

jQuery : JavaScript Library

โดย : โนมยิตร อารมณ์สวัสดิ์

E-mail : nn.it@hotmail.com

Tel. 080-992-6565

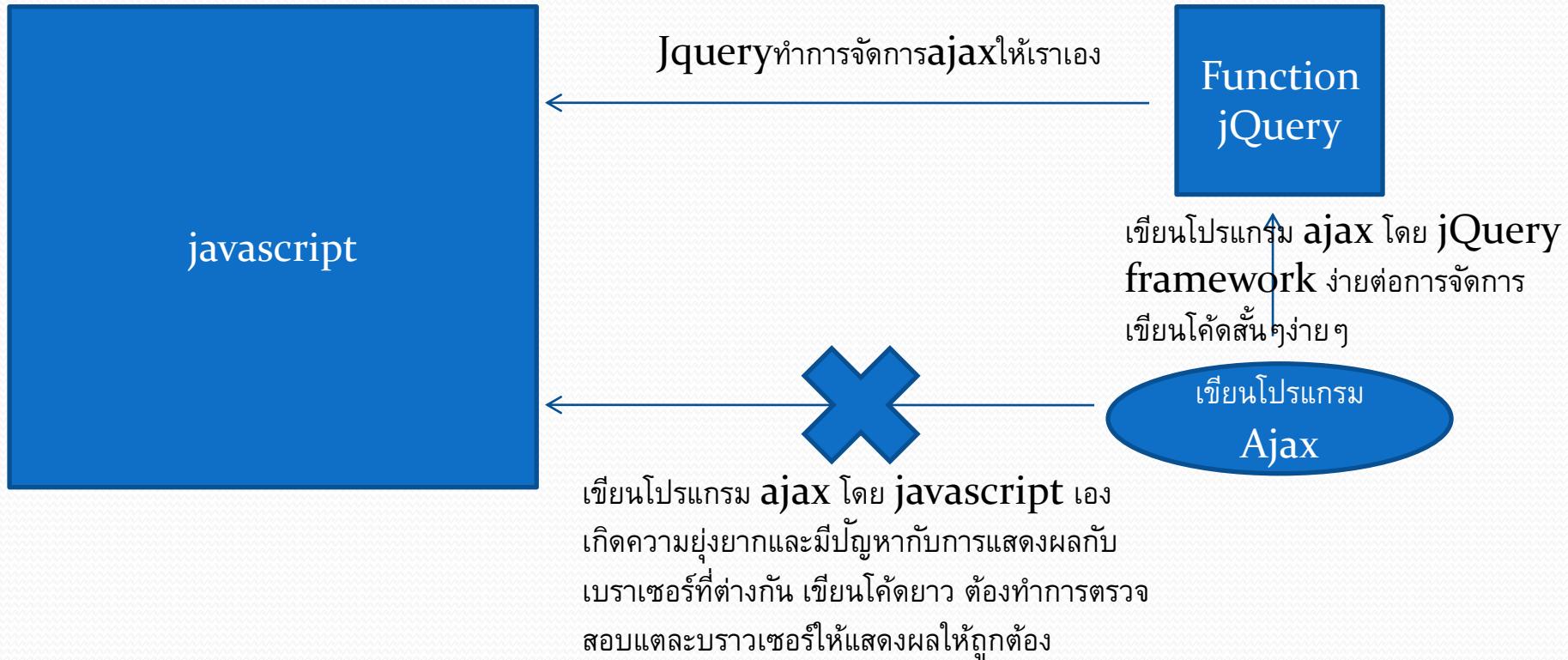
jQuery ?

