

# IS216 การพัฒนาโปรแกรมประยุกต์โดยผู้ใช้

หัวข้อ

การสร้างเว็บเพจโดยใช้ HTML5

และ CSS ขั้นต้น

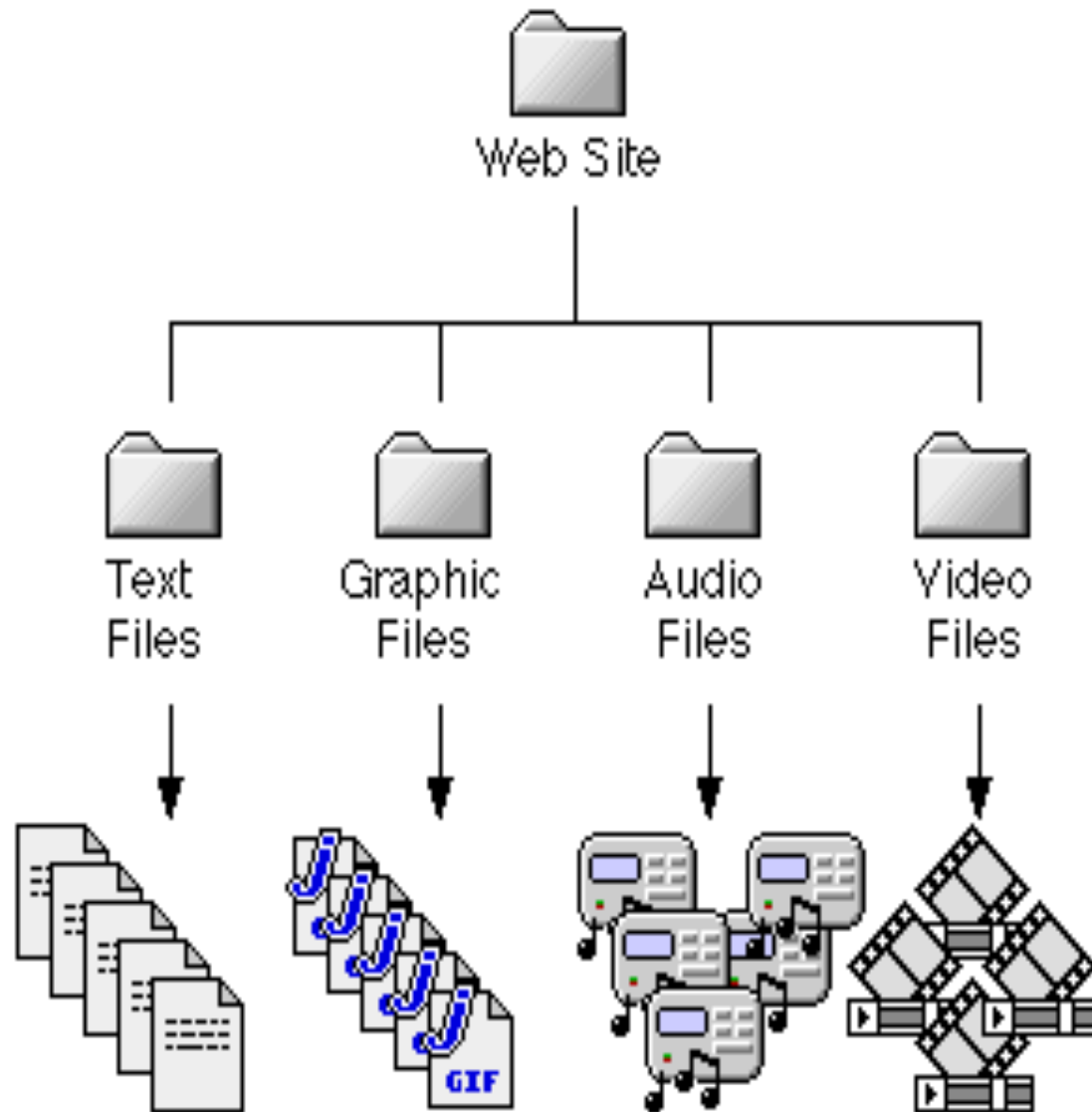
Part IV Multimedia, Table และ อื่น ๆ

ผศ.วันชัย ขันดี wanchai@tbs.tu.ac.th

คณะพาณิชยศาสตร์และการบัญชี มธ.

