jQuery

write less, do more.

Introduction to jQuery

jQuery is a lightweight, "write less, do more", JavaScript library.

The purpose of jQuery is to make it much easier to use JavaScript on your website.

jQuery takes a lot of common tasks that require many lines of JavaScript code to accomplish, and wraps them into methods that you can call with a single line of code.

jQuery also simplifies a lot of the complicated things from JavaScript, like AJAX calls and DOM manipulation.

Introduction to jQuery(Contd.)

Features Of jQuery,

DOM manipulation: The jQuery made it easy to select DOM elements, negotiate 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.

Introduction to jQuery(Contd.)

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 zipped).
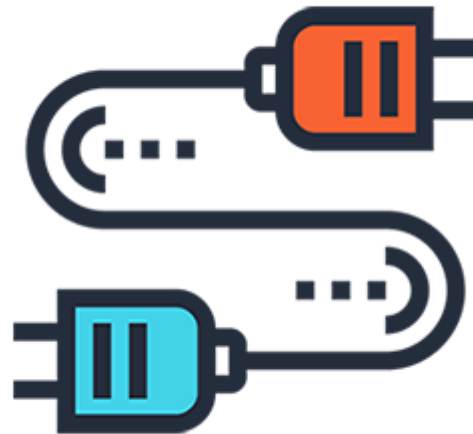
Cross Browser Support: The jQuery has cross-browser support, and works well in IE 6.0+, FF 2.0+, Safari 3.0+, Chrome and Opera 9.0+

Latest Technology: The jQuery supports CSS3 selectors and basic XPath syntax.

List Of jQuery Plugins

A plug-in is a software component that adds a specific feature to an existing computer program.

Plugins are the section of code and these codes are written in a JavaScript file. These JavaScript files are used to provide jQuery methods that work together with jQuery library methods.

List Of jQuery Plugins(Contd.)

Complete list of jQuery plugins,

Plugin	Description
Page Piling	This feature provides Pile your sections one over another and access them by scrolling or by anchor links.
Flickerplate	They are responsible for touch detections and events such as custom Modernizr and jQuery Finger library.
Multiscroll	Create split web pages along with divided multiple vertical scrolling panels.
RowGrid	It is very lightweight and supports an infinite scrolling feature.

List Of jQuery Plugins(Contd.)

Plugin	Description
Alertify	Provides a pre-designed customizable notification system along with interactive browser dialogs.
Image ProgressBars	Here is a jquery plugin that turns photos into vertical progress bars.
DrawSVG	SVG images in a variety of ways for a developer's web pages, which is very lightweight and easy to use.
Tagsort	Displaying tags or filter elements based on different tags in a DOM.
Logos Distort	Animating a parallax environment for 3D scenes in the user browser.

List Of jQuery Plugins(Contd.)

Plugin	Description
Filer	Filer uploader plugin for easy implementation of uploading the files
Product Tour	Generating mobile-friendly customizable step-by-step guided tours for web applications.
menu-aim	Develop effective dropdown menus along with easy navigation of sub-menu contents.
Flipping Gallery	Flip many images in a gallery in various directions with the autoplay feature
table2excel	To export HTML tables to excel (.xls) files.
Vertical Dot Navigation	That help in navigating or scrolling through long web pages.

List Of jQuery Plugins(Contd.)

Plugin	Description
CLNDR	It is used in appointing dates of various events in the calendar format.
GMaps	Is used Google maps in a variety of ways.
Highcharts	Implement charts for representing or comparing data for web and mobile applications.
Jcrop	The quickest and easiest approach to include picture cropping capability in
hc-offcanvas-nav	Create toggled off-canvas navigation with flexible multi-level menu and sub-menu items.

List Of jQuery Plugins(Contd.)

Plugin	Description
Tooltip	Good responsive placement (try resizing your browser window on the demo website!).
bootstrapSelect	Select elements into a bootstrap dropdown with colors and a search field.
Timepicker	Add the date picker section on webpages.
Poppa	Reliable, cross-browser friendly, and lightweight plugin that makes client-side validation very simple.

Steps to include jQuery in Web pages

There are currently two ideal solutions for embedding the jQuery library in a web page:

1. Use the Google-hosted content delivery network (CDN) to include a version of jQuery.

2. Download your own version of jQuery from jQuery.com and host it on your own server or local filesystem.

First method is ideal when connection to internet is reliable and unlimited even while development.

Second method is better for developing in an offline environment but not ideal for hosting compared to CDN.

Steps to include jQuery in Web pages(Contd.)