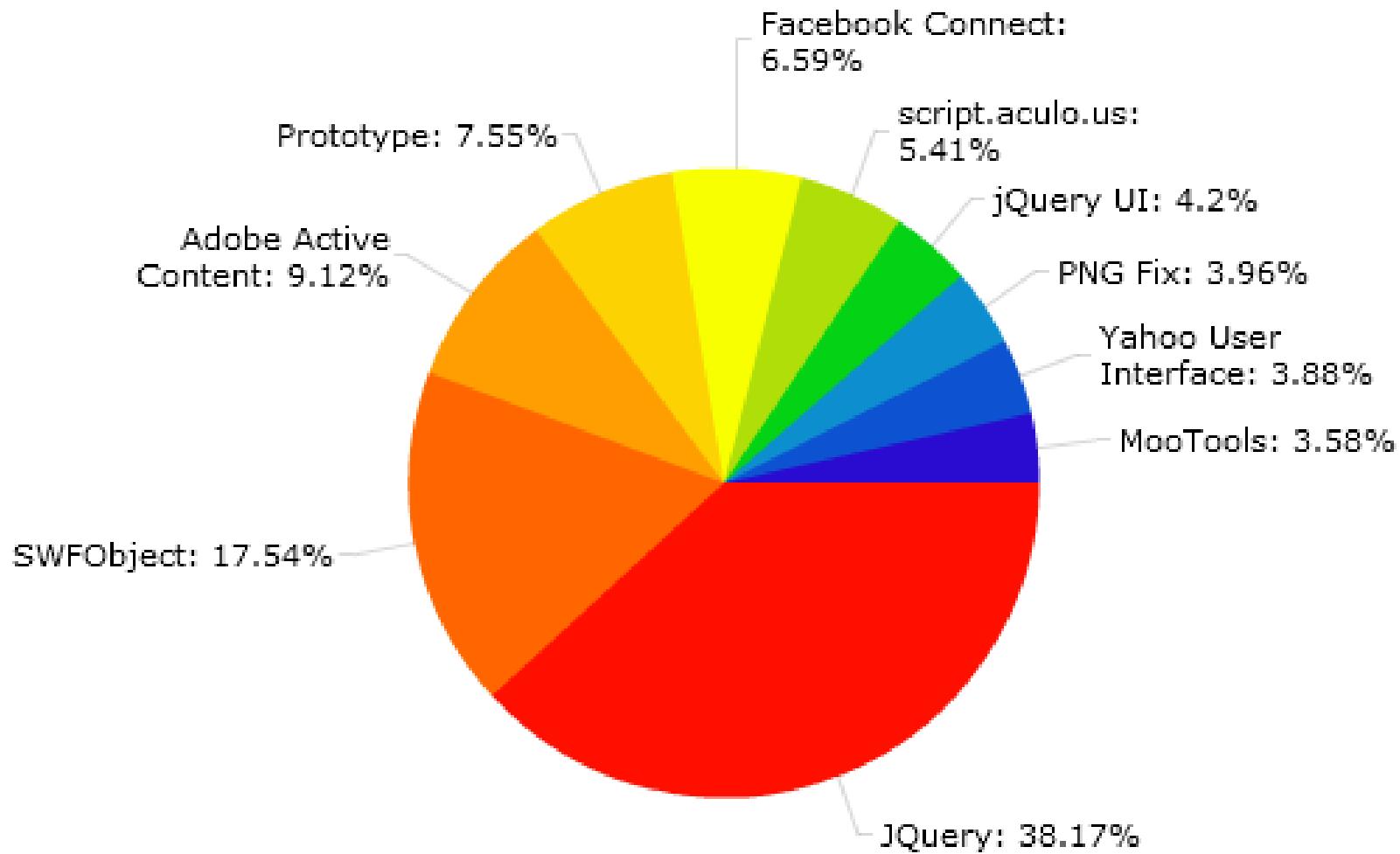
คืออะไร

JavaScript library

- jQuery JavaScript library ก็คือการนำภาษา JavaScript มาสร้างเป็นพังก์ชันใหม่ในรูปแบบของ framework เพื่อให้ง่ายต่อการใช้งาน และไม่ต้องมาเขียนใหม่เองทั้งหมดตั้งแต่ต้น อย่างเช่นการเขียนโปรแกรมในรูปแบบ ajax ก็สามารถทำได้แบบง่ายๆเพียง code ไม่กี่บรรทัด หรือจะเขียนโปรแกรมดัก Event (เหตุการณ์) ต่างๆ เช่น การ click, rollover ก็เขียนได้ง่ายขึ้นและลดปัญหาความไม่เข้ากันของเบราว์เซอร์ที่ต่างกัน

