# jQuery

# What is jQuery?

- 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.

# Important core features supported by 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 featurerich 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.

# How to use jQuery?

There are two ways to use jQuery.

- **Local Installation** − Download jQuery library on local machine and include it in HTML code.

- **CDN Based Version** − Include jQuery library into HTML code directly from Content Delivery Network (CDN).

```
CDN:
<script type = "text/javascript"
  src = "https://ajax.googleapis.com/ajax/libs/jquery/2.1.3/jquery.min.js">
</script>
```

```
<head>
<script src="jquery-3.5.1.min.js"></script>
</head>
```

- How to get CDN path?

https://code.jquery.com/

```
<script src="https://code.jquery.com/jquery-2.2.4.js"
integrity="sha256-iT6Q9iMJYuQiMWNd9lDyBUStIq/8PuOW33aOqmvFpqI="
crossorigin="anonymous"></script>
```

# jQuery Syntax

Basic **syntax** is: $(selector).action()
- A $ sign to define/access jQuery
- A (selector) to "query (or find)" HTML elements
- A jQuery action() to be performed on the element(s)

**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".

# First working example

1. &lt;!DOCTYPE html&gt;

2. &lt;html&gt;

3. &lt;head&gt;

4. &lt;script src="jquery-3.6.0.js"&gt;&lt;/script&gt;

5. &lt;script&gt;

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

7.   $("button").click(function(){

8.     $("p").hide();

9.   });

10. });

11. &lt;/script&gt;

12. &lt;/head&gt;

13. &lt;body&gt;

Use CDN - https://ajax.googleapis.com/ajax/libs/jquery/3.6/jquery.min.js

14. &lt;h2&gt;This is a heading&lt;/h2&gt;

# The Document Ready Event

- jQuery methods in example, is inside a document ready event

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

   // jQuery methods go here...

});
```

- This is to prevent any jQuery code from running before the document is finished loading (is ready).
- The jQuery team has also created an even shorter method for the document ready event

```
$(function(){

   // jQuery methods go here...

});
```

# jQuery Selectors

- jQuery selectors allow you to select and manipulate HTML element(s).

## The element Selector

- You can select all <p> elements on a page like this: $("p")

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

## The #id Selector

```
$(document).ready(function(){
  $("button").click(function(){
    $("#test").hide();
  });
});
```

<p id="test">This is another paragraph.</p>

## The .class Selector

```
$(document).ready(function(){
  $("button").click(function(){
    $(".test").hide();
  });
});
```

`<h2 class="test">This is a heading</h2>`

## href example

```
<script>
$(document).ready(function(){
  $("button").click(function(){
    $("[href]").hide();
  });
});
</script>
```

`<p><a href="www.example1.com/">HTML Tutorial</a></p>`
`<p><a href="www.example2.com/">CSS Tutorial</a></p>`

# Examples of jQuery Selectors

| Syntax | Description |
| --- | --- |
| $("*") | Selects all elements |
| $(this) | Selects the current HTML element |
| $("p.intro") | Selects all <p> elements with class="intro" |
| $("p:first") | Selects the first <p> element |
| $("ul li:first") | Selects the first <li> element of the first <ul> |
| $("ul li:first-child") | Selects the first <li> element of every <ul> |
| $("[href]") | Selects all elements with an href attribute |
| $("a[target='_blank']") | Selects all <a> elements with a target attribute value equal to "_blank" |
| $("a[target!='_blank']") | Selects all <a> elements with a target attribute value NOT 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 |

# $("ul li:first-child")

<!DOCTYPE html>
<html>
<head>
<script
src="jquery-3.6.0.js">
</script>

```
<script>
$(document).ready(function(){
  $("button").click(function(){
    $("ul li:first-child").hide();
  });
});
</script>
```
</head>
<body>

 What about second child?

```
$("ul li:nth-child(2)").hide();
```
<p>List 1:</p>

# $("ul li:first")

```
<script>
$(document).ready(function(){
  $("button").click(function(){
    $("ul li:first").hide();
  });
});
</script>




<script>
$(document).ready(function(){
```

# jQuery 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 |

**click()** function is executed when the user clicks on the HTML element

<!DOCTYPE html>

<html>

<head>

<script src="jquery-3.6.0.js">

</script>

<script>

$(document).ready(function(){

  $("p").**click**(function(){

    $(this).hide();

  });

});

</script>

</head>

<body>

# **dblclick()** is executed when the user double clicks HTML element

<!DOCTYPE html>

<html>

<head>

<script src="jquery-3.6.0.js">

</script>

<script>

$(document).ready(function(){

  $("p").**dblclick**(function(){

    $(this).hide();

  });

});

</script>

</head>

<body>

# mouseenter()

```
<!DOCTYPE html>
<html>
<head>
<script src="jquery-3.6.0.js"></script>
<script>
$(document).ready(function(){
  $("#p1").mouseenter(function(){
    alert("You entered p1!");
  });
});
</script>
</head>
<body>

<p id="p1">Enter this paragraph and you will get the alert !!!</p>
```

```
<!doctype html>
<html lang="en">
<head>
 <meta charset="utf-8">
 <title>hover demo</title>
 <style>
 ul {
   margin-left: 20px;
   color: blue;
 }
 li {
   cursor: default;
 }
 span {
   color: red;
 }
 </style>
 <script src=" jquery-3.6.0.js "></script>
</head>
<body>