Include jQuery from CDN Link:

CDN stands for Content Delivery Network which is basically a set of servers used for storing and delivering data. Basically, these jQuery library files are already uploaded to various CDNs and we can use them directly on our web page. Then, we don't need to download any files on our local machine. By inserting following line in head tag completes the work,

```
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
```

Steps to include jQuery in Web pages(Contd.)

Download the jQuery library:

First, we will download the jQuery library files to our localhost from the jQuery Website. After downloading, we will add the downloaded files to our web page in this manner.

jQuery download link - https://jquery.com/download/

```
<script src="file_name_with_full_path"></script>
```

jQuery Syntax

The jQuery syntax is tailor-made for selecting HTML elements and performing some action on the element(s).

Basic syntax is: **$(selector).action()**

Where,

A **$** sign to define/access jQuery

A **(selector)** to "query (or find)" HTML elements

A jQuery **action()** to be performed on the element(s)

jQuery Syntax(Contd.)

Examples:

$(this).hide() - hides the current element.

$("p").hide() - hides all <p> elements.

$(".test").hide() - hides all elements with class="test".

$("#test").hide() - hides the element with id="test".

jQuery Selectors

jQuery selectors allow you to select and manipulate HTML element(s). jQuery selectors are used to "find" (or select) HTML elements based on their name, id, classes, types, attributes, values of attributes and much more. It's based on the existing CSS Selectors, and in addition, it has some own custom selectors.

All selectors in jQuery start with the dollar sign and parentheses: $(). Types of selectors,

      Element Selector

      Id Selector

      Class Selector

jQuery Selectors(Contd.)

Element Selector

The jQuery element selector selects elements based on the element name. You can select all <p> elements on a page like this: $("p")

Example

```
<!DOCTYPE html>
<html>
<head>
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.7.1/jquery.min.js">
    </script>
<script>
$(document).ready(function(){
    $("button").click(function(){
```

```
        $("p").hide();
    });
});
</script>
</head>
<body>
  <h2>This is a heading</h2>
  <p>This is a paragraph.</p>
  <p>This is another paragraph.</p>
  <button>Click me to hide paragraphs</button>
</body>
</html>
```

jQuery Selectors(Contd.)

#id Selector

The jQuery #id selector uses the id attribute of an HTML tag to find the specific element. An id should be unique within a page, so you should use the #id selector when you want to find a single, unique element.

To find an element with a specific id, write a hash character, followed by the id of the HTML element:$("#test")

```html
<!DOCTYPE html>
<html>
<head>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.7.1/jquery.min.js">
</script>
<script>
$(document).ready(function(){
  $("button").click(function(){
    $("#test").hide();
  });
});
</script>
</head>
<body>
  <h2>This is a heading</h2>
  <p>This is a paragraph.</p>
  <p id="test">This is another paragraph.</p>
  <button>Click me</button>
</body>
</html>
```

jQuery Selectors(Contd.)

.class Selector

The jQuery .class selector finds elements with a specific class. To find elements with a specific class, write a period character, followed by the name of the class: $(".test")

```html
<!DOCTYPE html>
<html>
<head>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.7.1/jquery.min.js">
</script>
<script>
$(document).ready(function(){
  $("button").click(function(){
    $(".test").hide();
    });
  });
</script>
</head>
<body>
    <h2 class="test">This is a heading</h2>
    <p class="test">This is a paragraph.</p>
    <p>This is another paragraph.</p>
    <button>Click me</button>
</body>
</html>
```

jQuery Selectors(Contd.)

Some more selectors,

Syntax	Description
$("*")	Selects all elements
$("p.intro")	Selects all <p> elements with class="intro"
$("p:first")	Selects the first <p> element
$("ul li:first")	Selects the first element of the first
$("ul li:first-child")	Selects the first element of every
$("[href]")	Selects all elements with an href attribute
$("a[target='_blank']")	Selects all <a> elements with a target attribute value equal to "_blank"
$(":button")	Selects all <button> elements and <input> elements of type="button"
$("tr:even")	Selects all even <tr> elements
$("tr:odd")	Selects all odd <tr> elements

The jQuery document ready event

jQuery provides the ready() method, which is a custom event handler that is typically bound to the DOM's document object. The ready() method is passed a single parameter, a function, that contains the JavaScript code that should be executed once the DOM is ready to be traversed and manipulated.

The following is a simple example of this event opening an alert() window once the DOM is ready but before the page is completely loaded:

The jQuery document ready event(Contd.)

