

# JavaScript 3

ผศ. ดร. เกียรติแก้ว ธเนศวร

[kejkaew.tha@mail.kmutt.ac.th](mailto:kejkaew.tha@mail.kmutt.ac.th)

## ตารางเรียน

สัปดาห์	วันที่	หัวข้อ/รายละเอียด	การบ้าน (ผู้ตรวจ)	ส่ง Assignment
1	8/8/2566	Introduction to Web programming, Basic HTML 1	Assignment 1 (KT,KN)	1
2	15/8/2566	Basic HTML 2 + CSS 1	Assignment 2 (KT,KN)	1,2
3	22/8/2566	CSS 2	Assignment 3 (KT)	1,2,3
4	29/8/2566	Responsive Web design	Assignment 4 (KT,KN)	2,3,4
5	5/9/2566	JavaScript 1 + DOM	Assignment 5 (KT,KN)	3,4,5
6	12/9/2566	JavaScript 2 + GitHub	Assignment 6 (KT)	4,5,6
7	19/9/2566	JavaScript 3	Assignment 7 (KT,KN)	5,6,7
8	26/9/2566	AJAX + JSON	Assignment 8 (KT,KN)	6,7,8
9	3/10/2566	Node.js 1	Assignment 9 (KT)	7,8,9
	10/10/2566	No class	ส่ง portfolio web page (KT)	
10	17/10/2566	Node.js 2	Assignment 10 (KT,KN)	8,9,10
11	24/10/2566	Node.js 3	Assignment 11 (KT,KN)	9,10,11
12	31/10/2566	Node.js 4 + MySQL	Assignment 12 (KT)	10,11,12
13	7/11/2566			10,11,12
14	14/11/2566			11,12
15	21/11/2566			
16	28/11/2566	ส่งโปรเจค (9:00-16:00)	KT	
		สอบปลายภาค ในตาราง	KT	

## ส่ง portfolio web page

---

- กำหนดส่งวันที่ 12 ต.ค. 2566 ก่อน 23:55 น.
- ส่งโดยอัปโหลด zip file ลง LEB2 และพิมพ์ URL link ของ website ของนศ (เช่น GitHub) ตอนส่งใน LEB2 ด้วย

## ขั้นตอนการ zip file

---

1. นำ website files ทั้งหมดใส่ใน folder เดียวกัน
2. Compress file โดยใช้ ZIP compression
3. อัป zip file ลง LEB2 ที่กำหนดให้ พร้อมทั้งใส่ URL link ของ website ของเรา
4. ส่งโดยดูรายละเอียดตามหน้าถัดไป
5. อย่าลืม references (แหล่งอ้างอิงทั้งหลาย) ถ้าใครไปเอา source codes, template หรือ เนื้อหาบางอย่าง มาจากที่อื่น ควรใส่แหล่งอ้างอิง

Portfolio web page (อยู่ใน index.html) ของนักศึกษาแต่ละคน มีรายละเอียดข้อมูลดังนี้

- มีชื่อ - นามสกุล
- งานอดิเรก
- ทักษะ ความสามารถ
- ความสนใจ
- รูป
- มี Link ไปเว็บอื่น เช่น Facebook, Instagram ของเรา
- หรือรายละเอียดอื่น ๆ (ข้อมูลส่วนตัวเช่น เลขบัตรประชาชน ไม่เอานะ!)

ใน project.html เนื้อหาของเพจเป็นเรื่องเกี่ยวกับ project ต่าง ๆ ที่เราเคยทำหรือคิดว่าจะทำ มีมากกว่า 1 อย่างก็ได้

- เช่น senior project ที่เรากำลังทำ หรือ project จากที่เคยทำในห้องเรียน
- ทำ link เชื่อมระหว่าง portfolio (index.html) กับ project (project.html) ด้วย

HTML ทั้ง 2 files ใช้ CSS file อันเดียวกัน และ ทั้ง 2 HTML files ต้องมี CSS แบบ external CSS นะ

