

## 4. jQuery

- 1) Why jQuery?
- 2) The way of writing
- 3) Selector Syntax
- 4) Selector
- 5) DOM Events
- 6) DOM Effect
- 7) Manipulation HTML
- 8) Traversing

## 1) What is jquery?

Jquery ဟာ JavaScript ကိုအခြေခံထားတဲ့ library တခုဖြစ်ပါတယ်။

JavaScript ကိုပိုမိုလွယ်ကူအောင်ရေးသားနိုင်ဖို့

ကျစ်ကျစ်လစ်လစ်တည်ဆောက်ထားတဲ့ Library တခုဖြစ်ပါတယ်။

## 2) The way of writing

Jquery ကို download လုပ်နိုင်ဖို့အတွက် ဒီလင့်ကနေ တဆင့် Download လုပ်ပါ။

<https://jquery.com/download/>

### Syntax

```
$(document).ready(function(){  
});  
  
$(function(){  
// Short Document Ready  
});
```

## 3) Selector Syntax

```
$(selector).action()
```

Jquery ရဲ့ syntax ဟာ ရိုးရှင်းပါတယ်။ \$ ကတော့ Element ကို accessing လုပ်ဖို့အတွက်ဖြစ်ပါတယ်။ Selector ကတော့ Html ဘက်က element ကို select လုပ်တာဖြစ်ပါတယ်။ Action ကတော့ selector ကို ဘယ်လို event or appear ဖြစ်မလဲဆိုတာသတ်မှတ်ပါတယ်။

Example :

1. `$("p").text("Hello World!");`
2. `$(".demo").click(function(){  
 $(this).hide(200);  
});`

#### 4) Selector

Jquery မှာ selector တွေများစွာရှိပါတယ်။html element တွေရဲ့ id,name, Attribute,type,value စတာတွေကို accessing လုပ်နိုင်ပါတယ်။

`$("*")`

Universal HTML ဘက်ကelement အားလုံးပေါ်မှာသက်ရောက်ပါတယ်။

`$("p.demo")`

P tag ထဲကClass attribute နဲ့ရေးထားတဲ့ demo element(s) တွေပေါ်မှာeffectဖြစ်စေပါတယ်။

`$("p:first")`

P tag ထဲက ပထမဆုံး elementကို effect ဖြစ်စေပါတယ်။

`$("p span")`

P tag ရဲ့child သို့မဟုတ် nested child တွေထဲက span tag တွေကို effect ဖြစ်စေပါတယ်။

`$("p > span")`

P tag ထဲက direct child span ကို effect သက်ရောက်စေပါတယ်။

`$("p + span")`

P tag နောက်ကနေလာတဲ့ next level တွာ span tag ကို effect သက်ရောက်စေပါတယ်။

`$("p ~ span")`

P tag နောက်ကနေလာတဲ့ level တွာ span tag ကို effect သက်ရောက်စေပါတယ်။

`$("ul li:first")`

Ul tag ထဲ ပထမဆုံး li ကို သက်ရောက်ပါတယ်။ပထမဆုံး element တခုကိုသာလျှင်သက်ရောက်စေပါတယ်။

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

Ul tag ထဲ ပထမဆုံး li ကို သက်ရောက်ပါတယ်။multiple element  
တိုင်းကိုသက်ရောက်ပါတယ်။

`$("ul li:nth-child(2)")`

Ul tag ထဲ နှစ်ကြိမ်မြောက် li ကို သက်ရောက်ပါတယ်။multiple element  
တိုင်းကိုသက်ရောက်ပါတယ်။

`$("[href]")`

Attribute မှာ href လို့ရေးထားတဲ့ element ကို သက်ရောက်စေပါတယ်။

`$("a[target='_blank']")`

A tag ရဲ့ attribute target မှာ \_blank လို့ရေးထားတဲ့ a tag ကို effect  
ဖြစ်စေပါတယ်။

`$("a[target!='_blank']")`

A tag ရဲ့ attribute target မှာ \_blank လို့ရေးမထားတဲ့ a tag ကို effect  
ဖြစ်စေပါတယ်။

