

jQuery Introduction

- The jQuery is a JavaScript library.
- jQuery takes common tasks that cause many lines of code in javascript and wraps it in methods that can be called in a single line.

Use

- It is used to simplify the use of JavaScript on our website.
- It works the same in most of the standard browsers.

Ways to include

1. Download
2. CDN

The advantage of using hosted jQuery from Google is that many users have downloaded it while using other websites and it loads from cache, leading to faster loading time. Also, most CDNs ensure that once a user requests a file from it, it will served from the nearest server, which also leads to faster loading.

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.

HTML/CSS method of jQuery

- These methods are used to manipulate HTML and CSS.
- Some basic methods are as follows :
 - `addClass()`: add class names to the selected element
 - `after()`: inserts content after the selected element
 - `attr()`: sets or returns attribute value of selected element
 - `css()`: set or return style of the selected element
 - `before()`: inserts content after selected elements

`append()`: inserts content at the end of the selected element
`remove()`: removes the selected element
`empty()`: removes all children of the selected element
`removeClass()`: remove class from selected element specified by arguments
`removeAttr()` : removes attribute/s of selected elements
etc.

jQuery selector

- jQuery selectors allow you to select and manipulate HTML element/s.
- jQuery selectors are used to select HTML elements based on their name, id, classes, types, attributes, values of attributes etc.
- All selectors in jQuery start with the dollar sign and parentheses: `$()`.
- Some basic selectors are as follows :

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

Events in jQuery

- All the different visitors' actions that a web page can respond to are called events.
- Basic events :

Mouse Events	Keyboard Events
click	keypress
dblclick	keydown
mouseenter	keyup
mouseleave	

Mouse Events	Keyboard Events
click	keypress
dblclick	keydown
mouseenter	keyup
mouseleave	

- Event handling programmatically :

In this kind of event handling what to do when an event occurs is specified by the programmer through a callback function which then passes in arguments.

Example

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

- Custom logic on event fire

In this kind of event programmers can bind their events in their desired order or logic using .on() and .trigger() methods.

Example

```
$(document).ready(function(){
    $("p").click(function(){
        $("p").on("myPara", function(event, param1, param2,
param3){
            alert(param1 + "\n" + param2 + "\n" + param3);
        });
        $("p").trigger("myPara", ['Pass', 'Along', 'Parameters']);
    });
});
```

jQuery Functions

- Some inbuilt jQuery functions are listed below :

map(): Translate all items in an array or object to a new array of items.

grep(): Finds the elements of an array that satisfy a filter function.

merge(): Merge the contents of two arrays into the first array.

`each()`: A generic iterator function used to iterate over both objects and arrays.

Regular expression in jQuery

- JavaScript has an Object called `RegExp` to work with Regular Expressions. The syntax is `RegExp (pattern, modifier)`. Where pattern indicates the pattern you are searching for, and the modifier tells how to go about the search/(pattern matching) such as whether will it be in the global scope or will it be case sensitive or not.
- Methods of the `RegExp` object includes `test()`, `exec()`, and `toString()`. `test()` returns true or false based on the success or failure of the search. `exec()` will return the matched text. `toString()` when applied to a regular expression object will return the pattern in a string.

- **Example**

```
var searchPattern = new RegExp("wor");
searchPattern.test("hello world");
or
var pattern=/[0-9]/;
var isMatch=pattern.test(val)
```

Callback Functions

- Callback functions in jQuery are the same as it is in JavaScript.
- Callback functions are function which is passed as argument in another function.
- Example

```
$togglerBtn.click(function () {
    $("#sideBar").toggle(750);
})
```

Deferred and Promise object

- Deferred objects are a representation of an operation that may not have been completed yet.
- They can be used to trigger callbacks when the operation is either completed or failed and can be chained together to create asynchronous flows.
- Whereas a Promise object in jQuery is a representation of the result of a deferred operation. It allows you to access the result of a deferred operation once it has been completed and to attach callbacks to be executed when the operation is completed or fails.

- Example

```
var def = $.Deferred();  
var promise = def.promise();
```