ใน portfolio HTML file มี link ไปหา หน้า page ที่เราทำ assignment  
ในห้องเรียน ได้แก่

- Cinnamon Roll Wreath with Apples
- Blog
- Register and login
- Square game
- ทั้งหมดนี้ ต้องมี CSS และ JavaScript แบบ external นะคะ

## คำแนะนำ

---

- การเรียก path ไปหารูป หรือ files ต่าง ๆ ให้ใช้แบบ relative file path โดยอ้างอิงจากตำแหน่ง HTML file นั้น
- ใช้ CSS หรือ JavaScript framework อะไรก็ได้
- **ไม่ให้ใช้ Content management system (CMS) หรือ website builder**



## เกณฑ์การให้คะแนน (15 คะแนน)

รายละเอียด	คะแนน
มีรายละเอียดงานต่าง ๆ ทั้งหมดครบถ้วน	5 คะแนน
แยก CSS และ JavaScript files ได้ถูกต้อง	6 คะแนน
Responsive design	2 คะแนน
ความสวยงาม	2 คะแนน
ไม่ได้ใส่แหล่งอ้างอิง (ถ้าจำเป็นต้องใส่ เช่น Cinnamon Roll หรือใช้ template จากที่อื่น)	-1 คะแนน/ต่อจุด
คำหยาบ ทุกประเภท + รูปที่ไม่เหมาะสม	ได้ 0 ทันที

# เราเรียนอะไรแล้วบ้าง

---

- HTML
- CSS
- JavaScript
  - Variable, control flow, array, event handler
  - Unobtrusive JavaScript
  - Form

# วันนี้เราจะเรียนเรื่อง

---

- Timer
- Create new elements or nodes
- JavaScript events (object)
- สอบ Quiz 5 คะแนน

## Setting a timer

method	description
<code>setTimeout(function,delayMS);</code>	arranges to call given function after given delay in ms
<code>setInterval(function,delayMS);</code>	arranges to call function repeatedly everydelayMSms
<code>clearTimeout(timerID);clearInterval(timerID);</code>	stops the given timer

- ทั้ง `setTimeout` และ `setInterval` ID จะ return ID สำหรับ timer
- เราสามารถใช้ ID ตัวนั้นในการ `clearTimeout/Interval` เพื่อหยุดเวลาได้

## ตัวอย่าง

HTML

```
<span id="output"></span>
<button id="clickme">Click me!</button>
```

Hello!

JS

```
window.onload = pageLoad;
function pageLoad() {
    document.getElementById("clickme").onclick= delayedMessage;
}

function delayedMessage() {
    document.getElementById("output").innerHTML= "Wait for it...";
    setTimeout(sayHello, 5000);
}

function sayHello() { // called when the timer goes off
    document.getElementById("output").innerHTML= "Hello!";
}
```

## ตัวอย่าง

JS

```
16 window.onload = pageLoad;
17 var timer = null; // stores ID of interval timer
18
19 function pageLoad() {
20     document.getElementById("clickme").onclick= delayMsg2;
21 }
22
23 function delayMsg2() {
24     if (timer === null) {
25         timer = setInterval(showMDT, 1000);
26     } else {
27         clearInterval(timer);
28         timer = null;
29     }
30 }
31 function showMDT() { // called each time the timer goes off
32     document.getElementById("output").innerHTML += " MDT!";
33 }
```

# Example

---

MDT! MDT! MDT! MDT! MDT! MDT! MDT! MDT!

## Passing parameters to timers

- เราสามารถส่ง parameter ผ่านทาง timer function ได้ โดยการเพิ่ม argument ใน timer function

JS

```
35 // timer and parameters
36 window.onload = pageLoad;
37
38 function pageLoad() {
39     document.getElementById("clickme").onclick= delayedMultiply;
40 }
41
42 function delayedMultiply() {
43     // 6 and 7 are passed to multiply when timer goes off
44     setTimeout(multiply, 2000, 6, 7);
45 }
46 function multiply(a, b) {
47     alert(a * b);
48 }
```