`$(":input")`

Input tag တိုင်းကို သက်ရောက်စေပါတယ်။

`$(":button")`

Button tag နဲ့ input type button တိုင်းကို effect ဖြစ်စေပါတယ်။

`$("tr:even")`

Table ရဲ့ tr even number မှာရှိတဲ့ row တွေပေါ်မှာ effect  
သက်ရောက်စေပါတယ်။

`$("tr:odd")`

Table ရဲ့ tr odd number မှာရှိတဲ့ row တွေပေါ်မှာ effect  
သက်ရောက်စေပါတယ်။

```
$("#span:parent")
```

Span tag ထဲ ဘာစာမှရေးမထားတဲ့ tag ကို effect သက်ရောက်စေပါတယ်။

```
$("#span:contains('demo')")
```

Span tag တွေရှိတဲ့အထဲက demo ဆိုတဲ့ စာသားပါတဲ့ span tag ကို သက်ရောက်စေပါတယ်။

## 5) DOM Events

### ■ Mouse Events

#### 1. click

```
$( "#target" ).click(function() {  
    // code  
});
```

#### 2. Contextmenu

Target ပေါ်မှာ right click နှိပ်လိုက်တဲ့အခါ ဖြစ်ပေါ်လာမယ့် event ဖြစ်ပါတယ်။

```
$( "#target" ).contextmenu(function() {  
    // code  
});
```

#### 3. Dblclick

Double click event နှိပ်လိုက်တဲ့အခါ ဖြစ်ပေါ်လာမယ့် event ဖြစ်ပါတယ်။

```
$( "#target" ).dblclick(function() {  
    // code  
});
```

#### 4. hover

```
$( "#target" ).hover(function() {  
    // mouse over  
}, function() {  
    // mouse out  
}  
);
```

5. mousedown

```
$( "#target" ).mousedown(function() {  
    // code  
});
```

6. mouseenter

```
$( "#target" ).mouseenter(function() {  
    // code  
});
```

7. mouseleave

```
$( "#target" ).mouseleave(function() {  
    // code  
});
```

8. mousemove

```
$( "#target" ).mousemove(function() {  
    // code  
});
```

9. mouseout

```
$( "#target" ).mouseout(function() {  
    // code  
});
```

10.mouseover

```
$( "#target" ).mouseover(function() {  
    // code  
});
```

11.mouseup

```
$( "#target" ).mouseup(function() {  
    // code  
});
```

12.toggle

```
$( "#target" ).toggle(function() {  
    // code  
});
```

## ■ Window Events

### 1. resize

```
$( window ).resize(function() {  
    // code  
});
```

### 2. scroll

```
$( window ).scroll(function() {  
    // code  
});
```

## ■ Keyboard Events

### 1. keydown()

```
$( "#target" ).keydown(function() {  
    // code  
});
```

### 2. keypress()

```
$( "#target" ).keypress(function() {  
    // code  
});
```

### 3. keyup()

```
$( "#target" ).keyup(function() {  
    // code  
});
```

## ■ Form Events

### 1. blur

```
$( "#target" ).blur(function() {  
    // code  
});
```

### 2. .change

```
$( "#target" ).change(function() {  
    // code  
});
```

### 3. .focus

```
$( "#target" ).focus(function() {  
    // code  
});
```

### 4. focusin

```
$( "#target" ).focusin(function() {  
    // code  
});
```

### 5. focusout

```
$( "#target" ).focusout(function() {  
    // code  
});
```

### 6. select

```
$( "#target" ).select(function() {  
    // code  
});
```

### 7. submit

```
$( "#target" ).submit(function() {  
    // code  
});
```



## Focus and Blur Vs Focusin and Focusout

HTML :

```
<div class="parent">
  <input type="text" />
</div>
<div class="log"></div>
```