ตัวอย่างการทำงานของ jQuery





ต้องรู้อะไรมาก่อน

เคยใช้ภาษา Javascript หรือไม่ก็ได้

จะเรื่องเบื้องต้น HTML/XHTML/DOM

จะเรื่องเบื้องต้น CSS

จะเรื่องเบื้องต้น Selectors

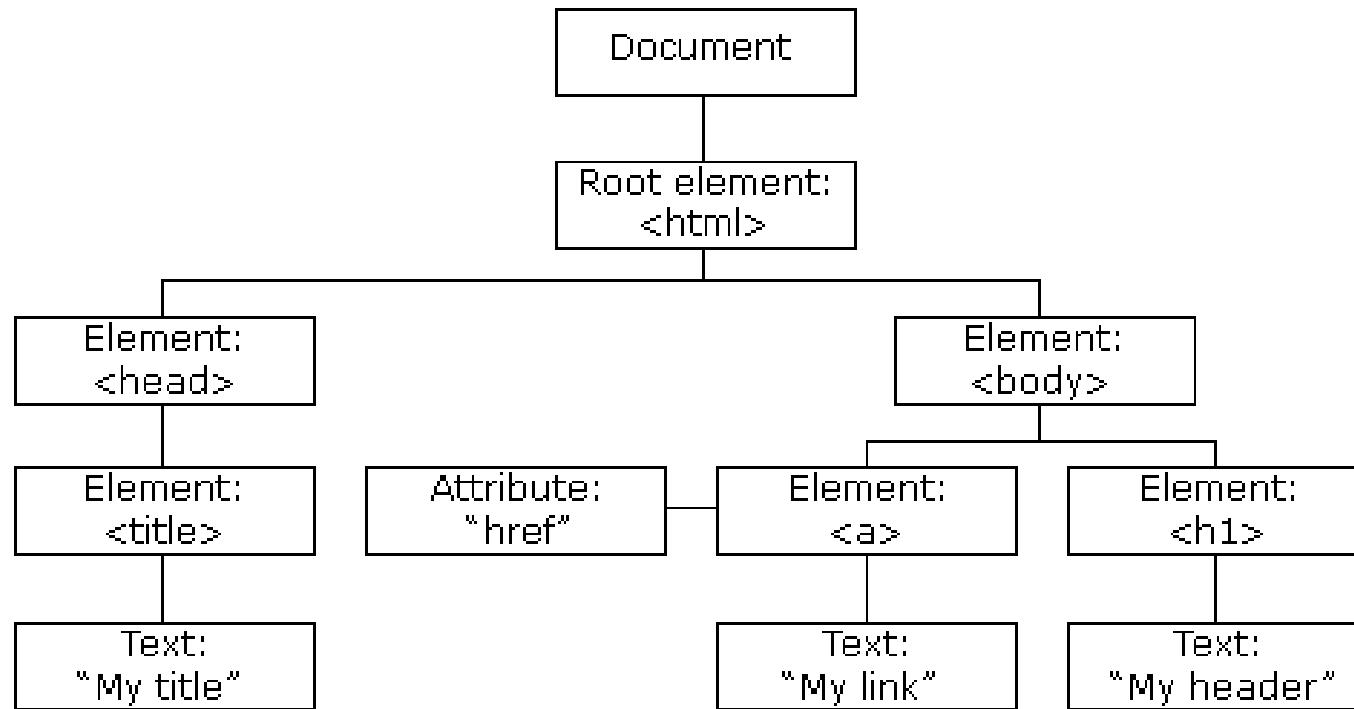
ความสามารถที่สำคัญในไลบรารี jQuery มีดังต่อไปนี้