# Example

---

This page says

42

OK

# The six global DOM objects

---

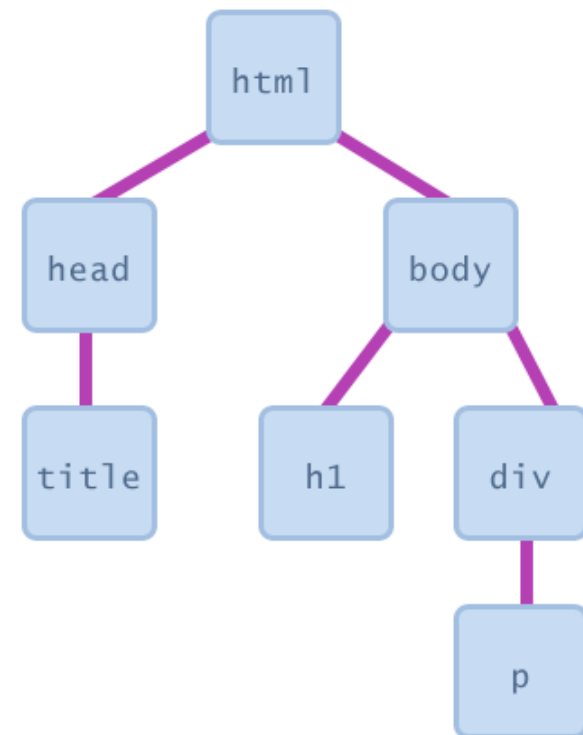
- Every JavaScript program can refer to the following global objects

name	description
document	current HTML page and its content
history	list of pages the user has visited
location	URL of the current HTML page
navigator	info about the web browser you are using
screen	info about the screen area occupied by the browser
window	the browser window

# The DOM tree

---

```
<html>
  <head>
    <title> ... </title>
  </head>
  <body>
    <h1> ... </h1>
    <div>
      <p> ... </p>
    </div>
  </body>
</html>
```



## เมื่อเราต้องการสร้าง element หรือ node ใหม่

- ทำไมเราถึงไม่ควรทำแบบนี้?

HTML

```
<div id="main"></div>
```

A paragraph!

JS

```
window.onload = pageLoad;

function pageLoad() {
    addParagraph();
}

function addParagraph() {
    document.getElementById("main").innerHTML+=
        "<p>A paragraph!</p>";
}
```

- สมมติว่า element ที่เราจะเพิ่มเข้าไปมันดูวุ่นวายมาก ดังตัวอย่าง
- มีโอกาสเกิดความผิดพลาดจากการเขียน code ได้สูงมาก

```
55 function addParagraph() {  
56     document.getElementById("main").innerHTML += "<p style='color: red;  
    margin-left: 50px';> A paragraph!</p>";  
57 }
```

A paragraph!

## การสร้าง nodes หรือ elements ใหม่

- เราจะทำการสร้าง node ใหม่ ก่อนที่เราจะเพิ่ม node นั้นลงใน document
- หลังจากสร้างแล้ว เราก็ต้อง add ลงไปใน document ด้วย

name	description
<code>document.createElement("tag")</code>	creates and returns a new empty DOM node representing an element of that type
<code>document.createTextNode("text")</code>	creates and returns a text node containing given text

## การเพิ่ม/ลด nodes ใน document

- เราสามารถเพิ่ม/ลดโดยใช้ method ต่อไปนี้

name	description
appendChild(node)	places given node at end of this node's child list
insertBefore(new,old)	places the given new node in this node's child list just before old child
removeChild(node)	removes given node from this node's child list
replaceChild(new,old)	replaces given child with new node

## Example:

---

JS

```
59 function addParagraph(){  
60     var p = document.createElement("p");  
61     p.innerHTML= "A paragraph!";  
62     document.getElementById("main").appendChild(p);  
63 }
```

A paragraph!



## การกำหนด Style โดยใช้ DOM object

---

- The Style object represents an individual style statement.
- Accessing a specified element's style object:

JS

```
function addParagraph(){  
    var p = document.createElement("p");  
    p.innerHTML= "A paragraph!";  
    document.getElementById("main").appendChild(p);  
    document.getElementById("main").style.color = "red";  
}
```

A paragraph!