<http://www.bus.tu.ac.th/usr/wanchai/IS216>

# Multimedia Web pages



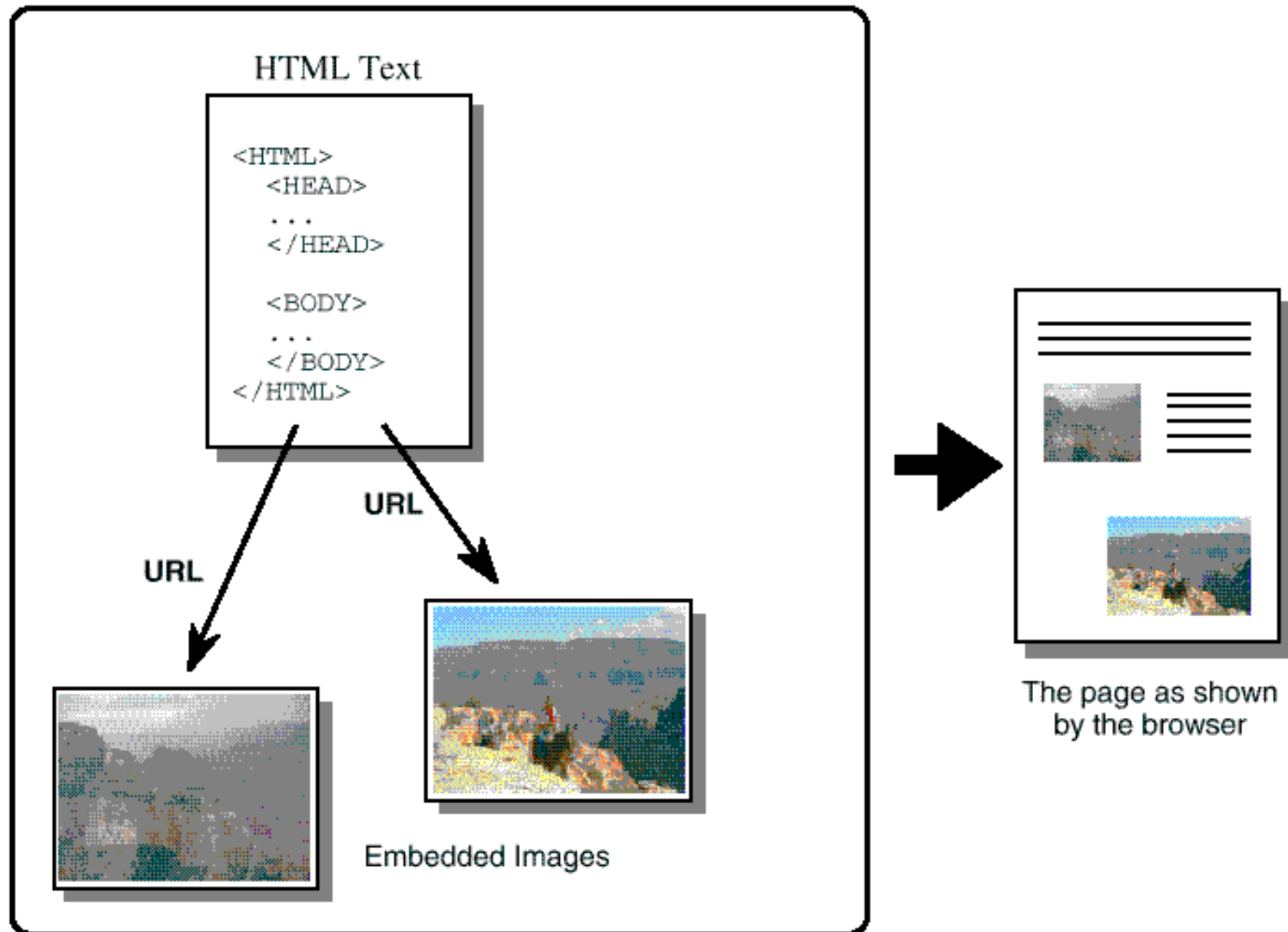
นอกจากใส่ข้อความ  
ในหน้าเว็บแล้ว เรา  
อาจใส่เนื้อหาที่เป็น  
สื่อประสม (multimedia  
content) ลงไปได้ด้วย

# Multimedia Contents

เนื้อหาที่เป็นสื่อประสม (multimedia content) เช่น

- รูปภาพ ( format ที่เบราว์เซอร์รู้จักและแสดงผลได้)
- เสียง ( midi, wave file, mp3 , wma, ogg)
- ภาพวีดิทัศน์ ( มีหลาย format เช่น mpg, wmv, mp4, ogg)
- ก่อน HTML5 การสร้างภาพเคลื่อนไหวบนหน้าเว็บต้องใช้ plugin เช่น Adobe Flash, Real Player และต้องเตรียมไฟล์ตามรูปแบบของ plugin นั้น ๆ
- HTML5 รองรับการแสดงผลภาพกราฟฟิกแบบ SVG โดยไม่ต้องใช้ plugin
- HTML5 มี canvas สำหรับเขียนภาพด้วยคำสั่งภาษา JavaScript

# การใช้รูปภาพในเอกสาร HTML



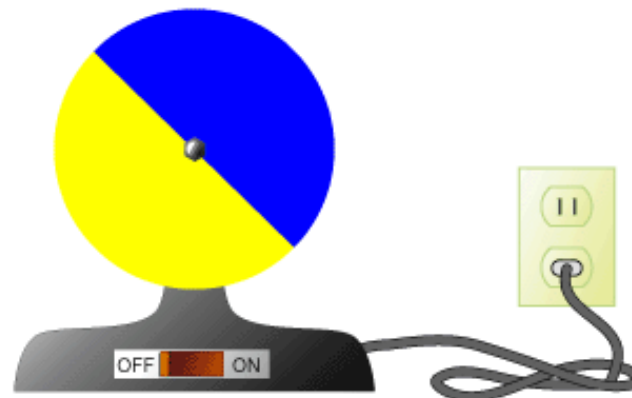
# ชนิดของรูปภาพที่สามารถแสดงในหน้าเว็บ

- JPEG      Joint Photographic Expert Group
- GIF      Graphic Interchange Format
- PNG      Portable Network Graphic

