

jQuery concept or what is JQuery?

- **jQuery** એ fast, small અને વધારે feature વાળી javascript library John Resig દ્વારા 2006 માં બનાવામાં હતી. તેનો હેતુ: **write less, do more** હતો.
- **jQuery** એ HTML document ને traversing, event handling, animating અને Ajax interactions ને simplify કરે છે.
- javascript ની complicated things ને jQuery simplify કરે છે. જેમ કે, Ajax call અને DOM manipulation.
- jQuery library માં નીચેના features હોય છે.
 - HTML/DOM manipulation
 - CSS manipulation
 - Event handling
 - Effects અને animations
 - AJAX Support
 - Lightweight
 - Cross Browser Support
- JQuery ઘણી બધી રીતે ઉમેરી શકાય છે. તમે આ રીતે કરી શકો છો. (1). jQuery.com માંથી jQuery library ને download કરો અથવા (2). CDN માંથી jQuery ને include કરી શકો છો. જેમ કે Google.
- તેને તમે HTML<script> tag સાથે add/reference કરી શકો છો. (<script> tag એ <head> tag section ની અંદર હોય છે.)

jQuery Syntax:

- jQuery ની Syntax એ selecting HTML elements પર અમુક action perform કરવા માટે બનાવામાં આવી છે.

`$(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".

- ઉદાહરણ માની બધી જ jQuery મેથડ, document ready event ની અંદર છે.

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

`// jQuery methods go here... });`

- ડોક્યુમેન્ટ સંપૂર્ણ load થાય ત્યાર બાદ **\$(document).ready()** method એ function ને execute કરવા માટે allow કરે છે.

jQuery Events

- Event એ વિવિધ પ્રકાર ના visitor ની action છે જે webpage/web application ને respond આપી શકે છે. જ્યારે કોઈ moment થાય ત્યારે ચોક્કસ event ને represent કરે છે.
- જુદી જુદી event ના ઉદાહરણ નીચે મુજબ છે.
 - A mouse click
 - Selecting a radio button
 - A web page loading
 - Moving a mouse over an elements
 - Submitting an HTML form
 - A keystroke on your keyboard
- નીચે કેટલીક સામાન્ય રીતે વપરાયેલી DOM events છે.

Mouse Events	Keyboard Events	Form Events	Document/Window Events
click	Keypress	submit	load
dblclick	Keydown	change	resize
hover	Keyup	focus	scroll
mouseenter		blur	unload
mouseleave			

Mouse Events

- Mouse events એવી events છે જે mouse ની action સાથે હોય છે. જેમ કે, click, double-click, mouse down, hover વગેરે હોય શકે છે.
- Mouse events નો ઉપયોગ બે જુદી જુદી રીતે કરી શકાય છે.
 - Click handler ને HTML elements સાથે bind કરવાનો અને,
 - jQuery code ની અંદર click events ને call કરવાનો છે.

1) click()

- click() method એ event handler function ને HTML element સાથે જોડે છે. જ્યારે user HTML elements પર ક્લિક કરે છે ત્યારે ફંક્શન execute થાય છે.

Syntax:

Selected elements માટે click event ને trigger કરવા માટે : \$(selector).click()

Click event માં function ને attach કરવા માટે : \$(selector).click(function) // function is optional

Example:

jQuery code:

```
<script src="http://ajax.googleapis.com/ajax/libs/jquery/1.11.2/jquery.min.js"> </script>
<script>
    $(document).ready(function(){
        $("p").click(function(){
            $(this).hide();
        });
    });
</script>
```

HTML Code:

```
<p>Click Me</p>
```

- ઉપરના ઉદાહરણમાં, <p> element પર ક્લિક કરતી વખતે click() method triggered થાય છે.

2) dblclick()

- dblclick event ત્યારે આવે છે જ્યારે element પર double-click કરવામાં આવે છે. dblclick() method dblclick events ને trigger કરે છે અથવા જ્યારે dblclick event occurs થાય ત્યારે run કરવા માટે ફંક્શન ને attached કરે છે.