- Set margin-left

JS

```
function addParagraph(){  
    var p = document.createElement("p");  
    p.innerHTML= "A paragraph!";  
    document.getElementById("main").appendChild(p);  
    document.getElementById("main").style.color = "red";  
    document.getElementById("main").style.marginLeft = "50px";  
}
```

A paragraph!

# Selecting groups of DOM objects

---

methods in document and other DOM objects (\* = HTML5):

name	description
getElementsByTagName	returns array of descendents with the given tag, such as "div"
getElementsByName	returns array of descendents with the given name attribute (mostly useful for accessing form controls)
querySelector*	returns the first element that would be matched by the given CSS selector string
querySelectorAll*	returns an array of all elements that would be matched by the given CSS selector string

## ตัวอย่าง

```
<p>This is my add</p>
<div id="address">
  <p>1234 Street</p>
  <p>Atlanta, GA</p>
</div>
```

HTML

```
68 window.onload = pageLoad;
69 function pageLoad(){
70     var addrParas = document.querySelectorAll("p");
71     //var addrParas = document.querySelectorAll("#address p");
72     for (var i= 0; i< addrParas.length; i++) {
73         addrParas[i].style.backgroundColor= "yellow";
74     }
75 }
```

JS

This is my add

1234 Street

Atlanta, GA

Hello World,

## ตัวอย่าง

```
<p>This is my add</p>
<div id="address">
  <p>1234 Street</p>
  <p>Atlanta, GA</p>
</div>
```

HTML

```
67 // Selecting groups of DOM objects
68 window.onload = pageLoad;
69 function pageLoad(){
70     // var addrParas = document.querySelectorAll("p");
71     var addrParas = document.querySelectorAll("#address p");
72     for (var i= 0; i< addrParas.length; i++) {
73         addrParas[i].style.backgroundColor= "yellow";
74     }
75 }
```

JS

This is my address

1234 Street

Atlanta, GA

## Getting/setting CSS classes

---

- JS DOM's **className** property ใช้สำหรับ **get หรือ set CSS class**
- JS DOM's **classList** property ใช้สำหรับ **เช็ค เพิ่ม ลด classes**
- **classList** เหมือนกับ **className** DOM property, แต่ **classList** สามารถใส่ class ได้หลายตัวโดยไม่ต้องกำหนดให้ แต่ละ class ทำการเว้นวรรค

## classList property

Method	Description
<code>add(class1, class2, ...)</code>	Adds one or more class names to an element.
<code>contains(class)</code>	Returns a Boolean value (True/False), indicating whether an element has the specified class name
<code>remove(class1, class2, ...)</code>	Removes one or more class names from an element.
<code>toggle(class, true false)</code>	Toggles between a class name for an element.

## ตัวอย่าง

HTML

```
<button id="clickme">Click me!</button>  
<div id="main"> </div>
```

CSS

```
.highlight{  
  color:pink;  
  font-size: 30px;  
  background-color: green;  
}
```

Click me!

A paragraph!



## classList.toggle(): ตัวอย่าง

```
77 // CSS classes
78 function addParagraph(){
79     var p = document.createElement("p");
80     p.innerHTML= "A paragraph!";
81     document.getElementById("main").appendChild(p);
82     document.getElementById("main").style.color = "red";
83     document.getElementById("main").style.marginLeft = "50px";
84 }
85
86 window.onload = pageLoad;
87
88 function pageLoad() {
89     addParagraph();
90     var x = document.getElementById("clickme");
91     x.onclick = highlightField;
92 }
93
94 function highlightField(){
95     var text = document.getElementById("main");
96     text.classList.toggle("highlight")
97 }
```