ภาพทั้ง 3 ชนิดนี้เหมาะกับการส่งผ่านทางอินเทอร์เน็ตเพราะสามารถบีบอัดข้อมูลให้เล็กลงได้เพื่อประหยัดเวลาในการส่ง



JPEG



GIF



PNG

# คุณสมบัติของ JPEG

- คุณภาพระดับภาพถ่าย(Photograph-quality)
- 24 บิตต่อจุดภาพ ได้ 16.7 ล้านสี
- บีบอัดได้ดีถ้าใช้กับภาพถ่ายหรือภาพวาดที่มีการไล่เฉดสีเหมือนธรรมชาติขนาดของแฟ้มมักจะเล็กกว่า GIF
- ไม่เหมาะกับภาพลายเส้นเช่นการ์ตูน ขนาดของแฟ้มจะโต
- ไม่เหมาะกับการซ่อนภาพอักษร บีบอัดได้ไม่ดี
- Progressive JPEG files โหลดได้เร็วกว่า JPEG ธรรมดา

สกุลของแฟ้มเป็น *.jpg*

# คุณสมบัติของ GIF

- 8 bit ต่อจุดภาพ (pixel) แสดงได้ 256 สี
- GIF24 ใช้ 24 bit ต่อจุดภาพแสดงได้ 16 ล้านสี
- เหมาะสำหรับภาพที่เป็นสีพื้น ไม่ไล่เฉดสี เช่น โลโก้ ภาพลายเส้น หรือรูปทรงเรขาคณิต
- รองรับการทำให้โปร่งใส (Transparency)
- รองรับการทำ Interlacing
- รองรับการทำภาพเคลื่อนไหว Animation

สกุลของแฟ้มเป็น *.gif*

# คุณสมบัติของ PNG

- สร้างขึ้นมาทดแทน GIF ซึ่งสงวนลิขสิทธิ์ในการใช้
- สามารถแสดงสีได้ 16.7 ล้านสี
- เหมาะสำหรับภาพที่เป็นสีพื้น เช่น โลโก้ ภาพลายเส้น ภาพวาด หรือรูปทรงเรขาคณิต สามารถไล่เฉดสีได้ดีกว่า GIF
- รองรับการทำให้โปร่งใส (Transparency)
- รองรับการทำภาพเคลื่อนไหว (Animation) แต่บางเบราว์เซอร์ไม่รองรับ

สกุลของแฟ้มเป็น *.png*



## การวางรูปภาพในเอกสาร HTML

```
<html>
<head>
<title>Image Placement</title>
</head>
<body>
This is my picture

</body>
</html>
```

# รูปแบบของ HTML สำหรับวางรูปภาพ

- *Inline Image*  

- *hypertext link*  
You can view <a href="mypic.jpg"> my picture</a>.  
my picture จะเป็น link เมื่อคลิกจะนำรูป mypic.jpg มาแสดง
- *hyperlink anchor*  
<a href="target.html"></a>  
รูปภาพ stat.gif จะเป็นลิงค์ เมื่อคลิกจะนำ target.html มาแสดง
- *background image*  
<body background="marble.gif">  
จะนำภาพ marble.gif มาแสดงผลต่อเรียงกันจนเต็มหน้าเอกสาร

# <img /> tag Attributes

- **src** แหล่งที่อยู่ (URL) ของแฟ้มรูปภาพ
- **alt** ข้อความที่ต้องการให้แสดงแทนรูปภาพ
- **border** จำนวน pixel ของเส้นกรอบสี่เหลี่ยมรอบรูป (deprecated)
- **height** กำหนดความสูงของภาพหน่วยเป็น pixel
- **width** กำหนดความกว้างของภาพหน่วยเป็น pixel หรือ %
- **hspace** เว้นที่ว่างด้านซ้ายและขวาของภาพ (pixel) (deprecated)
- **vspace** เว้นที่ว่างด้านบนและล่างของภาพ (pixel) (deprecated)
- **ismap** ทำ image map ประมวลผลที่ server
- **usemap** ทำ image map ประมวลผลที่ browser
- **align** ตำแหน่งที่ต้องการจัดวางรูปภาพ (deprecated)

# Attribute **align** ของแท็ก **img**

- top
- middle
- bottom
- left
- right

*ไม่มี center*

Attribute **align** ถูก Deprecated ใน HTML4 และไม่รองรับใน HTML5

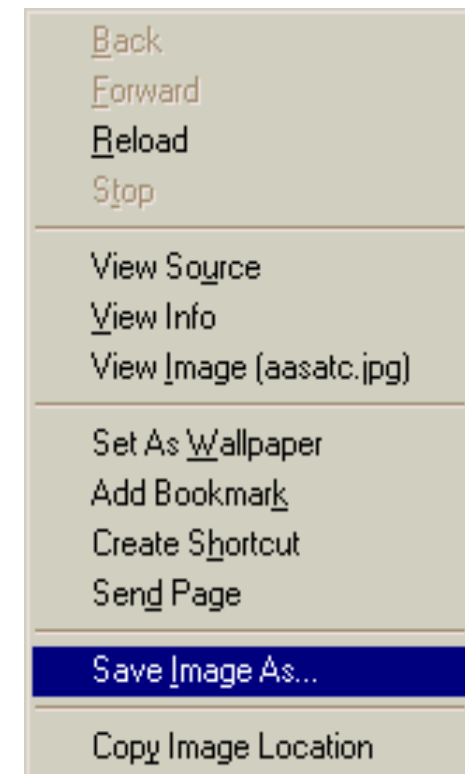
## ตัวอย่างการใช้ **img**

```

```

## การ save ภาพจากเว็บ (สำหรับผู้ที่ไม่ทราบ)

- ใช้เมาส์ชี้ที่รูปภาพแล้วคลิกด้วยปุ่มขวาของเมาส์จะปรากฏ pop up menu
- เลือกรายการ Save Picture As ...
- เลือก drive / folder ที่ต้องการ save แล้วคลิกปุ่ม save