- 1 เข้าถึงอิลิเมนต์ในเอกสาร (**Selector**)
- 2 การแสดงผลของเว็บ **Cross Browser**)
- 3 แก้ไขคอนเท้นต์ของเอกสาร (**DOM**)
- 4 โต้ตอบกับการทำงานของผู้ใช้ (**Event**)
- 5 แสดงภาพเคลื่อนไหวให้เอกสาร (**Effect**)
- 6 ดึงข้อมูลจากเซิร์ฟเวอร์โดยไม่ต้องรีเฟรช (**Ajax**)

วิจัยโครงสร้าง HTML

```
<html>
  <head>
    <title>My title</title>
  </head>
  <body>
    <h1>My header</h1>
    <a href="test.html">My link</a>
  </body>
</html>
```

Document Object Model(DOM)



โครงสร้างการเขียนjQuery

- เรียกใช้งานไฟล์ jQuery
- เขียนโค้ดสำหรับการโหลด DOM
- เขียนโค้ดโดยการผูก Selector กับ event ที่ต้องการ
- เขียนโค้ดโดยการผูก Selector กับ method ที่ต้องการ

ตัวอย่าง

```
<script type="text/javascript" src="jquery.js"></script>

<script type="text/javascript">
$(document).ready(function(){
    .... Selector + event ....
    .... Selector + method ....
});
</script>
```

เปรียบเทียบ ได้กับ onload ของ JavaScript

- ผูกติดกับเอกสาร html โค้ดไม่เป็นระเบียบ
- เกิดการทำงานช้าๆ ในเพจต่างๆ
- ถูกเรียกในอีเวนต์ (event handlers)ต่างๆ เช่น การคลิ๊ก
- จำเป็นต้องแก้โค้ดหลายๆ เพจ

ทดสอบการใช้งาน jQuery

```
$(selector).method() // or .event()
```

Selector จะบุชื่อ selector ที่ต้องการให้เมธอดนั้นทำงานด้วย method() ที่ต้องการใช้งาน เป็น methode ที่ jQuery จัดเตรียมไว้ให้เรียกใช้งาน

ตัวอย่าง

```
<script type="text/javascript">  
$(document).ready(function(){  
  
    $("#btn1").click(function(){  
  
        alert("Hello jQuery");  
  
        console.log("Hello jQuery");  
  
    });  
  
});  
  
</script>
```

Firebug Download

Selector

เป็นการเลือก Tag หรือ Element เพื่อผูก Event หรือผูก Method หรือกำหนด CSS

ประเภทของ Selector เมื่องต้น

TAG Selector เขียนโดยอ้างชื่อ Tag ของ XHTML

ID Selector เขียนโดยมีเครื่องหมาย # ไว้ด้านหน้า

Class Selector เขียนโดยมีเครื่องหมาย . ไว้ด้านหน้า

Attribute Selector เขียนไว้ระหว่างเครื่องหมาย [....] (กัมปุ)
ตัวอย่าง

`$("p").method(); //Tag Selector`

`$("#btn1").method(); //ID Selector`

`$(".font18").method(); //Class Selector`

`$("[type=radio]").method(); //Attribute Selector`

Attribute Selector

การเปรียบเทียบค่าของ Attribute เพื่อเลือกกระทำได้ สรุปได้ดังนี้

[attr=value]	[attr!=value]	[attr^=value]
[attr\$=value]	[attr*=value]	

[attr=value] เลือก Attribute ที่มีค่าเท่ากับ value เช่น

`$("#[name=inputName1]").`เลือก Attribute name ที่มีค่าเท่ากับ inputName1

`<input type="text" name="inputName1" />`

`<input type="text" name="inputName2" />`

ลองเขียนโค้ดตามหนังสือ

jQuery Event

.bind()

.click()

.hover()

.dbclick()

.focus()

.blur()

.keydown()

.keypress()

.keyup()

.live()

.mousemove()

.mouseover()

.mouseout()

.mouseup()

.mousedown()

.change()

.error()

ตัวอย่าง

```
$("#btn").click(function(){
    alert("Click Event");
});
```

ลองเขียนโค้ดตามหนังสือ

DOM With jQuery

jQuery มี method ในการเข้าถึงข้อมูลต่างๆ กับ Tag หรือข้อมูลที่อยู่ระหว่าง Tag ได้โดยอาศัยหลักการของ DOM