## className: จากตัวอย่างเดิม

```
function addParagraph(){
    var p = document.createElement("p");
    p.innerHTML= "A paragraph!";
    document.getElementById("main").appendChild(p);
    document.getElementById("main").style.color = "red";
    document.getElementById("main").style.marginLeft = "50px";
}

window.onload = pageLoad;

function pageLoad() {
    addParagraph();
    var x = document.getElementById("clickme");
    x.onclick = highlightField;
}

function highlightField(){
    var text = document.getElementById("main");
    if (!text.className){
        text.className = "highlight";
    }else if (!text.classList.contains("invalid")){
        text.className = "";
    }
}
```

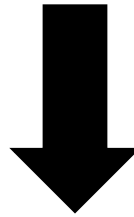
## ตัวอย่าง classList add

```
.highlight2{  
    text-align: center;  
    width: 50%;  
}
```

```
function highlightField(){  
    var text = document.getElementById("main");  
    if (!text.className){  
        text.className = "highlight";  
    }else if (!text.classList.contains("invalid")){  
        text.classList.add("highlight2");  
    }  
}
```

Click me!

A paragraph!

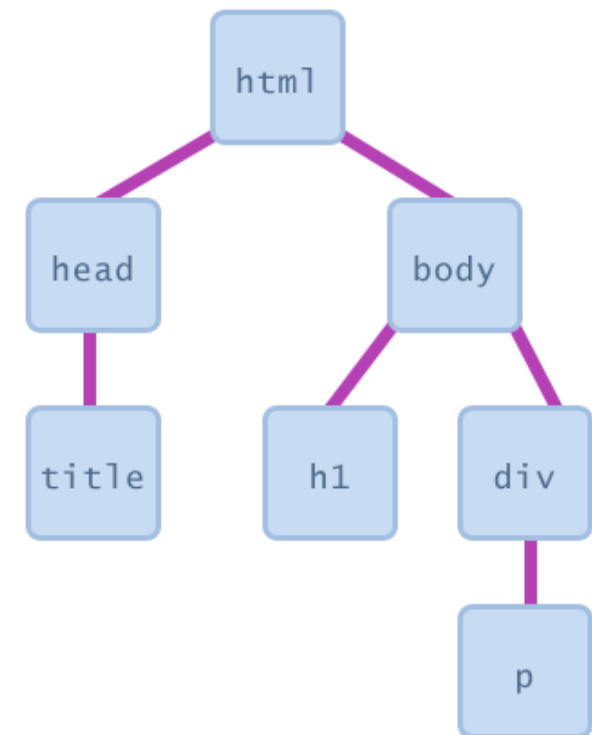


Click me!

A paragraph!

## จาก The DOM tree

```
<html>
  <head>
    <title> ... </title>
  </head>
  <body>
    <h1> ... </h1>
    <div>
      <p> ... </p>
    </div>
  </body>
</html>
```