## ฝึกปฏิบัติใส่รูปภาพ

เขียนเว็บให้มีการแสดงรูปภาพลักษณะต่าง ๆ เช่น

- inline image
- hypertext anchor
- background

## ก่อน HTML5

การนำเสนอไฟล์วิดีโอ

```
<img dynsrc= ชื่อไฟล์วิดีโอ  
loop = จำนวนรอบ | infinite />
```

การนำเสนอไฟล์เสียง

```
<bgsound src= ชื่อไฟล์เสียง  
loop = จำนวนรอบ | infinite />
```

หมายเหตุ

1. แท็กเหล่านี้เป็นแท็กส่วนเพิ่มของ IE ไม่ใช่แท็กที่กำหนดในมาตรฐาน HTML 4.0 หรือ XHTML 1.0
2. การใส่เสียงควรใช้ด้วยความระมัดระวัง เพราะอาจรบกวนหรือสร้างความรำคาญให้ผู้ชม



# tag ใหม่ใน HTML5



# The **<audio>** tag

## *Function:*

Define sound, such as music or other audio streams.

## *Attributes:*

- src** url ของเสียงที่ต้องการเล่น
- type** ชนิดของไฟล์เสียง
- autoplay="autoplay"** ถ้าระบุจะเล่นทันทีถ้าพร้อม
- controls="controls"** ถ้าระบุจะมีปุ่มควบคุมขึ้นมาให้ผู้ใช้งานบังคับ
- loop="loop"** ถ้าระบุ จะวนซ้ำ
- preload= auto | metadata | none** ต้องการให้โหลดไฟล์เสียงทันทีที่ เปิดหน้าเว็บเพื่อเตรียมพร้อมสำหรับการเล่นหรือไม่ จะไม่มีผลถ้าระบุ autoplay

## *End tag:*

**</audio>**

## *Contains:*

*text* ควรเป็นข้อความที่บอกผู้ใช้ในกรณีที่เบราว์เซอร์ไม่รองรับแท็กนี้

## ตัวอย่างการใส่เสียง

```
<audio
  src="mysong.ogg" type="audio/ogg"
  src="mysong.mp3" type="audio/mp3"
  autoplay="autoplay"
  loop="loop"
>
```

Your browser does not support the audio tag.

```
</audio>
```

หมายเหตุ เบราว์เซอร์แต่ละตัวรองรับรูปแบบของไฟล์เสียงแตกต่างกัน ส่วนใหญ่รองรับ mp3 เพื่อแก้ปัญหารูปแบบไฟล์เสียงที่แต่ละเบราว์เซอร์รองรับไม่เท่ากัน เราอาจเตรียมเพิ่มเสียงมากกว่า 1 แบบและใส่ตัวบอกลักษณะประจำ (attribute) src หลายตัว

# The **<video>** tag

## *Function:*

Define video, such as a movie clip or other video streams.

## *Attributes:*

|                            |   |
|----------------------------|---|
| <b>src</b>                 | url ของวิดีโอที่ต้องการเล่น   |
| <b>type</b>                | ชนิดของไฟล์วิดีโอ   |
| <b>autoplay="autoplay"</b> | ถ้าระบุจะเล่นทันทีถ้าพร้อม (บางเบราว์เซอร์ไม่รองรับ)                  |
| <b>controls="controls"</b> | ถ้าระบุจะมีปุ่มควบคุมขึ้นมาให้ผู้ใช้งานบังคับ                         |
| <b>height</b>              | ความสูง player หน่วยเป็น pixel  |
| <b>width</b>               | ความกว้างของ player หน่วยเป็น pixel                                   |
| <b>preload="preload"</b>   | ถ้าระบุ จะโหลดวิดีโอทันทีที่เปิดหน้าเว็บเพื่อเตรียมพร้อมสำหรับการเล่น |

## *End tag:*

**</video>**

## *Contains:*

text ควรเป็นข้อความที่บอกผู้ใช้ในกรณีที่เบราว์เซอร์ไม่รองรับแท็กนี้

# ตัวอย่างการใช้วิดีโอ

<video

src="movie.ogg" type="video/ogg"

src="movie.mp4" type="video/mp4"

autoplay="autoplay"

controls="controls"

height="300" width="450"

>

Your browser does not support the video tag.