ตัวอย่าง

```
$(“#content”).text();
```

เป็นการเข้าถึงข้อมูลภายใน Tag โดยมองข้อมูลเป็นแบบ Text

```
$(“#content”).html();
```

เป็นการเข้าถึงข้อมูลภายใน Tag โดยมองข้อมูลเป็น HTML

DOM Method (1)

Method ที่ใช้ทำงานกับ DOM มีดังต่อไปนี้

.add()

.children()

.each()

.end()

.eq()

.filter()

.find()

.first()

.last()

.has()

.is()

.next()

.not()

.parent()

.prev()

.sibling()

.slice()

DOM Method (2)

.text()
.html()
.attr()
.after()
.before()
.append()
.css()
.empty()
.insertAfter()
.insertBefore()

.addClass()
.remove()
.removeAttr()
.removeClass()
.toggleClass()
.warp()
.warpAll()

jQuery With CSS (1)

jQuery สามารถเข้าถึงจัดการกำหนดค่าต่างๆ กับ CSS ในแต่ละ Element ได้โดยผ่านเมธอดต่างๆ ที่มีให้

`.css(Properties,Value)`

`.addClass(Css Class)`

`.removeClass(Css Class)`

`.toggleClass(Css Class)`

jQuery With CSS (2)

ตัวอย่างการใช้งาน method เกี่ยวกับ Css

```
$("#content").css("color","#FFoooo");
//เป็นการกำหนดค่าสีให้ตัวอักษรภายใต้ element #content
$("#content").addClass("font24");
//เป็นการกำหนดค่า CSS Class
$("#content").removeClass("font24");
//เป็นการลบค่า CSS Class
$(".btn").click(function(){
    $(this).toggleClass("bgRed");
    //เป็นการกำหนดค่า CSS Class แต่มีการเช็คก่อนว่ามี Class นี้อยู่หรือไม่
    //ถ้าอยู่แล้วจะลบ Class นี้出去 แต่ถ้าไม่มีให้ทำการกำหนดค่า CSS Class แต่ถ้ามี Class นี้อยู่แล้ว
    //ให้ทำการลบ CSS Class
});
```

jQuery With CSS 3

ตัวอย่างการใช้งาน method .css เกี่ยวกับ Css

```
$(document).ready(function(){
    $("table tr:odd").css({"background":"#cccccc"});
    $("table tr:even").css({"background":"#ffccff"});
});
```

jQuery Effect

เป็นการแสดงผลในแบบพิเศษ เช่น การซ่อน/แสดงข้อมูล จัดการข้อมูลในรูปแบบสไลด์ ขึ้น-ลง เป็นต้น

Method effect มีดังนี้

.animate()

.show()

.hide()

.slideDown()

.slideUp()

.toggle()

.slideToggle()

.fadeIn()

.fadeOut()

jQuery Effect animate()

Animate Method

ตัวอย่าง

สำหรับสร้างภาพเคลื่อนไหวแบบกำหนดเอง

```
$("#box").animate({'width':'+50px});
```

ใช้งานกับ css properties

รูปแบบ

```
$(selector).animate();
```

การประยุกต์ใช้งาน

```
$(document).ready(function(){
  $("#b1").click(function(){
    $("#content").animate({'width':'100px;'},function(){
      $(this).animate({'height':'200px'});
    });
  });
});
```

jQuery Effect show(),hide(),toggle()

SHOW/HIDE Method

ช่องและแสดงข้อมูล

`$(selector).show(duration,[callback])`

`$(selector).hide(duration,[callback])`

`$(selector).toggle(duration,[callback])`//สลับการทำงาน

Duration คือ ระยะเวลาในการดำเนินการของ Effect กำหนดได้ 2 รูปแบบคือ

Keyword เช่น **slow,normal,fast**

Number เช่น 1000,2000(หน่วยเป็น milliseconds)

[callback] เป็นฟังก์ชันในการดำเนินการต่อ หลังจากจบการทำงานของ effect

jQuery Effect example hide(),show()