Syntax:

Selected elements માટે click event ને trigger કરવા માટે: \$(selector). dblclick()

Click event માં function ને attach કરવા માટે: \$(selector). dblclick(function) // function is optional

Example:

jQuery code:

```
<script src="http://ajax.googleapis.com/ajax/libs/jquery/1.11.2/jquery.min.js"> </script>
<script>
    $(document).ready(function(){
        $("p").dblclick(function(){
            $(this).hide();
        });
    });
</script>
```


HTML Code:

<p>Double-Click on Me</p>

3) hover()

- hover() method જ્યારે mouse pointer એ select કરેલા HTML element માં enter થાય ત્યારે અને તેને leave કરે છે ત્યારે fire થાય છે.
- hover() methodમાં બે ફંક્શન છે અને તે mouseenter() અને mouseleave() method નું combination છે. માઉસ જ્યારે HTML elements માં enter થાય છે ત્યારે પહેલું ફંક્શન executed થાય છે અને માઉસ જ્યારે HTML elements leave કરે છે ત્યારે બીજું ફંક્શન executed થાય છે.

Syntax:

\$(selector).hover(inFunction, outFunction)

- **inFunction** એ required છે જ્યારે mouseenter event occurs થાય ત્યારે run કરવા માટે આ ફંક્શન specify કરવામાં આવે છે.
- **outFunction** એ optional છે જ્યારે mouseleave event occurs થાય ત્યારે run કરવા માટે આ ફંક્શન specify કરવામાં આવે છે.

Example:

jQuery code:

```
<script src="http://ajax.googleapis.com/ajax/libs/jquery/1.11.2/jquery.min.js"> </script>
<script>
    $(document).ready(function(){
        $("#MouseHover").hover(function(){
            $("div").css("background-color", "pink");
        }, function(){
            $("div").css("background-color", "lightblue");
        });
    });
</script>
```

HTML Code:

<div id="MouseHover">Hover the mouse pointer to see the effect.</div>

- ઉપર નો code <div> elements નો background color બદલી નાખે છે જ્યારે માઉસ નું pointer તેના પર hover થાય છે.

Keyboard Events

- **jQuery keyboard events** કોઈ પણ HTML element સાથે હોઈ શકે છે. પરંતુ તે ફક્ત જે element ને focus કરે છે તે event ને fire કરશે.

- મોટા ભાગ ની કીબોર્ડ event form elements (HTML input elements) સાથે જોડાયેલ હોય છે. જેમ કે basic કીબોર્ડ events **keypress, keydown, keyup, keyrelease** વગેરે.

1) keypress()

- કીબોર્ડ માં keypress કરવામાં આવે પછી **jquery keypress event** trigger થાય છે. જ્યારે keypress event occur થાય ત્યારે તેમાં attach કરેલ ફંક્શન run થાય છે.
- Keypress event એ keydown event જેવી જ છે. જ્યારે બટન press કરવામાં આવે ત્યારે event occur થાય છે.
- જ્યારે modifier keys (**ALT, CTRL, SHIFT, and ESC**) press કરવામાં આવે છે. ત્યારે keypress event fire થશે નહિ.

Syntax:

Selected elements માટે click event ને trigger કરવા: \$(selector).keypress()

Click event માં function ને attach કરવા માટે: \$(selector).keypress(function) // function is optional

Example:

jQuery code:

```
<script src="http://ajax.googleapis.com/ajax/libs/jquery/1.11.2/jquery.min.js"></script>
<script>
    i = 0;
    $(document).ready(function(){
        $("#txtName").keypress(function(){
            $("#label").text(i += 1);
        });
    });
</script>
```

HTML Code:

Enter your name: <input type="text" id="txtName">

<p>Total Keypresses are: <label>0</label> </p>

- ઉપરોક્ત code એ <input> field માં keypress થાય એ number ને count કરે છે. જે user દ્વારા enter કરવામાં આવે છે.

2) keydown()

- જ્યારે કીબોર્ડ ની key press કરવામાં આવે ત્યારે keydown event trigger થાય છે. **keydown()** method એ **keydown** event ને trigger કરે છે અથવા જ્યારે keydown event થાય ત્યારે તેમાં attach કરેલ ફંક્શન run થાય છે.

- જ્યારે modifier keys (**ALT, CTRL, SHIFT, and ESC**) press કરવામાં આવે છે ત્યારે keydown event fire થશે.

Syntax:

Selected elements માટે click event ને trigger કરવા: \$(selector).keydown()

Click event માં function ને attach કરવા માટે: \$(selector).keydown(function) //function is optional

Example:

jQuery code:

```
<script src="http://ajax.googleapis.com/ajax/libs/jquery/1.11.2/jquery.min.js"> </script>
<script>
    $(document).ready(function(){
        $("#txtName").keydown(function(){
            $("#txtName").css("background-color", "pink");
        });
        $("#txtName").keyup(function(){
            $("#txtName").css("background-color", "lightblue");
        });
    });
</script>
```

HTML Code:

<p>Enter your Name and see the effect of keydown and keyup event.</p>

Enter your name: <input type="text" id="txtName">

- જ્યારે કીબોર્ડ ની keypress કરવામાં આવે અથવા release કરવામાં આવે ત્યારે ઉપર ના કોડ માં <input> field નો background color set કરે છે.

