

Experiment No. 07

Title: Write a jQuery script to handle event generated by client and demonstrate use of ajax methods.

Objective:

1. To study use of jQuery.
2. To study how to use jQuery library.

Theory:

jQuery is a fast, small, and feature-rich JavaScript library. It makes things like HTML document traversal and manipulation, event handling, animation, and Ajax much simpler with an easy-to-use API that works across a multitude of browsers. With a combination of versatility and extensibility, jQuery has changed the way that millions of people write JavaScript.

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.

The jQuery library contains the following features:

- HTML/DOM manipulation
- CSS manipulation
- HTML event methods
- Effects and animations
- AJAX
- Utilities

Adding jQuery to Your Web Pages -

There are several ways to start using jQuery on your web site. You can:

- Download the jQuery library from jquery.com
- Include jQuery from a CDN, like Google

Downloading jQuery-

There are two versions of jQuery available for downloading:

- Production version - this is for your live website because it has been minified and compressed
- Development version - this is for testing and development (uncompressed and readable code)

Both versions can be downloaded from jquery.com.

The jQuery library is a single JavaScript file, and you reference it with the HTML <script> tag (notice that the <script> tag should be inside the <head> section):

Example :

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

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()**

- 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 all current elements.
- **\$("p").hide()** – hides all <p> elements.
- **\$(".test").hide()** – hides all elements with class="test"
- **\$("#test").hide()** – hides all elements with id="test"

| 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 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" |
| \$("#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 |

What are Events?

All the different visitors' actions that a web page can respond to are called events. An event represents the precise moment when something happens.

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 |

Example 1:

```

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

```

Example 2:

```

$("p").dblclick(function(){
    $(this).hide();
});

```

Example 3: multiple event handlers to a <p> element

```

$("p").on({
    mouseenter: function(){
        $(this).css("background-color", "lightgray");
    },
    mouseleave: function(){
        $(this).css("background-color", "lightblue");
    },
    click: function(){
        $(this).css("background-color", "yellow");
    }
});

```

jQuery AJAX Methods:

AJAX is the art of exchanging data with a server, and update parts of a web page - without reloading the whole page. The following table lists all the jQuery AJAX methods:

| Method | Description |
|------------------------------------|--|
| \$.ajax() | Performs an async AJAX request |
| \$.ajaxPrefilter() | Handle custom Ajax options or modify existing options before each request is sent and before they are processed by \$.ajax() |
| \$.ajaxSetup() | Sets the default values for future AJAX requests |

| | |
|---|--|
| <code>\$.ajaxTransport()</code> | Creates an object that handles the actual transmission of Ajax data |
| <code>\$.get()</code> | Loads data from a server using an AJAX HTTP GET request |
| <code>\$.getJSON()</code> | Loads JSON-encoded data from a server using a HTTP GET request |
| <code>\$.parseJSON()</code> | Deprecated in version 3.0, use <code>JSON.parse()</code> instead. Takes a well-formed JSON string and returns the resulting JavaScript value |
| <code>\$.getScript()</code> | Loads (and executes) a JavaScript from a server using an AJAX HTTP GET request |
| <code>\$.param()</code> | Creates a serialized representation of an array or object (can be used as URL query string for AJAX requests) |
| <code>\$.post()</code> | Loads data from a server using an AJAX HTTP POST request |
| <code>ajaxComplete()</code> | Specifies a function to run when the AJAX request completes |
| <code>ajaxError()</code> | Specifies a function to run when the AJAX request completes with an error |
| <code>ajaxSend()</code> | Specifies a function to run before the AJAX request is sent |
| <code>ajaxStart()</code> | Specifies a function to run when the first AJAX request begins |
| <code>ajaxStop()</code> | Specifies a function to run when all AJAX requests have completed |
| <code>ajaxSuccess()</code> | Specifies a function to run when an AJAX request completes successfully |
| <code>load()</code> | Loads data from a server and puts the returned data into the selected element |
| <code>serialize()</code> | Encodes a set of form elements as a string for submission |
| <code>serializeArray()</code> | Encodes a set of form elements as an array of names and values |

Procedure:

- Create the different paragraphs using <p> tag.
- Link the jquery library using script tag.
- Apply different actions on paragraph like hide, show, toggle, slideup, slidedown using different event like click, dblclick, mouseover, keypress etc.
- Use Ajax methods to update parts of a web page - without reloading the whole page.

Key Concept: jQuery, CDN