ตัวอย่างการใช้งาน Effect Show/Hide

```
$(“#btn1”).click(function(){
    $(“#box”).hide(“slow”);
    $(“#box”).hide(“normal”);
    $(“#box”).hide(“fast”);
});
```

```
$(“#btn1”).click(function(){
    $(“#box”).show(1000);
    $(“#box”).show(5000);
    $(“#box”).show(10000);
});
```

jQuery Effect slideDown(),slideUp(),slideToggle()

Slide Content

ສ່ໄລດໍ ແສດງ/ຫຼອນ ຂໍອມູດ

.slideDown([duration],[callback])
.slideUp([duration],[callback])
.slideToggle([duration],[callback])

ຕົວຢ່າງ

```
$("#btn1").click(function(){
    $("#box").slideUp("slow");
});
$("#btn1").click(function(){
    $("#box").slideToggle("slow");
});
```

jQuery Effect fadeOut(), fadeIn(), fadeTo()

Fade Effect

สำหรับจัดการข้อมูลให้แสดง Effect แบบค่อยๆ ปรากฏหรือหาย
รูปแบบ

```
$(selector).fadeOut(duration)  
$(selector).fadeIn(duration)  
$(selector).fadeTo(duration,opacity)
```

Duration มีหน่วยเป็น มิลลิวินาที 1000 มิลลิวินาที = 1 วินาที

Opacity กำหนดความเข้มของภาพ ระบุค่าเป็นตัวเลขมีค่าตั้งแต่ 0 – 1

ตัวอย่าง

```
$("#box1").fadeOut(1000);  
$("#box1").fadeTo(1000,0.5);
```

ลองเขียนโค้ดตามหนังสือ

jQuery With Form Input

jQuery สามารถทำงานร่วมกับ Form กรอกข้อมูลได้ เช่น การรับค่าข้อมูล
ตัวอย่าง

การเข้าถึงข้อมูล **Tag input**

Var email = \$("#email").val();

การเข้าถึงค่าโดยผ่าน **method attribute**

Var email = \$("#email").attr("value");

รับค่าจาก input หลายค่า

Var dataininput = \$('input').serialize();

/* กรณีรับค่าจาก input หลายค่าแล้วต้องการส่งไปประมวลผลทั้งหมด
ค่าที่ได้รับเป็นรูปแบบ input1=val1&input2=val2&input3... */

การรับค่าจาก Input

เป็นการรับค่ามาจากฟร์ม client side เพื่อประมวลผลที่ฟร์ม server side

.serialize() เป็น method ในการส่งค่าข้อมูลทั้งหมดหลายๆ ค่าภายใน form ตัวอย่างการส่งข้อมูลไป server โดยใช้ method .serialize();

```
$(“form”).submit(function(){
    var $xxx = $(“input”).serialize();
    $.ajax({
        url:”url.php”,
        type:”get”,
        dataType:”html”,
        data:{“param”:xxx},
        success:function(data){
            alert(data);
        }
    });
});
```

```
<form>
    <input type=“text” id=“name” name=“name” />
    <input type=“text” id=“surname” name=“surname” />
</form>
```

\$.serialize() ข้อมูลที่ถูกส่งไปจะมีค่าเป็น query string คือ k1=v1&k2=v2

Method อื่นๆ ที่ทำงานกับ Form

.blur()	.change()	.focus()	.select()
.serialize()	.serializeArray()	jQuery.param()	.submit()

ตัวอย่าง

```
Var values = ({x1:18,x2:55});  
Var v2 = jQuery.param(values);  
//v2 = x1=18&x2=55  
แปลงข้อมูลจาก json เป็นการ query string
```