3) keyup()

- જ્યારે keypress અને પછી release કરવામાં આવે ત્યારે keyup event fire થાય છે. જ્યાં સુધી keyrelease નહિ થાય ત્યાં સુધી આ event fire થશે નહીં. **keyup()** method keyup ની event ને trigger કરે છે અથવા જ્યારે keyup event થાય ત્યારે તેમાં attach કરેલ ફંક્શન run થાય છે.
- keyup અને keypress / keydown વચ્ચે એ તફાવત છે કે જ્યારે key નીચે hold કરેલી હોય ત્યારે આ event fire નહિ થાય પરંતુ બીજી keyboard event key નીચે hold કરેલી હશે ત્યારે fire થશે.

Syntax:

Selected elements માટે click event ને trigger કરવા: \$(selector).keyup()

Click event માં function ને attach કરવા માટે: \$(selector).keyup(function) // function is optional

Example:

jQuery code:

```
<script src="http://ajax.googleapis.com/ajax/libs/jquery/1.11.2/jquery.min.js"></script>
<script>
    $(document).ready(function(){
        $("#txtName").keydown(function(){
            $("#txtName").css("background-color", "pink");
        });
        $("#txtName").keyup(function(){
            $("#txtName").css("background-color", "lightblue");
        });
    });
</script>
```

HTML Code:

<p>Enter your Name and see the effect of keydown and keyup event.</p>
Enter your name: <input type="text" id="txtName">

- **Note: jQuery** માં કોઈ keyrelease event નથી પરંતુ આપણે તેને keyup ની event તરીકે લઈ શકીએ છીએ.

Form Events

- jQuery form elements ની action ને handle કરવા માટે ઘણી form events આપે છે. Form events એ standard JavaScript events છે પરંતુ jQuery દ્વારા add કરેલ ઘણા form જુદી જુદી events support છે. જેમ કે submit, change, select, blur વગેરે.

1) submit()

- Submit event ત્યારે આવે છે જ્યારે કોઈ form એ form submission element (button, input (type button અથવા image) અને enter (અમુક caseમાં) નો ઉપયોગ કરે. આ event નો ઉપયોગ <form> elements પર જ થઈ શકે છે.
- submit() મેથડ એ form ને સબમિટ કરો ત્યારે event ને trigger કરે છે અથવા submit event થાય તેમાં attach કરેલ ફંક્શન run થાય છે.

Syntax:

Selected elements માટે click event ને trigger કરવા: \$(selector).submit()

Attach a function to the submit event: \$(selector).submit(function) // function is optional

Example:

jQuery code:

```
<script src="http://ajax.googleapis.com/ajax/libs/jquery/1.11.2/jquery.min.js"> </script>
<script>
    $(document).ready(function(){
        $("#Registraion").submit(function(){
            alert("Great!! Form submitted Successfully");
        });
    });
</script>
```

HTML Code:

```
<form id="Registraion" action="#">
<label>FirstName: </label> <input type="text" name="FirstName" id="FirstName" />
    <input type="submit" value="Submit">
</form>
```

2) ready()

- જ્યારે DOM load થાય ત્યારે ready event occur થાય છે. કારણ કે ડોક્યુમેન્ટ ready થાય પછી આ event occur થાય છે.
- Ready event નો હેતુ એ છે કે તે ડોક્યુમેન્ટ ને load કર્યા પછી શક્ય તેટલું વહેલું થવું જોઈએ.
- Ready મેથડ સ્પષ્ટ કરે છે જ્યારે ready event થાય ત્યારે શું થાય છે. <Body onload = ""> સાથે ready() મેથડ નો ઉપયોગ એક સાથે થવો જોઈએ નહીં.
- html document load થયા પછી ready event occurs થાય છે જ્યારે બધા content load થઈ જાય પછી onload event occur થાય છે onload event DOM માં standard event છે જ્યારે ready event jquery માટે special છે.