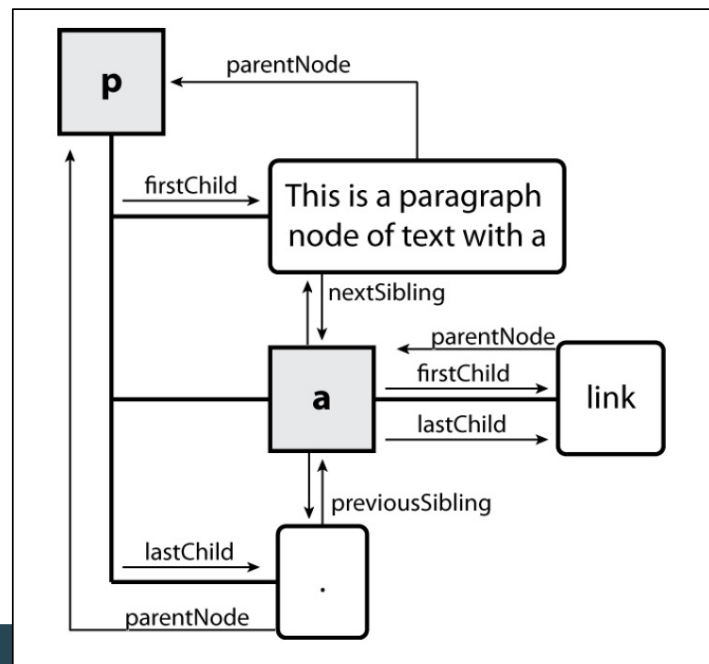
## Traversing the DOM tree manually

ทุก node ของ DOM object จะมี properties ต่อไปนี้ สำหรับอ้างอิงถึง node อื่นๆ

<b>name(s)</b>	<b>description</b>
firstChild, lastChild	start/end of this node's list of children
childNodes	array of all this node's children
nextSibling, previousSibling	neighboring nodes with the same parent
parentNode	the element that contains this node

## ตัวอย่าง DOM tree properties

`<p id="foo">This is a paragraph of text with a  
<a href="/path/to/another/page.html">link</a>.</p>`



# Removing a node from the page

- `obj.parentNode.removeChild(obj);`

HTML

```
<button id="clickme2">Click me!</button>
<div id="main2">
  <h2 id="removeMe"> Hello World, </h2>
</div>
```

JS

```
110 // removing a node
111 window.onload = pageLoad;
112
113 function pageLoad() {
114   var x = document.getElementById("clickme2");
115   x.onclick = removeMe;
116 }
117 function removeMe(){
118   var x = document.getElementById("removeMe");
119   x.parentNode.removeChild(x);
120 }
```

Click me!

**Hello World,**



## Check whether HTML element exist

- Using `Node.contains()` to check whether HTML element exist

```
122 function removeMe(){
123     var x = document.getElementById("removeMe");
124     if (document.body.contains(x)){
125         x.parentNode.removeChild(x);
126     }else{
127         var h2 = document.createElement("h2");
128         h2.innerHTML= "Hello World,";
129         h2.id = "removeMe";
130         document.getElementById("main2").appendChild(h2);
131     }
132 }
```

## JavaScript events (object)

---

- The event object

```
function name(event) {
```

```
    // an event handler function ...
```

```
}
```

property name	description
type	what kind of event, such as "click" or "mousedown"
target	the element on which the event occurred
timeStamp	when the event occurred