การเขียน Function กับ jQuery

```
var functionName = function(a,b){  
    Alert("Total Sum"+(a+b));  
};  
  
functionName(10,20);
```

jQuey WorkShop

ผู้เข้าอบรมสร้าง input สำหรับการกรอกข้อมูล จากนั้น
ให้ทำการสร้าง Action เพื่อรับค่าจาก Input นำมาแสดง
ที่หน้าเว็บเพจ(<div id="result"></div>) โดยการแสดงผลให้มี
การใช้ Effect Method ที่ได้เรียนรู้มา

jQuery With Ajax

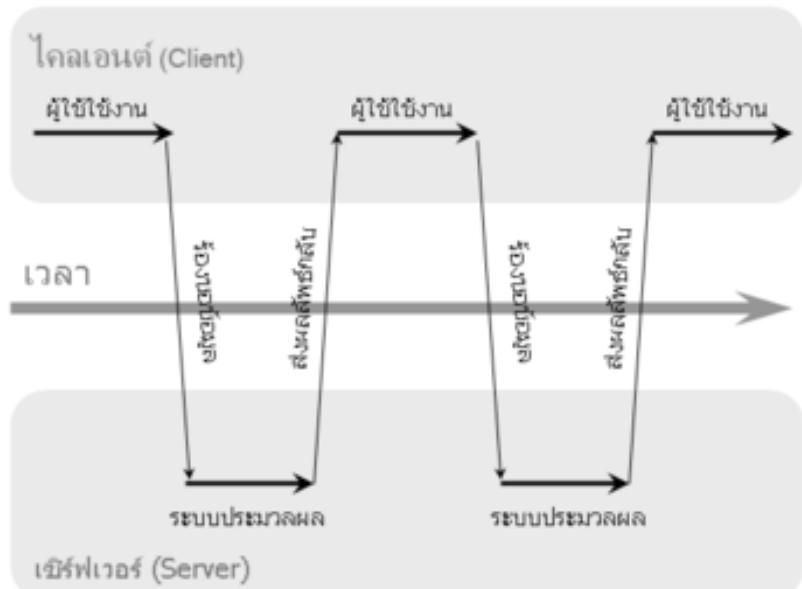
Asynchronous JavaScript And XML

เป็นการนำเอา Client Side Script เชื่อมต่อกับ Server side Script
เช่น

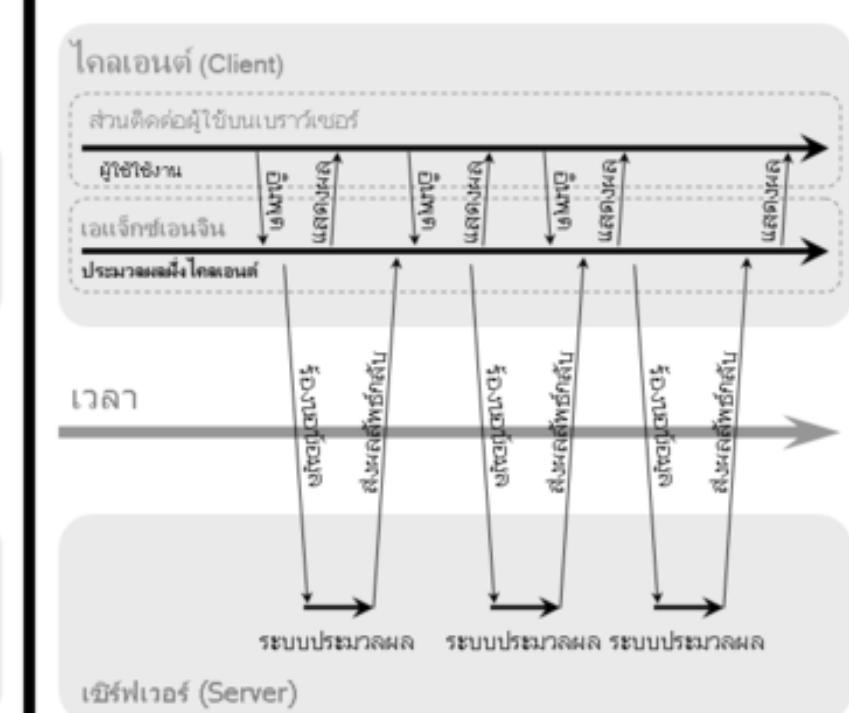
Javascript เรียกใช้งาน PHP, JSP, ASP

- โหลดเนื้อหาข้อมูลที่ต้องการโหลดใหม่
- ลด Data ในการรับ-ส่งข้อมูลระหว่าง Web Application
- ใช้งานผ่าน HTTP Request

Ajax?