Syntax:

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

Ready () methodનો ઉપયોગ current document પર જ થઈ શકે છે, તેથી કોઈ selector આવશ્યક નથી: \$(function) // function is required

Example:

jQuery code:

```
<script src="http://ajax.googleapis.com/ajax/libs/jquery/1.11.2/jquery.min.js"> </script>
<script>
    $(document).ready(function(){
        $("#Registraion").submit(function(){
```



```

        alert("Great!! Form submitted Successfully");
    });
});
</script>

```

HTML Code:

```

<form id="Registraion" action="#">
<label>FirstName: </label> <input type="text" name="FirstName" id="FirstName"/>
    <input type="submit" value="Submit">
</form>

```

Document/ Windows Events

1) load()

- જ્યારે parent element ની નીચેના બધા sub element અને બીજા element load થાય છે ત્યારે load() event fire થાય છે
- Load event ત્યારે થાય છે જ્યારે images, iframes, frames, scripts અથવા window ડોક્યુમેન્ટ જેવા specify element સંપૂર્ણ પણે (completely) load થાય.
- Load () method jQuery version 1.8 માં disapprove કરવામાં આવી હતી.

Syntax:

```
$(selector).load(function) /*function is optional */
```

Example:

jQuery code:

```

<script src="http://ajax.googleapis.com/ajax/libs/jquery/1.11.2/jquery.min.js"> </script>
<script>
    $(document).ready(function(){
        $("#LoadImage").load(function(){
            alert("Image loaded successfully.");
        });
    });
</script>

```

HTML Code:

```



```

2) resize()

- જ્યારે browserની window resize થાય ત્યારે resize event occur થાય છે. આ event ને properly work કરવા માટે "window" element નું attached હોવું આવશ્યક છે.

- `resize()` method એ `resize` event ને trigger કરે છે અથવા જ્યારે `resize` event થાય ત્યારે તેમાં attached કરેલ ફંક્શન run થાય છે.

Syntax:

Selected elements માટે click event ને trigger કરવા: `$(selector).resize()`

Click event માં function ને attach કરવા માટે: `$(selector).resize(function)` // function is optional

Example:

jQuery code:

```
<script src="http://ajax.googleapis.com/ajax/libs/jquery/1.11.2/jquery.min.js">
</script>
<script>
    $(document).ready(function(){
        $(window).resize(function(){
            alert("You resized me!");
        });
    });
</script>
```

HTML Code:

```
<p>Try to resize your browser window.</p>
```

3) scroll()

- Scroll event ત્યારે થાય છે જ્યારે યુઝર specify element માં scroll કરે છે.
- Scroll event બધા scrollable elements અને window object માટે work કરે છે.

Syntax:

Selected elements માટે click event ને trigger કરવા: `$(selector).scroll()`

Click event માં function ને attach કરવા માટે: `$(selector).scroll(function)` //function is optional

Example:

jQuery code:

```
<script src="http://ajax.googleapis.com/ajax/libs/jquery/1.11.2/jquery.min.js"> </script>
<script>
    $(document).ready(function(){
        $("#intro").scroll(function(){
            alert("Great!! You scrolled me!");
        });
    });
</script>
```


</script>

HTML Code:

<p>Try the scroll in the div</p>

<div id="intro" style="overflow:scroll; height:100px; width: 200px;"> Lorem ipsum dolor sit amet, consectetur adipiscing elit. Mauris dapibus fermentum leo, ut consectetur libero sagittis eu. Sed vel interdum diam. Curabitur quis magna dolor.

</div>

4) unload()

- જ્યારે યુઝર navigates પેજ થી દુર જાય છે ત્યારે unload event occur થાય છે.
- unload event triggered થાય છે જ્યારે પેજ leave કરવા માટેની link પર ક્લિક કરવામાં આવે છે addressbar માં એક નવું URL type કરવામાં આવે છે તે forward અથવા back button નો ઉપયોગ થાય છે, browser window બંધ થાય છે પેજ ફરીથી load થાય છે
- unload() method જ્યારે unload event થાય છે ત્યારે શું થાય છે તે specify કરે છે.
- unload() method jquery version 1.8 માં disapprove કરવામાં આવી છે.

