jQuery is a fast and concise JavaScript Library created by John Resig in 2006 with a nice motto − **Write less, do more**.

jQuery simplifies HTML document traversing, event handling, animating, and Ajax interactions for rapid web development.

jQuery is a JavaScript toolkit designed to simplify various tasks by writing less code. Here is the list of important core features supported by jQuery −

* **DOM manipulation** − The jQuery made it easy to select DOM elements, traverse them and modifying their content by using cross-browser open source selector engine called **Sizzle**.
* **Event handling** − The jQuery offers an elegant way to capture a wide variety of events, such as a user clicking on a link, without the need to clutter the HTML code itself with event handlers.
* **AJAX Support** − The jQuery helps you a lot to develop a responsive and feature-rich site using AJAX technology.
* **Animations** − The jQuery comes with plenty of built-in animation effects which you can use in your websites.
* **Lightweight** − The jQuery is very lightweight library - about 19KB in size ( Minified and gzipped ).
* **Cross Browser Support** − The jQuery has cross-browser support, and works well in IE 6.0+, FF 2.0+, Safari 3.0+, Chrome and Opera 9.0+
* **Latest Technology** − The jQuery supports CSS3 selectors and basic XPath syntax.

**What is jQuery**

jQuery is a fast, lightweight, and feature-rich JavaScript library that is based on the priciple *"write less, do more"*. It's easy-to-use APIs makes the things like HTML document traversal and manipulation, event handling, adding animation effects to a web page much simpler that works seamlessly across all the major browsers. jQuery also gives you the ability to create an Ajax based application in a quick and simple way.

Companies like Google, Microsoft and IBM are using the jQuery for their applications. So you can easily understand how popular the jQuery is?

**Advantages of jQuery**

The biggest advantage of jQuery comes from its selectors that allow you to traverse the DOM tree of an HTML document's structure in an efficient manner.

Additionally, using the jQuery inbuilt methods you can create animations and effects like sliding transition, showing or hiding an element, etc. with a single line of code.

Here are some more advantages, why one should opt for jQuery:

* **Save lots of time** — You can save lots of time and efforts by using the jQuery inbuilt effects and selectors and concentrate on other development work.
* **Simplify common JavaScript tasks** — jQuery considerably simplifies the common JavaScript tasks. Now you can easily create feature rich and interactive web pages with fewer lines of codes, a typical example is retrieving the information from a server and updates the page without refreshing.
* **Easy to use** — jQuery is very easy to use. Anybody with the basic working knowledge of HTML, CSS and JavaScript can start development with jQuery.
* **Compatible with browsers** — jQuery is created with modern browsers in mind and it is compatible with all major modern browsers such as Mozilla Firefox, Google Chrome, Safari, Internet Explorer, and Opera.
* **Absolutely Free** — And the best part is, it is completely free to download and use.

## Standard jQuery Syntax

A jQuery statement typically starts with the dollar sign ($) and ends with a semicolon (;).

In jQuery, the dollar sign ($) is just an alias for jQuery. Let's consider the following example code which demonstrates the most basic statement of the jQuery.

* <script type="text/javascript">
* $(document).ready(function(){
* // Some code to be executed...
* alert("Hello World!");
* });
* </script>

**Explanation of code**

If you are completely new to the jQuery, you might think what that code was all about. OK, let's go through each of the parts of this script one by one.

* The <script> element — Since jQuery is just a JavaScript library, so the jQuery code can be placed inside the [<script>](https://www.tutorialrepublic.com/html-reference/html-script-tag.php) element. However, if you want to place it in an [external JavaScript file](https://www.tutorialrepublic.com/html-tutorial/html-scripts.php), which is preferred, you just remove this part.
* The $(document).ready(*handler*); — This statement is typically known as ready event. Where the *handler* is basically a function that is passed to the ready() method to be executed safely as soon as the document is ready to be manipulated i.e. when the DOM hierarchy has been fully constructed.

**Tag Name**

Represents a tag name available in the DOM. For example **$('p')**selects all paragraphs <p> in the document.

**Tag ID**

Represents a tag available with the given ID in the DOM. For example **$('#some-id')** selects the single element in the document that has an ID of some-id

**Tag Class**

Represents a tag available with the given class in the DOM. For example **$('.some-class')** selects all elements in the document that have a class of some-class.

## What are Events

Events are often triggered by the user's interaction with the web page, such as when a link or button is clicked, text is entered into an input box or textarea, selection is made in a select box, key is pressed on the keyboard, the mouse pointer is moved etc. In some cases, the Browser itself can trigger the events, such as the page load and unload events.

jQuery enhances the basic event-handling mechanisms by offering the events methods for most native browser events, some of these methods are ready(), click(), keypress(), focus(), blur(), change(), etc. For example, to execute some JavaScript code when the DOM is ready, you can use the jQuery ready() method, like this:

1. <script type="text/javascript">
2. $(document).ready(function(){
3. // Code to be executed
4. alert("Hello World!");
5. });
6. </script>

## Mouse Events

A mouse event is fired when the user click some element, move the mouse pointer etc. Here're some commonly used jQuery methods to handle the mouse events.

## The click() Method

The jQuery click() method attach an event handler function to the selected elements for "click" event. The attached function is executed when the user clicks on that element. The following example will hide the [<p>](https://www.tutorialrepublic.com/html-reference/html-p-tag.php) elements on a page when they are clicked.

1. <script type="text/javascript">
2. $(document).ready(function(){
3. $("p").click(function(){
4. $(this).slideUp();
5. });
6. });
7. </script>

## The dblclick() Method

The jQuery dblclick() method attach an event handler function to the selected elements for "dblclick" event. The attached function is executed when the user double-clicks on that element. The following example will hide the <p> elements when they are double-clicked.

1. <script type="text/javascript">
2. $(document).ready(function(){
3. $("p").dblclick(function(){
4. $(this).slideUp();
5. });
6. });
7. </script>

## The hover() Method

The jQuery hover() method attach one or two event handler functions to the selected elements that is executed when the mouse pointer enters and leaves the elements. The first function is executed when the user place the mouse pointer over an element, whereas the second function is executed when the user removes the mouse pointer from that element.

The following example will highlight <p> elements when you place the cursor on it, the highlighting will be removed when you remove the cursor.

1. <script type="text/javascript">
2. $(document).ready(function(){
3. $("p").hover(function(){
4. $(this).addClass("highlight");
5. }, function(){
6. $(this).removeClass("highlight");
7. });
8. });
9. </script>

## The mouseenter() Method

The jQuery mouseenter() method attach an event handler function to the selected elements that is executed when the mouse enters an element. The following example will add the class highlight to the <p> element when you place the cursor on it.

1. <script type="text/javascript">
2. $(document).ready(function(){
3. $("p").mouseenter(function(){
4. $(this).addClass("highlight");
5. });
6. });
7. </script>