# Mouse events

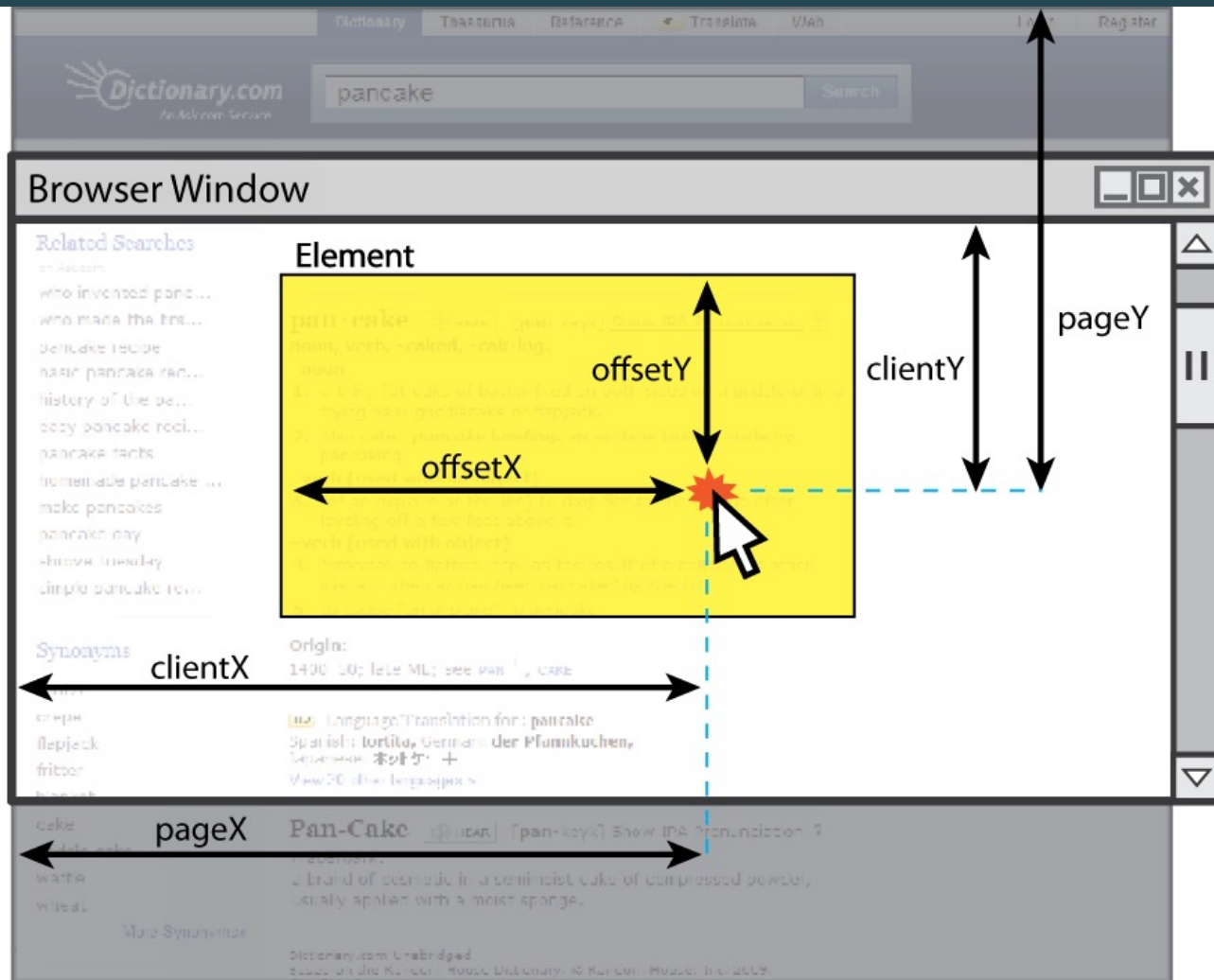
<b>click</b>	user presses/releases mouse button on the element
<b>dblclick</b>	user presses/releases mouse button twice on the element
<b>mousedown</b>	user presses down mouse button on the element
<b>mouseup</b>	user releases mouse button on the element

<b>mouseover</b>	mouse cursor enters the element's box
<b>mouseout</b>	mouse cursor exits the element's box
<b>mousemove</b>	mouse cursor moves around within the element's box

## Mouse event objects

- The event passed to a mouse handler has these properties:

<b>property/method</b>	<b>description</b>
<b>clientX, clientY</b>	coordinates in <i>browser window</i>
<b>screenX,screenY</b>	coordinates in <i>screen</i>
<b>offsetX, offsetY</b>	coordinates in <i>element</i> (non-standard)
<b>pageX, PageY</b>	coordinates in <i>webpage</i>
<b>button</b>	integer representing which button was pressed (0=Left, 1=Middle, 2=Right)



# ตัวอย่าง

screen : (781, 535) client : (699, 250) button : 0

HTML

```
<div id="target">Move the mouse over me!</div>
```

JS

```
window.onload= pageLoad;

function pageLoad() {
    var target = document.getElementById("target");
    target.onmousemove= target.onmousedown= showCoords;
};

function showCoords(event) {
    document.getElementById("target").innerHTML =
        "screen : (" + event.screenX+ ", " + event.screenY+ ")\n"+
        "client : (" + event.clientX+ ", " + event.clientY+ ")\n" +
        "button : " + event.button;
}
```

## สอบ Quiz วันนี้

---

- สอบ 5 คะแนน
- ห้ามคุยกับเพื่อน ถาม บอก ใดๆ ทั้งนั้น ในทุกช่องทางสื่อสาร
- เหมือนกัน ได้ 0 ทั้งคนลอกและให้ลอก
- เปิด slide ได้ทุกอย่าง
- เริ่มสอบหลังจากเรียน
- ใครสอบที่หลังจะได้คะแนน 40% จากคะแนนที่ได้