</video>

หมายเหตุ เบราว์เซอร์แต่ละตัวรองรับรูปแบบของไฟล์วิดีโอแตกต่างกัน ส่วนใหญ่รองรับไฟล์ตามมาตรฐาน h.264 เพื่อแก้ปัญหาที่เบราว์เซอร์ต่าง ๆ รองรับมาตรฐานข้อมูลวิดีโอที่แตกต่างกัน อาจต้องเตรียมไฟล์วิดีโอหลายรูปแบบ แล้วระบุ attribute src หลายตัว

# Scalable Vector Graphic (SVG)

- เป็นมาตรฐานภาพกราฟิกที่สามารถย่อขยายได้
- กำหนดภาพกราฟิก 2 มิติด้วยภาษา XML
- สามารถทำภาพเคลื่อนไหวได้
- เป็นมาตรฐานแนะนำของ W3C
  - SVG version 1.0 4 September 2001
  - SVG Version 1.1 ปรับปรุงใหม่ 16 August 2011

## การแสดงผลภาพกราฟิก SVG

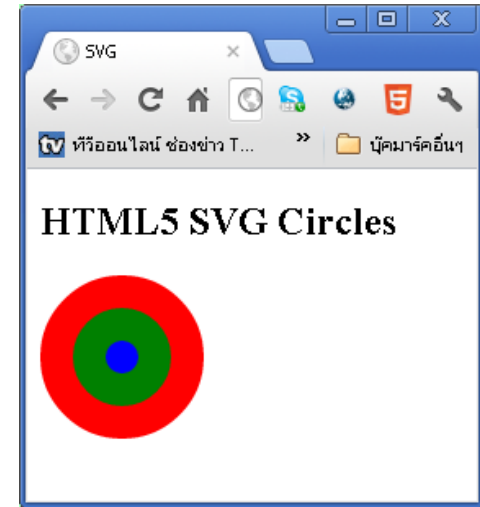
- เบราว์เซอร์รุ่นเก่าต้องติดตั้ง plugin เช่น Adobe SVG Viewer
- เบราว์เซอร์รุ่นใหม่ (ที่รองรับ HTML5) ส่วนใหญ่สามารถแสดงผลภาพกราฟิก SVG ได้โดยไม่ต้องติดตั้ง plugin
- ใช้ แท็ก **<svg>**
- รายละเอียดเรื่อง svg อยู่นอกขอบเขตวิชานี้ แต่ต้องการให้ภาพคร่าว ๆ ว่า ทำได้อย่างไร

# ตัวอย่างการใช้ SVG เขียนวงกลมซ้อนกัน

```

<!DOCTYPE html>
<head>
<title>SVG Demo</title>
</head>
<body>
<h2>HTML5 SVG Circles</h2>
<svg height="120" >
    <circle cx="50" cy="50" r="50" fill="red" />
    <circle cx="50" cy="50" r="30" fill="green" />
    <circle cx="50" cy="50" r="10" fill="blue" />
    Your browser does not support SVG.
</svg>
</body>
</html>

```





# The **<canvas>** tag

## *Function:*

ใช้สร้างพื้นที่สำหรับวาดภาพกราฟฟิก ต้องใช้ร่วมกับสคริปต์สำหรับเขียนโปรแกรมเช่น จาวาสคริปต์ (JavaScript)

## *Attributes:*

**height** ความสูงของพื้นที่สำหรับวาดภาพ หน่วยเป็น pixel

**width** ความกว้างของพื้นที่สำหรับวาดภาพ หน่วยเป็น pixel

## *End tag:*

**</canvas>**

## *Contains:*

*text* ควรเป็นข้อความที่บอกผู้ใช้ในกรณีที่เบราว์เซอร์ไม่รองรับแท็กนี้หรือบรรจุแท็กอื่นสำหรับทำงานทดแทน

การเขียนจาวาสคริปต์ อยู่นอกขอบเขตวิชานี้

## ตัวอย่างการใช้ **<canvas>**

```
<canvas id="animated_logo"  
    height="300" width="400"  
>  
  
</canvas>
```

**id** เป็น attribute ทั่วไป ใช้ตั้งชื่อสำหรับการอ้างอิงแท็กตัวใดตัวหนึ่ง  
เนื่องจากแท็ก **<canvas>** ต้องใช้ร่วมกับจาวาสคริปต์จึงจำเป็นต้องตั้งชื่อ  
เพื่อให้จาวาสคริปต์สามารถอ้างอิงได้