Jquery :

```
function log(str){
    $('#log').append('<div></div>').text(str);
}

$('.parent')
    .focusin(function(){log('div focusin');})
    .focusout(function(){log('div focusout');})
    .focus(function(){log('div focus');})
    .blur(function(){log('div blur');});
$('input')
    .focusin(function(){log('input focusin');})
    .focusout(function(){log('input focusout');})
    .focus(function(){log('input focus');})
    .blur(function(){log('input blur');});
});
```

## 6) DOM Effects

### ■ Hide/Show

1. `hide(duration[,easing][,complete])`

Target element ကို hide ထားချင်တဲ့အခါ သုံးပါတယ်။

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

2. `show(duration[,easing][,complete])`  
Target element ကို show ထားချင်တဲ့အခါ သုံးပါတယ်။

```
$( "button" ).click(function() {  
    $( "p" ).show( "slow" );  
});
```

3. `toggle([duration][,complete])`  
Target element ကို toggle ဖြစ်ချင်တဲ့ အခါ သုံးပါတယ်။

```
$( ".target" ).toggle();
```

## ■ Fade

1. `fadeIn([duration][,easing][,complete])`  
Show ကို ချက်ချင်းပေါ်မလာပဲအချိန်မှတ်ပြီး ပေါ်စေချင်တဲ့အခါ သုံးပါတယ်။

```
$( "a" ).click(function() {  
    $( "div" ).fadeIn( 3000, function() {  
        $( "span" ).fadeIn( 100 );  
    });  
});
```

2. `fadeOut([duration][,easing][,complete])`  
Hide ကို ချက်ချင်းမပျောက်ပဲ အချိန်သက်မှတ်ပြီးပျောက်စေချင်တဲ့အခါ သုံးပါတယ်။

```
$( "button" ).click(function(){  
    $( "#div3" ).fadeOut(3000);  
});
```

3. `fadeTo(duration,opacity[,easing][,complete])`  
လက်ရှိရှိနေတဲ့အခြေအနေကနေ သတ်မှတ်ထားတဲ့ အတိုင်းအတာထိ ပေါ်စေချင်(သို့) ပျောက်စေချင်တဲ့အခါ သုံးပါတယ်။

```
$( "p" ).first().click(function() {  
    $( this ).fadeTo( "slow", 0.33 );  
});
```

4. `fadeToggle(duration,opacity[,easing][,complete])`  
ပေါ်တာပျောက်တာ တလှည့်စီလုပ်စေချင်တဲ့အခါ သုံးပါတယ်။

```
$("#button").click(function(){  
    $("#p").fadeToggle();  
});
```

## ■ Slide

1. `slideDown([duration][,easing][,complete])`  
Target element ကို Silde view (အပေါ်ကနေအောက်) အနေနဲ့ ပေါ်လာစေချင်တဲ့အခါသုံးပါတယ်။

```
$( "#target" ).click(function() {  
    $( "#myid" ).slideDown( "slow", function() {  
        // code for complete.  
    });  
});
```

2. `slideToggle([duration][,easing][,complete])`  
Silde view ကို toggle အနေနဲ့ ပြုချင်တဲ့အခါ သုံးပါတယ်။

```
$( "button" ).click(function() {  
    $( "p" ).slideToggle( "slow" );  
});
```

3. `slideUp([duration][,easing][,complete])`  
Target element ကို Silde view (အောက်ကနေအပေါ်) အနေနဲ့ ပျောက်သွားစေချင်တဲ့အခါသုံးပါတယ်။

```
$( "#target" ).click(function() {  
    $( "#myid" ).slideUp( "slow", function() {  
        // code for complete.  
    });  
});
```

## ■ Animate

1. `animate(properties[,duration][,easing][,complete])`  
One time animation အတွက် အသုံးပြုပါတယ်။

```
$( "#target" ).click(function() {  
    $( "#myid" ).animate({  
        color: "teal",  
        left: "+=50",  
        height: "200px"  
    }, 5000, function() {  
        // Animation complete.  
    });  
});
```

2. `stop([clearQueue][,jumpToEnd])`  
Animation တွေကို stop လုပ်တာ effect တွေကို ရပ်စေချင်တဲ့အခါ  
သုံးပါတယ်။

```
$("#flip").click(function(){  
    $("#panel").slideDown(5000);  
});  
  
$("#stop").click(function(){  
    $("#panel").stop();  
});
```

