши





События

mouseup — отжатие клавиши мыши

mousemove — движение курсора

mouseenter — наведение курсора на элемент, не срабатывает при переходе фокуса на дочерние элементы

mouseleave — вывод курсора из элемента, не срабатывает при переходе фокуса на дочерние элементы

mouseover — наведение курсора на элемент

mouseout — вывод курсора из элемента





Что такое AJAX?

AJAX — это аббревиатура, которая означает Asynchronous Javascript and XML. На самом деле, AJAX не является новой технологией, так как и Javascript, и XML существуют уже довольно продолжительное время, а AJAX — это синтез обозначенных технологий. AJAX чаще всего ассоциируется с термином Web 2.0 и преподносится как новейшее Web-приложение.





Возможности AJAX

- Возможность создания удобного Web-интерфейса
- Активное взаимодействие с пользователем
- Частичная перезагрузка страницы, вместо полной
- Удобство использования