# การสร้างตาราง

## การสร้างตาราง

มีแท็กที่เกี่ยวข้องต่าง ๆ ดังนี้

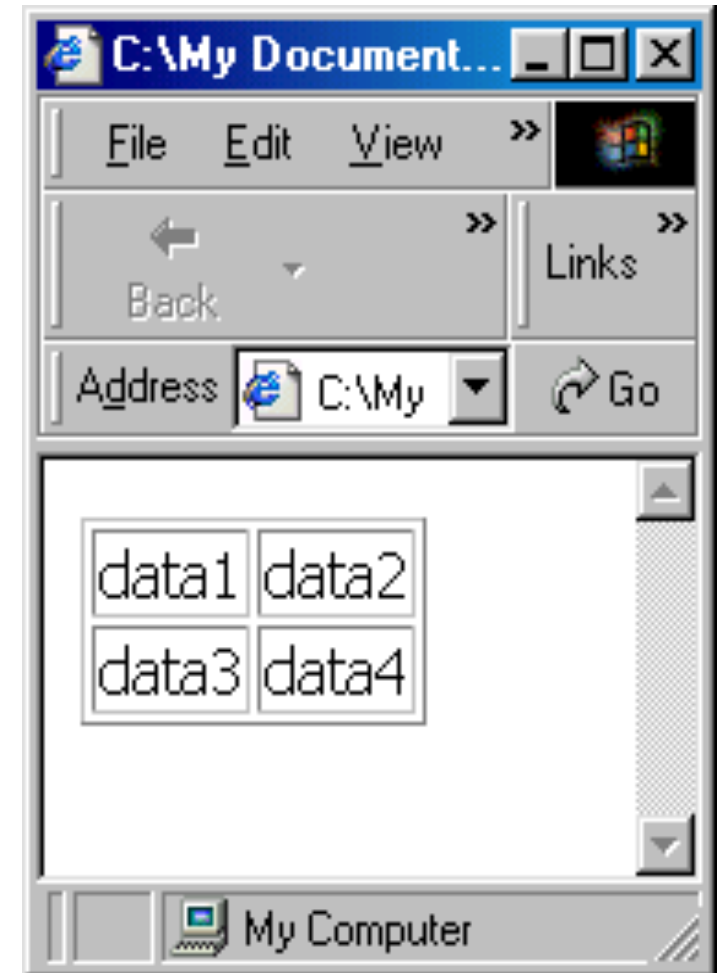
- **<table>** กำหนดตาราง
- **<th>** กำหนดหัวตาราง
- **<tr>** กำหนดแถวของตาราง
- **<td>** กำหนดข้อมูลของตาราง (เรียกว่า cell)
- **<caption>** ใช้ตั้งชื่อตาราง
- **<colgroup>** กำหนดกลุ่มของคอลัมน์เพื่อจัดรูปแบบ
- **<col />** กำหนดค่า attribute ให้กับคอลัมน์ของตาราง

ดูรายละเอียดเพิ่มเติมในเว็บไซต์

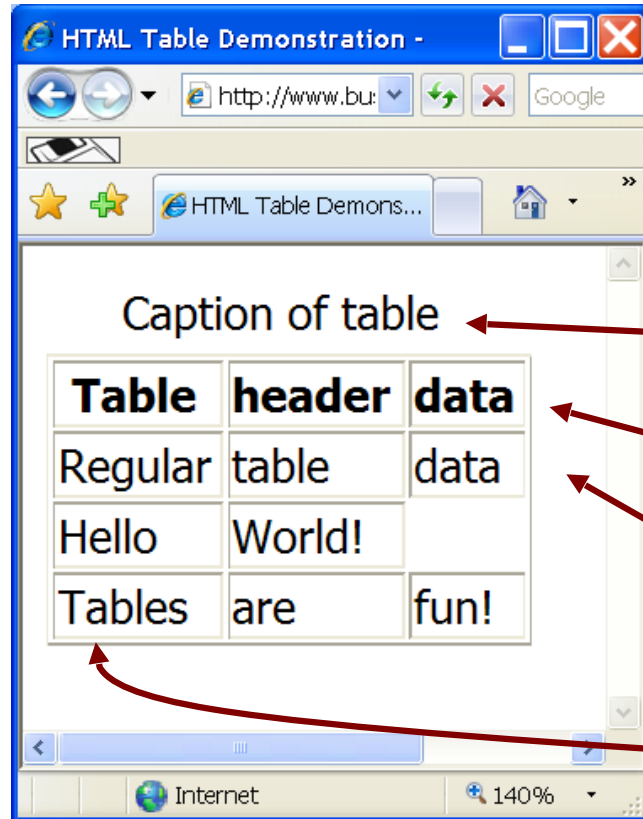
<http://www.bus.tu.ac.th/usr/wanchai/html/table.htm>

# A Simple Table

```
<table border="1">  
  <tr>  
    <td>data1</td>  
    <td>data2</td>  
  </tr>  
  <tr>  
    <td>data3</td>  
    <td>data4</td>  
  </tr>  
</table>
```



# ส่วนประกอบของ Table



แท็กที่เกี่ยวข้อง

`<table>`

`<caption>`

`<th>` table head

`<tr>` table row

`<td>` table data

# The **<table>** Tag

Function:

สร้างตารางสำหรับนำเสนอข้อมูลหรือจัด layout

Attributes: (ทุกตัวที่กล่าวถึงต่อไปนี้จะเลิกใช้ใน HTML5)

**align**=left | center | right

**width** ความกว้างของตาราง หน่วยเป็น **pixel**

**border** ความหนาของเส้นรอบตาราง หน่วยเป็น **pixel**

**bgcolor** สีพื้นของตาราง

End tag:

**</table>**

Contains:

**<caption>**, **<tr>**

ดูคำอธิบาย attribute เพิ่มเติมและ ตัวอย่างในเว็บ

<http://www.bus.tu.ac.th/usr/wanchai/html/>

สามารถเข้าด้วยชื่อย่อ **j.mp/html\_ex**

# The `<tr>` Tag

Function:

จัดข้อมูลในแต่ละแถวของตาราง

Attributes:

**align**=left | center | right

**bgcolor** สีพื้นของเซลล์ในบรรทัด

End tag:

`</tr>`

Contains:

`<th>`, `<td>`

*attribute ทุกตัว (มีมากกว่านี้) เลิกใช้ใน HTML5 ให้ใช้ CSS แทน*



# The `<td>` Tag

Function:

บอกขอบเขตของข้อมูลที่จะใส่ในเซลล์ของตาราง

Attributes:

**align**=left | center | right

**valign**=top | middle | bottom

**bgcolor** สีพื้นของเซลล์

**colspan** = จำนวนคอลัมน์ที่ต้องการให้เซลล์นี้ครอบคลุม

**rowspan** = จำนวนบรรทัดที่ต้องการให้เซลล์นี้ครอบคลุม

(deprecated)

