



JQuery

- * JQuery is JS library
- * JQuery is developed completely based on JS
- * JQuery is open source JS library
- * JQuery is not replacement of JS. It simplifies the JS task
- * JQuery is a cross platform JS library designed to simplify the client side scripting of HTML
- * JQuery is used to overcome the load on server. Many task can be performed in client side using JQuery.
- * Its punch line is "Write less do more..."

Inventor of JQuery - John Resig
Created in 2006

Features of JS

- * Simple
 - * DOM manipulation
 - * DOM Traversing
 - * Event Handling
 - * Styles Modifications
 - * Effects and Animation
 - * AJAX Support (Asynchronous JS & XML)
 - * JQuery - UI (user Interface)
 - * CSS3 Compliant
 - * Light Weight Library
 - * Cross Browser Support
 - * Interpreted
- Extensible Markup Language
↑



```
Ex- $(document).ready(function(){  
    alert("Welcome to JQuery.");  
});
```

Explanation of above code -

① \$ -

- * \$ is ^{factory} jquery function. It is used to begin the coding in jquery.
- * \$ function is used for DOM traversing.
- * \$ function is used to select an object from f
- * \$ is short form of "jquery()" function.

② \$(document) -

- * document is a built-in object of DOM (whole webpage)
- * \$(document) means selecting object of whole page.

③ ready() -

- * ready is an event of jquery.
- * This event is used to run some code after page load / page ready.
- * ready() event is equivalent to "onload" event of JS.

Note - In jQuery, when we use an event then we have to specify event handler function to perform that events. For event handlers we can use named or unnamed function both.

Ex -

```
function demo()
{
    alert("welcome to our page");
}
$(document).ready(function demo());
```

DOM Accessing in jQuery: -

A We can access the object of any tag from webpage by using jQuery.

A To access/select an object we have to use "\$()" function.

Syntax - \$("#selector")

Note :- We have main 3 types of selectors -
Id selector (#), class selector (.) and tag based selector (without any symbol).

\$() function returns the object of associated tag.

DOM Manipulation in jQuery -

We can read and modify (write) the content of any HTML tag by using jQuery.

To perform DOM manipulation jQuery provides below methods -

- 1- `val()` \Rightarrow value property in JS
- 2- `text()` \Rightarrow innerText property in JS
- 3- `html()` \Rightarrow innerHTML property in JS
- 4- `prop()`
- 5- `attr()`

Syntax of `val()` / `text()` / `html()`

① To get

`var name = Object of tag.Function Name();`

Ex-

```
var name = $("#txtname").val();
```

```
var para = $("#mypara").text();
```

~~var~~

② To Set -

`Object of tag.FunctionName(value/variable);`

Ex-

```
$("#txtname").val("Ajit Singh");
```

```
$("#mydiv").text("my name is Raj");
```

```
$("#mydiv").html("My Name is Raj");
```



- ④ prop() ⇒ This function is used to get/set the property of a tag
- ⇒ This property function get/set the property in real format. (not in string format).
- ⇒ We can use this function to work with checked, disabled, readonly property.

Syntax to get:

varname = object-of-tag.prop("Property Name");

Ex- var x = \$("#btn").prop("disabled");

Syntax to set:

object of tag.prop("prop-name", "prop-value");

Ex- \$("#chk").prop("checked", true);

⑤ attr() ⇒

- ① This function is used to get/set the attributes of HTML tags
- ② We can use this function to read/write src, title (of img tag), rows, cols (textarea), type (of input tag) etc.
- ③ This function reads or write data in string format (not real format).
Its syntax is same as "prop()", "function"



Managing Styles in JQuery

- ① We can read and manage css properties by using jquery
- ② JQuery provides below some methods-
 - ① css()
 - ② addClass()
 - ③ removeClass()
 - ④ toggleClass()

① css()

This function is used to get/set the css property

Syntax-to get :-

var name = obj. of tag. css ("property-name");

Ex-

var col = \$("#dv"). css ("color");

Syntax to set :-

obj. of tag. css ("property-name", "value");

obj. of tag. css ({ "property": "value", "property": "value", ... });

for
multiple
css

\$("#dv"). css ("background-color", "blue");

Note: `addClass()` junction is used to add/apply a CSS class in selected object during runtime.

* `removeClass()` junction is used to remove the CSS class from selected tag.

* `toggleClass()` junction creates toggle effect (add/remove).

Syntax of `addClass()` -

Object of tag `addClass("Name of CSS class");`

Ex. `$("#txt").addClass("form-control");`

Syntax of `toggleClass()` & `removeClass()` is same as above.

Some of DOM Events

click
dblclick
mouseenter
mouseleave

} Mouse
Events

keypress
keydown
keyup

} Keyboard
Events

submit
change
focus
blur

} Form
Events

load
resize
scroll
unload

} Document
Events

Some others are `hide()`, `show()`, `toggle()`,
`fadeIn()`, `fadeOut()`, `fadeToggle()`.



`slideDown()` - used to slide down an element
`slideUp()` - used to slide up an element
`slideToggle()` - used to slide up/down both
`animate({params}, speed, callback);`
`stop()` - used to stop an animation

jQuery Chaining

With jQuery, you can chain together actions/methods.

Chaining allows us to run multiple jQuery methods within a single statement.

Ex- `$("#p1").css("color", "red").slideUp(2000).slideDown(2000);`

Add New HTML Content

`append()` - Inserts content at the end of selected items.

`prepend()` - Inserts content at the beginning of the selected items.

`after()` - Inserts content after the selected elements (generally used with `img`).

`before()` - Inserts content before the selected items.



Remove Elements/Contents

remove() - Removes the selected element (and its child elements).

empty() - Removes the child elements from the selected element.

jQuery Dimensions Methods

width()

height()

innerWidth()

innerHeight()

outerWidth()

outerHeight()

Working with collection in jQuery

We can perform get/set operations in collection by using jQuery.

each() function is used to work with collections.

Syntax-

```
CollectionObject.each(function() {
    // statements
});
```

Note- We can access current object from collection inside each function by using \$(this).each() function works like a loop.

Effects and Animations using jQuery

jQuery provides features to create effect and animations

To implement effect and animation, we have to use below functions -

- | | | | |
|---|---------------|----|--------------|
| 1 | show() | 7 | fadeIn() |
| 2 | hide() | 8 | fadeOut() |
| 3 | toggle() | 9 | fadeToggle() |
| 4 | slideDown() | 10 | animate() |
| 5 | slideUp() | | |
| 6 | slideToggle() | | |



⇒ We have to pass duration of effect/animation inside above functions. We can pass this duration in milliseconds or in string format.

⇒ Duration of string format can be -
slow
normal
fast

⇒ Duration in milliseconds can be 1000, 1800, 4000 etc

Syntax of first 9 functions-

obj- of- `func. functionName (duration);`

Ex- `$("#dv"). fadeIn ("slow");`
`$("#btn"). hide (2300);`

⑩ animate()

This function is used to create custom animation

Syntax-

object of `func. animate (Effects, duration);`

Ex-

`$("#im"). animate ({width: 200px}, 3000);`