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
<script type="text/JavaScript" src="http://ajax.googleapis.com/ajax/libs/jquery/1.3.2/jquery.min.js"></script>
<script type="text/JavaScript">
    $(document).ready(function(){              //DOM not loaded, must use ready event
        alert($('p').text());
    });
</script>
</head>
<body>
<p>The DOM is ready!</p>
</body>
</html>
```

The jQuery document ready event(Contd.)

The ready() event handler method is jQuery's replacement for using the JavaScript core window.onload event.

It can be used as many times as you like. When using this custom event, it's advisable that it be included in your web pages after the inclusion of stylesheet declarations and includes.

Doing this will ensure that all element properties are correctly defined before any jQuery code or JavaScript code will be executed by the ready() event.

The jQuery Event Handling methods

An event refers to the actions performed by the site visitor during their interactivity with the website (or webpage). There can be various types of events such as

> The user clicks on the button.

> The user moves the mouse pointer over an image.

> The user pressed any key from the keyboard, etc.

jQuery Syntax For Event Methods

In jQuery, most DOM events have an equivalent jQuery method. To assign a click event to all paragraphs on a page, you can do this:

$("p").click(action_function());

The jQuery Event Handling methods(Contd.)

Here are some common DOM events:

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

Explore https://www.geeksforgeeks.org/jquery-event-methods-complete-reference/ for more events

The jQuery Event Handling methods(Contd.)

```html
<!DOCTYPE html>
<html>
  <head>
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.7.1/jquery.min.js"></script>
    <script>
        $(document).ready(function(){
            $("#p1").mouseenter(function(){
                alert("You entered p1!");
            });
        });
    </script>
  </head>
  <body>
    <p id="p1">Enter this paragraph.</p>
  </body>
</html>
```
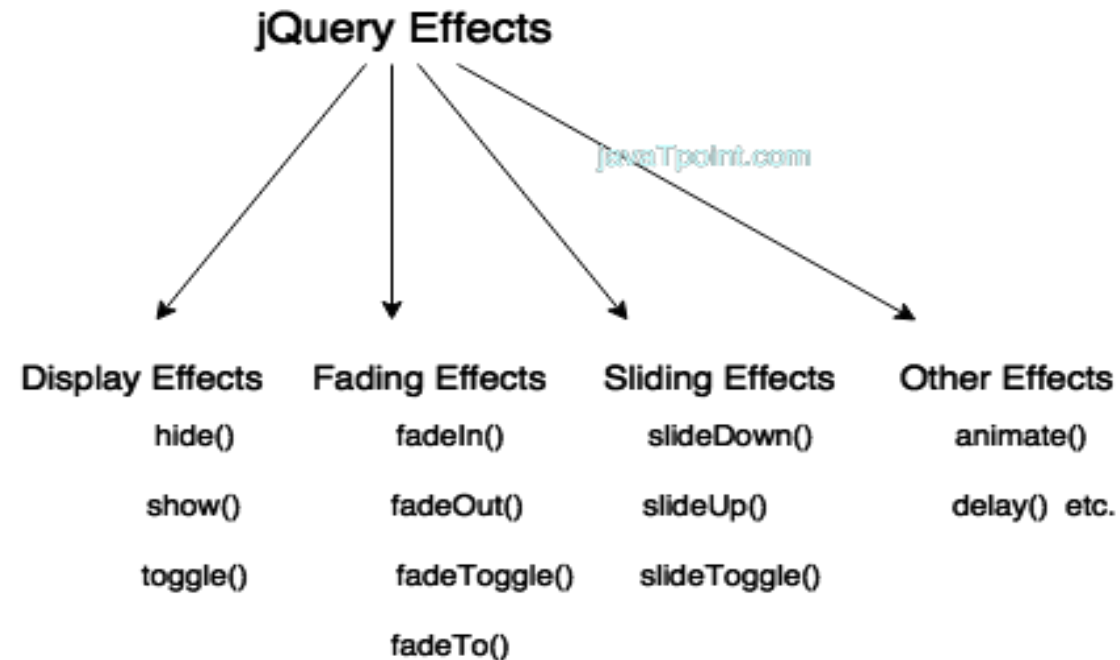
Effects of jQuery

jQuery enables us to add effects on a web page. jQuery effects can be categorized into fading, sliding, hiding/showing and animation effects.

Effects of jQuery(Contd.)

jQuery provides many methods for effects on a web page. A complete list of jQuery effect methods are given below:

Method	Description
animate()	performs animation.
clearQueue()	It is used to remove all remaining queued functions from the selected elements.
delay()	sets delay execution for all the queued functions on the selected elements.
dequeue()	It is used to remove the next function from the queue, and then execute the function.
fadein()	shows the matched elements by fading it to opaque. In other words, it fades in the selected elements.

Effects of jQuery(Contd.)

Method	Description
fadeout()	shows the matched elements by fading it to transparent. In other words, it fades out the selected elements.
fadeto()	adjusts opacity for the matched element. In other words, it fades in/out the selected elements.
fadetoggle()	shows or hides the matched element. In other words, toggles between the fadeIn() and fadeOut() methods.
finish()	It stops, removes and complete all queued animation for the selected elements.
hide()	hides the matched or selected elements.
queue()	shows or manipulates the queue of methods i.e. to be executed on the selected elements.

Effects of jQuery(Contd.)

Method	Description
show()	displays or shows the selected elements.
slidedown()	shows the matched elements with slide.
slidetoggle()	shows or hides the matched elements with slide. In other words, it is used to toggle between the slideUp() and slideDown() methods.
slideup()	hides the matched elements with slide.
stop()	stops the animation which is running on the matched elements.
toggle()	shows or hides the matched elements. In other words, it toggles between the hide() and show() methods.

Effects of jQuery(Contd.)

Example,