Syntax:

\$(selector).unload(function) // function is optional

Example:

jQuery code:

<script src="http://ajax.googleapis.com/ajax/libs/jquery/1.11.2/jquery.min.js"> </script>

<script>

```
$(document).ready(function(){
    $(window).unload(function(){
        alert("Please Wait for the page load.");
    });
});
```

</script>

HTML Code:

Click Here to go Google.com

bind() Method

- jQuery દ્વારા events સાથે deal કરવાના સૌથી મહત્વ નો factor bind() અને unbind() મેથડ છે.
- Event triggers અને handlers ને તમારા DOM element માં add કરવા માટે bind() method ખુબજ ઉપયોગી છે અને custom elements ને પણ bind કરી શકે છે

- `unbind()` method એ event handlers ને select કરેલા element માંથી દૂર કરે છે. આ method બધા અથવા select કરેલા event handler ને remove કરે છે અથવા event occur થાય ત્યારે run થતી specify ફંક્શન ને બંધ કરે છે
- Possible event values blur, focus, load, resize, scroll, unload, click વગેરે.
- `bind()` method select કરેલા elements માટે એક અથવા વધુ event handlers ને જોડે (bind) છે અને જ્યારે event occur થાય ત્યારે run કરવા માટે ફંક્શન specify કરે છે.

Syntax:

`$(selector).bind(event, data, function, map)`

- **event** જરૂરી parameter છે જે event થાય ત્યારે run કરવા માટે ફંક્શન ને specify કરે છે.
- **data** એ optional parameter છે જે ફંક્શન ની સાથે pass કરવા વધારાના data ને specify કરે છે.
- **function** જરૂરી parameter છે જે event થાય ત્યારે run કરવા માટે ફંક્શન ને specify કરે છે.
- **map** event map({event : function ,event : function ,.....}) ને specific કરે છે એક અથવા વધારે events આવેલી હોય છે અને જ્યારે event occur થાય ત્યારે ફંક્શન run થાય છે.

Example:

jQuery code:

```
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js"> </script>
<script>
    $(document).ready(function(){
        $("p").bind("mouseout mouseover", function(){
            $("p").toggle(".intro");
        });
    });
</script>
<style>
.intro {
    font-size: 150%;
    color: blue;
}
</style>
```

HTML Code:

```
<p class="intro">Hover the mouse pointer over this paragraph.</p>
```


Event Helper Methods

- jQuery event એ helper ફંક્શન નો એક set પણ આપે છે જેનો ઉપયોગ ઉપર mention કરેલા કોઈ પણ event type ને બંધ કરવા માટે અથવા eventને trigger કરવા માટે ઉપયોગ માં લેવાય છે.

- Trigger Methods:**

નીચે આપેલા ઉદાહરણ માં બધા paragraph પર blur event ને trigger કરે છે.

```
$("p").blur();
```

- Binding Methods:**

નીચે આપેલા ઉદાહરણ માં <div> પર ક્લિક event ને bind કરશે.

```
$("#div").click( function () {  
    // do something here  
});
```

- અહીં JQuery અને અન્ય દ્વારા આપવામાં આવેલી બાકી ની મેથડ નું list છે જે પહેલા જ આ chapter માં ચર્ચા થઈ ચૂકી છે.

- **blur():** દરેક match થતી element ની blur event ને triggers કરે છે.
- **blur(fn):** દરેક match થતી element ની blur event માં ફંક્શન ને bind કરે છે.
- **change():** દરેક match થતી element ની changed event ને triggers કરે છે.
- **change(fn):** દરેક match થતી element ને change event માં ફંક્શન ને bind કરે છે.
- **error():** દરેક match થતી element ની error event ને triggers કરે છે.
- **error(fn):** દરેક match થતી element ની error event માં ફંક્શન ને bind કરે છે.
- **focus():** દરેક match થતી element ની focus event ને triggers કરે છે.
- **focus(fn):** દરેક match થતી element ને focus event માં ફંક્શન ને bind કરે છે.
- **mousedown(fn):** દરેક match થતી element ની mousedown event માં ફંક્શન ને bind કરે છે.
- **mouseenter(fn):** દરેક match થતી element ની mouseenter event માં ફંક્શન ને bind કરે છે.
- **mouseleave(fn):** દરેક match થતી element ની mouseleave event માં ફંક્શન ને bind કરે છે.
- **mousemove(fn):** દરેક match થતી element ની mousemove event માં ફંક્શન ને bind કરે છે.
- **mouseout(fn):** દરેક match થતી element ની mouseout event માં ફંક્શન ને bind કરે છે.
- **mouseover(fn):** દરેક match થતી element ની mouseover event માં ફંક્શન ને bind કરે છે.
- **mouseup(fn):** દરેક match થતી element ની mouseup event માં ફંક્શન ને bind કરે છે.
- **select():** દરેક match થતી element ની select કરેલી event ને trigger કરે છે.
- **select(fn):** દરેક match થતી element ની select event માં ફંક્શન ને bind કરે છે.

Note: bind() અને helper મેથડ ના ઉદાહરણ ઉપર સમજાવેલા છે. (Refer Page No. 11-12)