เว็บแอปพลิเคชันทั่วไป (synchronous)



เว็บแอปพลิเคชันที่ใช้เอนจิ้นซ์ (asynchronous)

jQuery with Ajax Method

`$.ajax()`

โครงสร้าง

```
$.ajax({  
    'url':'url.php',  
    'type':'get/post',  
    'dataType':html/json/text',  
    success:function(data){
```

```
    .....  
    }  
});
```

<code>\$.post()</code>	<code>\$.post('file.php',function(data){ ... })</code>
<code>\$.get()</code>	<code>\$.get('file.php',function(data){ ... })</code>
<code>\$.load()</code>	<code>\$.load('url.php');</code>

jQuery Ajax with PHP

```
$(document).function({  
    $.ajax({  
        'url':'file.php',  
        'type':'get',  
        'dataType':'html',  
        success:function(data){  
            $('#date').html(data);  
        }  
    })  
});
```

การส่งค่ากับ `$.ajax()` method เป็นการส่งคำสั่งไปในรูปแบบ option
คือส่งไปในระหว่างเครื่องหมาย { ... }

```
file.php  
<?php  
    echo"Date : ",data(d/m/y);  
?>
```

Get and Post 1

ส่งค่า GET /POST โดยใช้ `$.ajax()` method

ส่งโดยผ่าน Method เนพาะแบบ

`.get()` สำหรับส่งค่าในรูปแบบ GET

`.post()` สำหรับส่งค่าในรูปแบบ POST

การส่งค่าในรูปแบบ GET จะเป็นการส่งข้อมูลในรูปแบบ query string

เช่น `k1=v1&k2=v2&k3=v3&....`

การส่งข้อมูลไปประมวลผลทางฟังก์ชัน Server

ตัวอย่าง

```
$.ajax({  
    'url':'process.php',  
    'type':'get',  
    'dataType':'html',  
    'data':"user=admin&pwd=pass123",  
    success:function(data){  
        data managemetn...  
    }  
});
```

`'data':{'k1':'v1','k2':'v2'}`

GET and POST 2

\$.get()

```
$.get("get.php", {user:'amin',pwd:'xxx'},  
      function(){  
        ....  
      }  
);
```

\$.post()

```
$.post("post.php", {user:'admin',pwd:'xxx'},  
       function(){  
         ....  
       }  
);
```

Load Conents

```
<script type="text/javascript">
    $(ducment).ready(function(){
        $("#news").load('news.html');
    });
</script>
<div id="news"></div>
```

เมธอด .load() เอาไว้โหลดข้อมูลจากไฟล์หรือ url เพื่อนำมาวางไว้ภายใต้ selector ที่กำหนด

jQuery with XML

การอ่านเอกสาร XML

```
$(data).find("student").each(function(){
    var $entry = $(this);
    $entry.attr("id");
    $entry.find("name").text();
    $entry.find("age").text();
});
```

```
<?xml verion="1.0" encoding="UTF-8"?>
<students>
    <student id="001">
        <name>Kosit</name>
        <age>27</age>
    </student>
    <student id="002">
        <name>Kongsin</name>
        <age>27</age>
    </student>
</students>
```

jQuery with json

Json Object

```
{"key1": "value1", "key2": "value2"}
```

อ่าน Json Object

```
$(data).each(function(index, entryIndex){  
    entryIndex["objectName"];  
});
```

jQuery with jsonp

แก้ปัญหาร้องขอข้อมูล json ข้ามโดเมน(cross domain)

ผู้ server

```
<?php  
$data={"key1":"value1" , "key2":"value2"};  
Print($_GET['callback'].('.$data.'));  
?>
```

ผู้ client

```
$.ajax({  
    url:"url.php"+"?callback=?",  
    type:"get",  
    dataType:"json",  
    success:function(data){  
        alert(data);  
    }  
});
```