(deprecated)

(deprecated)

End tag:

`</td>`

Contains:

ข้อความหรือแท็กอื่น เช่น `<a>` `<img>`

# The **<th>** Tag

Function:

ใช้กำหนดเซลล์หัวตาราง ใช้เช่นเดียวกับ <td> แต่แสดงข้อมูลในเซลล์ด้วยตัวหนา

Attributes:

**align**=left | center | right

(deprecated)

**valign**=top | middle | bottom

(deprecated)

**bgcolor** สีพื้นของเซลล์

(deprecated)

**colspan** = จำนวนคอลัมน์ที่ต้องการให้เซลล์นี้ครอบคลุม

**rowspan** = จำนวนบรรทัดที่ต้องการให้เซลล์นี้ครอบคลุม

End tag:

**</th>**

Contains:

ข้อความหรือแท็กอื่น เช่น **<a>** **<img>**

```
<table>  
<tr>  ROW1  </tr>  
<tr>  ROW2  </tr>  
</table>
```

ลำดับแถว

```
<table>  
<tr><th>Header1</th><th>Header2</th></tr>  
<tr><td>row2col1</td><td>row2col2</td></tr>  
<tr><td>row3col1</td><td>row3col2</td></tr>  
</table>
```

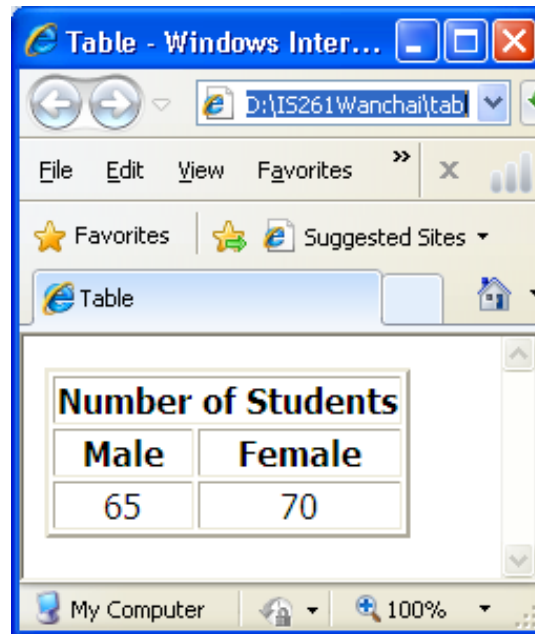
ลำดับแถว

ลำดับคอลัมน์

หัวคอลัมน์

# ฝึกสร้างตาราง

ให้นักศึกษาเขียนตารางให้มีลักษณะดังตัวอย่างต่อไปนี้



The screenshot shows a web browser window titled "Table - Windows Inter...". The address bar displays "D:\IS261\Wanchai\tab". The browser's menu bar includes "File", "Edit", "View", and "Favorites". Below the menu bar, there are sections for "Favorites" and "Suggested Sites". The main content area displays a table with the following structure:

| Number of Students |        |
|--------------------|--------|
| Male               | Female |
| 65                 | 70     |

The status bar at the bottom shows "My Computer", a lock icon, and a zoom level of "100%".

แถวแรกกินพื้นที่ 2 คอลัมน์  
ตัวเลขอยู่กลางเซลล์

การบ่งบอกลักษณะหรือสมบัติ  
หน้าเว็บโดยรวม

# The **<body>** tag

## *Function:*

define global characteristics of the page

## *Attributes:*

|                   |                                      |
|-------------------|--------------------------------------|
| <b>bgcolor</b>    | กำหนดสีพื้นของฉากหลัง                |
| <b>text</b>       | กำหนดสีของข้อความที่อยู่ใน body      |
| <b>link</b>       | กำหนดสีของ unvisited hypertext links |
| <b>alink</b>      | กำหนดสีของ active hypertext links    |
| <b>vlink</b>      | กำหนดสีของ visited hypertext links   |
| <b>background</b> | URL ของรูปภาพที่จะนำมาเป็นฉากหลัง    |

หมายเหตุ attribute เหล่านี้ถูกยกเลิก แนะนำให้ใช้ CSS แทน

ตัวอย่างการใช้แท็ก **<body>** ในการระบุสีข้อความและภาพฉากหลัง

```
<body bgcolor="blue">
```

```
<body bgcolor="black" text="white">
```

```
<body bgcolor="khaki" text="blue"  
background="woodtile.gif">
```

```
<body link="blue" alink="red"  
vlink="aqua">
```

หมายเหตุ ใช้ CSS ดีกว่า

# meta tag สำหรับระบุชุดอักษร

**meta** เป็นแท็กสำหรับให้รายละเอียดบางอย่างเพิ่มเติม จะใส่ไว้ในส่วนนำ (head) ของเอกสาร html

- ระบุชุดอักขระภาษาไทย

```
<meta http-equiv="Content-Type"
content="text/html; charset=tis-620" />
```

HTML5 ให้เขียนสั้น ๆ แบบนี้ได้

```
<meta charset="tis-620" />
```

charset สำหรับชุดอักขระไทยที่ใช้ได้มี

- tis-620
- windows-874

แนะนำให้ใช้ UTF-8 กรณีที่ใช้หลายภาษา และ Editor รองรับการเซฟแฟ้มที่มีการเข้ารหัสตัวอักษรแบบนี้ได้