## 7) Manipulation HTML

### ■ Content

#### 1. text()

Target element တွေထဲကစာသားတွေ ယူချင်တာ elements တွေထည့်ချင်တာ (သို့) စာသားတွေထည့်ချင်တဲ့အခါ သုံးပါတယ်။

Get

```
var ptext = $( "p" ).text();
```

Set

```
$( "p" ).text( "<b>Some</b>" );
```

#### 2. html()

Target element တွေထဲကhtml tag တွေယူချင်တာ elements တွေထည့်ချင်တဲ့အခါသုံးပါတယ်။

Get

```
var phtml = $( "p" ).html();
```

Set

```
$( "p" ).html( "<b>Some</b>" );
```

#### 3. val()

Input box , button တွေရဲ့value တွေ ယူချင်တဲ့အခါ သုံးပါတယ်။

Get

```
var inputValue = $( "input" ).val();
```

Set

```
$( "input" ).val( "Something" );
```

## ■ Attributes

### 1. attr()

Get

```
var link = $( "a" ).attr("href");
```

Set

```
$( "a" ).attr( "href","https://www.google.com" );
```

### 2. removeattr()

Get

```
$("#button").click(function(){  
    $("#p").removeAttr("style");  
});
```

## ■ Add

### 1. prepend()

Target tag ရဲ့ child tag ရဲ့ ဦးဆုံးမှာ elements  
တွေထည့်ချင်တဲ့အခါသုံးပါတယ်။

```
$("#button").click(function(){  
    $("#p").prepend(" <span>Prepend Text </span>.");  
});
```

### 2. append()

Target tag ရဲ့ child tag ရဲ့ နောက်ဆုံးမှာ elements  
တွေထည့်ချင်တဲ့အခါသုံးပါတယ်။

```
$("#button").click(function(){  
    $("#p").append(" <span>Append Text </span>.");  
});
```

### 3. after()

Target tag ရဲ့ level တူ tag ရဲ့ နောက်ဆုံးမှာ elements တွေထည့်ချင်တဲ့အခါသုံးပါတယ်။

```
$("#button").click(function(){  
    $("#p").before("<span>Before</span>");  
});
```

### 4. before()

Target tag ရဲ့ child tag ရဲ့ ရှေ့ဆုံးမှာ elements တွေထည့်ချင်တဲ့အခါသုံးပါတယ်။

```
$("#button").click(function(){  
    $("#p").after("<span>After</span>");  
});
```

## Prepend and append Vs before and after

```
<div class='a'>  
    <div class='b'>b</div>  
</div>
```

### prepend()

```
$("div.a").prepend("<div class='c'>C</div>");
```

Result:

```
<div class='a'>  
    <div class='c'>c</div>  
    <div class='b'>b</div>  
</div>
```

### append()

```
$("div.a").append("<div class='c'>C</div>");
```

Result:

```
<div class='a'>  
    <div class='b'>b</div>  
    <div class='c'>c</div>  
</div>
```

### before()

```
$("#div.a").before("<div class='c'>C</div>");
```

Result:

```
<div class='c'>c</div>
<div class='a'>
  <div class='b'>b</div>
</div>
```

### after()

```
$("#div.a").after("<div class='c'>C</div>");
```

Result:

```
<div class='a'>
  <div class='b'>b</div>
</div>
<div class='c'>c</div>
```

## ■ Remove

### 1. remove()

Target element ကို ဖျက်ချင်တဲ့အခါ သုံးပါတယ်။

```
$("#button").click(function(){
  $("#div1").remove();
});
```

### 2. empty()

Target element ထဲက elements တွေကို ဖျက်ချင်တဲ့အခါ သုံးပါတယ်။

```
$("#button").click(function(){
  $("#div1").empty();
});
```