<ul>
 <li>Milk</li>
```

# Working with CSS

```html
<!DOCTYPE html>
<html>
<head>
<script type="jquery-3.6.0.js">
</script>

<script>
  $(document).ready(function() {
    $("p").css("background-color", "pink");
  });
</script>
</head>

<body>
<p>This is first paragraph.</p>
```

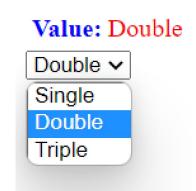# Working with Animation

```
<!DOCTYPE html>
<html>
<head>
<script src="jquery-3.6.0.js"></script>
<script>
$(document).ready(function(){
    $("button").click(function(){
        $("div").animate({left: '450px'});
    });
});
</script>
</head>

<body>
```

# Working with Select

```html
<!DOCTYPE html>
<html lang="en">
<head>

<title>val demo</title>
  <style>
 p {
   color: red;
   margin: 4px;
 }
 b {
   color: blue;
 }
  </style>
  <script src="jquery-3.6.0.js"></script>
</head>
<body>

<select id="single">
```

**Value:** Double

Double ⌄
| Single |
| Double |
| Triple |

# Working with class (addClass)

```html
<!DOCTYPE html>
<html>
<head>
<script src="jquery-3.6.0.js"></script>
<script>
$(document).ready(function(){
    $("button").click(function(){
        $("p:first").addClass("intro");
    });
});
</script>
```

**Also Check: hasClass()**

```html
<style>
.intro {
```

# jQuery wrap()

- jQuery wrap() method is used to wrap specified HTML elements around each selected element. The wrap () function can accept any string or object that could be passed through the $() factory function.

```
<!DOCTYPE html>
<html>
<head>
<script src=" "></script>

<script>
$(document).ready(function(){
   $("button").click(function(){
      $("p").wrap("<div></div>");
   });
});
</script>

<style>
div{background-color: pink;}
</style>
```

# jQuery serialize()

- jQuery **serialize()** method is used to create a text string in standard URL-encoded notation. It is used in form controls like <input>, <textarea>, <select> etc. It serializes the form values so that its serialized values can be used in the URL query string while making an AJAX request.

```
<!DOCTYPE html>
<html>
<head>
  <script src=" "></script>
<script>
$(document).ready(function(){
   $("button").click(function(){
      $("div").text($("form").serialize());
   });
});
</script>
</head>

<body>
<form action="">
```

# jQuery Traversing

# jQuery find() method

- The **find()** method in jQuery finds the descendant elements of the given selector. A descendant can be a child, grandchild, and so on. It is an inbuilt method in jQuery.

- For searching the descendant, the **find()** method traverse downwards from the selected element in the DOM tree.

- We can use the **"*"** selector for returning all descendant elements.

- To return all descendant elements of the given selector, we have to write:
  $(selector).find("*")

- The **children()** method works similarly to the find() method. Unlike the find() method, the children() method traverse a **single level down** the **DOM** tree, i.e., it returns the direct children.

```html
<!DOCTYPE html>
<html>
<head>
<style>
.main * {
  display: block;
  font-size: 20px;
  position: relative;
  border: 2px solid black;
  color: black;
  padding: 10px;
  margin: 17px;
}
</style>
<script src=" "></script>
<script>
function fun(){
$(document).ready(function(){
  $("#div1").find("ul").css({ "font-size": "30px", "color": "blue", "border": "6px dashed blue"});
});
```

# jQuery children() method

```
<!DOCTYPE html>
<html>
<head>
<style>
.main * {
  display: block;
  font-size: 20px;
  position: relative;
  border: 2px solid black;
  color: black;
  padding: 10px;
  margin: 17px;
}
</style>
<script src=" "></script>
<script>
function fun(){
$(document).ready(function(){
  $("div").children().css({ "font-size": "30px", "color": "blue", "border": "6px dashed blue"});
});
```

# With jQuery you can traverse up the DOM tree

An ancestor is a parent, grandparent, great-grandparent, and so on.

Traversing Up the DOM Tree

Three useful jQuery methods for traversing up the DOM tree are:

- o parent()
- o parents()
- o parentsUntil()

```
<script>
$(document).ready(function(){
  $("span").parent().css({"color": "red", "border": "2px solid red"});
});
</script>
```

```
<script>
$(document).ready(function(){
  $("span").parents().css({"color": "red", "border": "2px solid red"});
});
</script>
```

# jQuery AJAX

# jQuery ajax() method

- AJAX is an acronym for **Asynchronous JavaScript and XML**.

- It is a group of inter-related technologies like JavaScript, DOM, XML, HTML, XHTML, CSS, XMLHttpRequest etc.

- It allows us to send and receive data asynchronously without reloading the web page. So it is fast.

- The **ajax()** method in jQuery performs an AJAX request. It sends an synchronous HTTP request to the server.

- JQuery provides a rich set of AJAX methods for developing web applications. It is widely used for the requests.

- The syntax of using the ajax() method is:  **$.ajax({name:value, name:value, ... })**

# jQuery AJAX Example (test.html file is given)

```
<!DOCTYPE html>
<html>
<head>
<script src=" "></script>
<script>
$(document).ready(function(){
  $("button").click(function(){
jQuery.ajax({
url: 'test.html',
type: 'GET',
  success: function(data) {
    $("#para").html(data);
  }
});
```