```
<!DOCTYPE html>
<html>
<head>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.7.1/jquery.min.js"></script>
<script>
$(document).ready(function(){
  $("#btn1").click(function(){
    $("#box").animate({height: "300px"});
  });
```

Effects of jQuery(Contd.)

```
$("#btn2").click(function(){
    $("#box").animate({height: "100px"});
  });
});
</script>
</head>
<body>
<button id="btn1">Animate height</button>
<button id="btn2">Reset height</button>
<div id="box" style="background:#000;height:100px;width:100px;margin:6px;"></div>
</body>
</html>
```

Functions of jQuery

The idea of Function is to put some commonly or repeatedly done tasks together and make a function so that instead of writing the same code again and again for different inputs, we can call that function.

Like in other languages, jQuery also providing User defined functions and pre-defined functions.

Pre-defined functions are the modules developed by jQuery community, these are nothing but the functions we had used so far. For example, hide(), animate(), ready() etc.,

And user defined functions will provide the flexibility to create our own functions to increase code reusability.

Functions of jQuery(Contd.)

Let us explore some more functions of jQuery to set the content using jQuery. The following methods are used to set the content which are listed below:

text(): It is used to set or return the text content of selected elements.

html(): It is used to set or return the innerHTML content of the selected element.

val(): This parameter is used to set or return the value of attribute for the selected elements. This method apply on the HTML form elements.

Functions of jQuery(Contd.)

Example,

```
<!DOCTYPE html>
<html>
<head>
        <title>jQuery Set Content</title>
        <script                                                    src=
"https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js"> </script>
</head>
<body style="text-align:center;">
        <h1 id="GFG1" style = "color:green;"> GeeksForGeeks </h1>
        <h2 id="GFG2">jQuery Set Content</h2>
```

Functions of jQuery(Contd.)

```html
<p>Full Form: <input type="text" id="GFG3" value="GFG"></p>
<button id="btn1">Set 1 line</button>
<button id="btn2">Set 2 line</button>
<button id="btn3">Set 3 line</button>
<!-- Script to set the content -->
<script>
        $(document).ready(function() {
                $("#btn1").click(function() {  $("#GFG1").text("GEEKSFORGEEKS");  } );
                $("#btn2").click(function() {  $("#GFG2").html("<b>Set Content</b>");  } );
                $("#btn3").click(function() {  $("#GFG3").val("GeeksForGeeks"); } );
        });
</script>
</body>
</html>
```

Functions of jQuery(Contd.)

A function is a set of statements that takes input, do some specific computation and produce output.

Syntax:

$.fn.myFunction = function(){}

Example,

```
<!DOCTYPE html>
<html>
<head>
        <title> How to Define jQuery function ? </title>
        <script src="https://code.jquery.com/jquery-1.12.4.min.js"> </script>
```

Functions of jQuery(Contd.)

```
<script>
        $(document).ready(function() {
                $.fn.myFunction = function() {
                        document.getElementById("geeks").innerHTML
                                = "JQuery function is defined!";
                }
                $(".gfg").click(function(){
                        $.fn.myFunction();
                });
        });
</script>
</head>
```

Functions of jQuery(Contd.)

```html
<body style="text-align:center">
    <h1 style="color:green;"> GeeksforGeeks </h1>
    <h3> Defining function in jQuery </h3>
    <p id="geeks"></p>
    <button type="button" class="gfg"> Click </button>
</body>
</html>
```