## ■ CSS Class

### 1. addClass()

Target element မှာ Css class ကိုထပ်ထည့်ချင်တဲ့အခါ သုံးပါတယ်။

```
$("#button").click(function(){
    $("#h1, h2, p").addClass("myclass");
    $("#div").addClass("myclass ");
});
```

### 2. removeClass()

Target element မှာ Css class ကိုထပ်ထည့်ချင်တဲ့အခါ သုံးပါတယ်။

```
$("#button").click(function(){
    $("#h1, h2, p").removeClass("myclass");
});
```

### 3. toggleClass()

Target element မှာ toggle ကိုထည့်ချင်တဲ့အခါ သုံးပါတယ်။

```
$("#button").click(function(){
    $("#h1, h2, p").toggleClass("blue");
});
```

## ■ CSS Style Properties

### 1. css()

Target element မှာcss property တွေထည့်ချင်တဲ့အခါ သုံးပါတယ်။

```
$("#button").click(function(){
    $("#p").css("color","teal");
    $("#div").css({
        "color": "teal",
        "border": "1px solid black");
});
```

## 8) Traversing

### 1. parent()

Target element ရဲ့ ဦးဆုံးတွေတွဲ element ရဲ့ parent ကို effect ဖြစ်စေချင်တဲ့အခါသုံးပါတယ်။

```
$("#span").parent().css(  
    {  
        "color": "red",  
        "border": "2px solid red"  
    });
```

### 2. parents()

Target element ရဲ့ ဦးဆုံးတွေတွဲ element ရဲ့ parents တိုင်းကို effect ဖြစ်စေချင်တဲ့အခါသုံးပါတယ်။

```
$("#span").parents().css(  
    {"color": "red",  
     "border": "2px solid red"  
    });
```

### 3. find()

Target element ထဲက selector တွေကို ရှာချင်တဲ့အခါသုံးပါတယ်။

```
$("#ul").find("span").css(  
    {  
        "color": "red",  
        "border": "2px solid red"  
    });
```

## 5. JSON

- 1) Why Json?
- 2) Syntax
- 3) Data Types
- 4) Convert
- 5) Javascript Fetch

## 1) Why Json?

- JavaScript Object Notation
- Lightweight data format
- Base on Javascript
- Easy to read and write
- Can be used morden Programming Language

## 2) Syntax

```
1. {  
    "first_name" : "Martin",  
    "last_name" : "Olive",  
    "addresss" :  
        {  
            "city" : "New York",  
            "street" : "5 main St."  
        },  
    "online" : true,  
    "followers" : 987,  
    "Photos": null  
}  
  
2. {  
    "libraryName" : "Book World",  
    "book" : [  
        {  
            "id" : "001",  
            "bookName" : "Java SE",  
            "author" : "John"  
        },  
        {  
            "id" : "002",  
            "bookName" : "Php",  
            "author" : "Mary"  
        }  
    ]  
}
```

## ▪ Syntax Rules

1. Key.value pairs {"name" : "John"}
2. Double quotes
3. Specified Data Types
4. File type must be ".json"

Rule Check>>

<https://jsonlint.com/>

## 3) Data Type

- Number
- String
- Boolean
- Array
- Object
- Null

## 4) Convert

- **JSON.stringify()** javascript object or value to JSON String

Example :

```
var person = {  
    name : "David",  
    age : 30  
}  
person = JSON.stringify(person);  
console.log(person);  
// {"name" : "David" , "age" : 30}
```

- **JSON.parse()** JSON String to Javascript object

Example :

```
var person = '{"name": "David", "age": 30}';  
person = JSON.parse(person);  
  
console.log(person.name);  
// David
```

## 5) Javascript Fetch

```
fetch(url,[option])  
  .then(response => response.json())  
  .then(data => console.log(data));  
  .catch(error => console.log(error));
```

Javascript

```
let getData = () => {  
  fetch('http://www.example.com/file.json')  
    .then(res => res.json())  
    .then(data => console.log(data))  
    .catch(error => console.log(error))  
}  
  
getData();
```

file.json

```
{  
  "employee": {  
    "name": "sonoo",  
    "salary": 56000,  
    "married": true  
  }  
}
```