## Assignment 7:

---

- Download assignment7.zip file
- ทำการสร้าง square game โดยในเกมนี้ เราจะต้องกำหนดก่อนว่าเราจะให้มีสีเหลี่ยมทั้งหมดกี่อัน และสีของสีเหลี่ยม เมื่อเรากด Start แล้ว เราจะต้องกดลบสีเหลี่ยมทั้งหมด ภายในเวลา 30 วินาที ถึงจะชนะในเกมนี้
- การบ้าน ทำการตกแต่ง square game ของเรา ให้สวยงาม



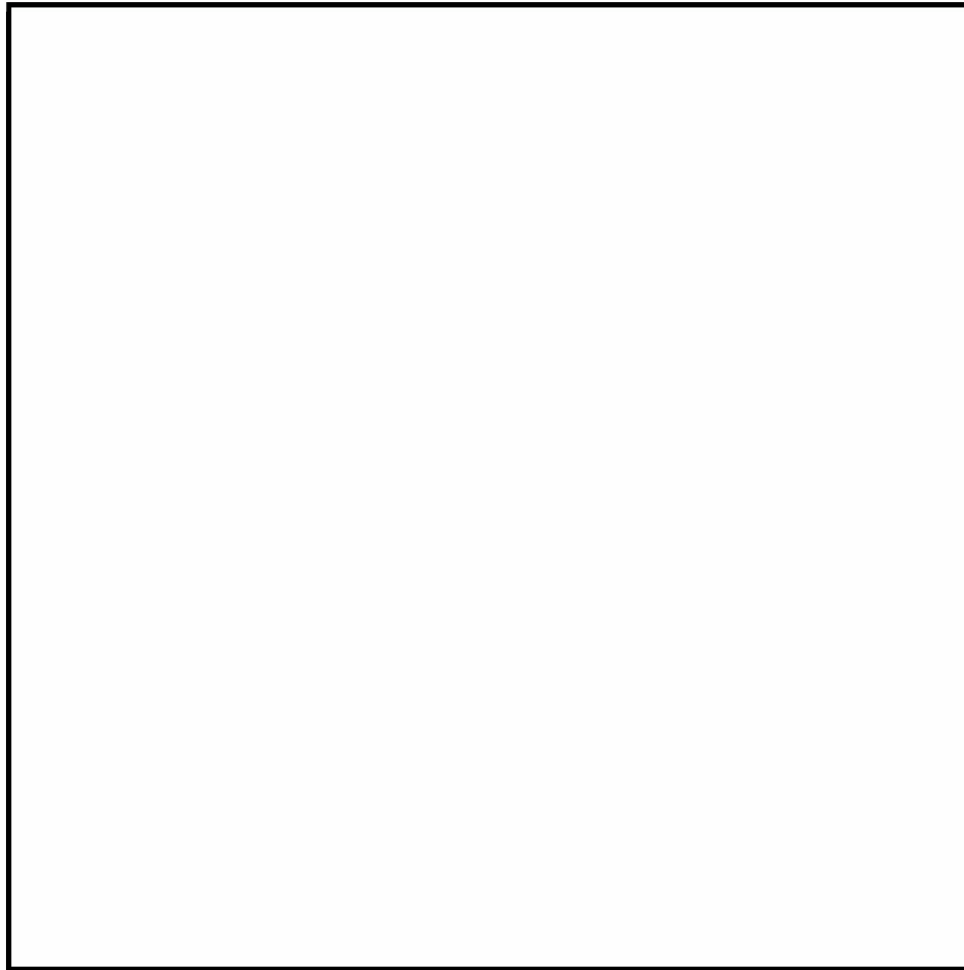
Control

Number of box: |

Maroon ▾

Start

Timer:



## Week 7 : Classroom game

---

- อย่าลืมทำ classroom game ใน LEB2

# References

---

- <https://www.w3schools.com/>
- <https://courses.cs.washington.edu/courses/cse154/17sp/>
- *Web Programming Step by Step* by Marty Stepp, Jessica Miller, and Victoria Kirst, <http://www.webstepbook.com/>