อย่าลืมปิดเครื่องหมายคำพูด





## meta tag สำหรับระบุคำสำคัญ

การระบุคำสำคัญ (keyword) ในเอกสาร html เป็นการอำนวยความสะดวกให้ search engine ทราบว่าเราต้องการใช้คำใดเป็นคำสำคัญของเป้าหมายในการค้นบ้าง

```
<meta name="keywords" content="mango, pine apple, banana" />
```

# The Meta Description Tag

แท็ก meta สามารถนำมาใช้อธิบายว่าหน้าเว็บที่เขียนเป็นเรื่องเกี่ยวกับอะไร ซึ่ง search engine มักนำข้อความอธิบายนี้ไปโชว์ในหน้าเว็บของ search engine

```
<meta name= "description" content=
  "เราขายมังคุด กล้วย และสัปรด ผลไม้ชั้นยอด คุณภาพดี มีคุณค่าทาง
  โภชนาการสูง" />
```

## ตัวอย่างการใช้ meta tag ต่าง ๆ

```
<html>
<head>
<title>IT knowledge</title>
<meta charset="tis-620" />
<meta name="keywords" content="IT, ICT,
    Information Technology" />
<meta name="description" content="สาระน่ารู้เกี่ยวกับไอที" /
>
</head>
<body>
ถ้าท่านกำลังหาเว็บไซต์เกี่ยวกับไอที ท่านมาถูกที่แล้ว
</body>
</html>
```

# การบอกชนิดของเอกสาร (doctype)

- ควรบอกชนิดของเอกสารเพื่อให้เบราว์เซอร์รู้ว่า เป็นเอกสารชนิดใด รุ่นใด
- ก่อนหน้านี้การบอกชนิดเอกสารค่อนข้างจำยากเช่น XHTML ต้องบอกดังนี้

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
```

- พอเป็น HTML5 สามารถบอกสั้น ๆ ได้ดังนี้

```
<!DOCTYPE HTML>
```



ถ้าจะใช้ HTML5 ควรบอก doctype เสมอเพื่อหลีกเลี่ยงปัญหาบางเบราว์เซอร์ไม่ทำงานอย่างที่คิด เพราะ HTML มีหลาย version ซึ่งอาจทำงานต่างกันในช่วง version ถ้าไม่บอก doctype เบราว์เซอร์ อาจเดาว่าเป็น version ที่แตกต่างจากที่คนเขียนตั้งใจก็เป็นได้

# MOBILE FRIENDLY WEBSITES



# Mobile Friendly Website

- เป็นการออกแบบเว็บไซต์ให้เป็นมิตรกับผู้ใช้อุปกรณ์พกพา เช่น ใช้ขนาดฟอนต์ให้เหมาะกับหน้าจอขนาดเล็ก จัด layout ไม่แน่นจนเกินไป
- แนวทางการออกแบบ
  - ออกแบบแยกกันไปเลยมักเป็นคนละไซต์ (Adaptive Web Design)
  - ใช้ไซต์เดียวแต่เขียนเว็บให้ปรับตัวให้เหมาะสมกับอุปกรณ์แต่ละชนิด (Responsive Web Design)

กูเกิ้ลให้ความสำคัญกับเว็บไซต์ที่เป็นมิตรกับอุปกรณ์พกพา ถ้าค้นด้วยกูเกิ้ลจะลิสต์เว็บไซต์ที่เป็นมิตรกับอุปกรณ์พกพาขึ้นก่อน



# Responsive Web Design (RWD)

- เป็นการออกแบบเว็บให้เหมาะกับอุปกรณ์หลากหลายชนิด นิยมปรับเปลี่ยน layout ให้เหมาะกับขนาดหน้าจอโดยอัตโนมัติ
- ใช้ ความสามารถของ CSS ในการตรวจสอบขนาดหน้าจอแล้วเลือกใช้ CSS style ที่เตรียมไว้แล้วให้เหมาะสมกับขนาดหน้าจอ
- อาจเตรียมรูปไว้หลายขนาดเพื่อให้เหมาะกับจอขนาดที่ต่างกัน



# Mobile Friendly ด้วย viewport

ด้วยการใส่แท็ก meta ไว้ใน head ดังนี้

```
<meta name="viewport" content=
"width=device-width, initial-scale=1.0, maximum-scale=1.0"
```

| Attribute     | Possible values                                     | Description   |
|---------------|---|---|
| width         | Integer value (in pixels) or constant device-width  | Defines the viewport width  |
| height        | Integer value (in pixels) or constant device-height | Defines the viewport height   |
| initial-scale | Floating value (0.1 to $n$ ); 1.0 is no scale       | Defines the initial zoom scale of the viewport                            |
| user-scalable | no or yes   | Defines whether we will allow the user to zoom in and out in the viewport |
| minimum-scale | Floating value (0.1 to $n$ ). 1.0 is no scale       | Defines the minimum zoom scale of the viewport                            |
| maximum-scale | Floating value (0.1 to $n$ ). 1.0 is no scale       | Defines the maximum zoom scale of the viewport                            |



# การ upload เว็บเพจของนักศึกษา

- FTP ไฟล์ที่ต้องการ upload ไปที่ server ชื่อ std.bus.tu.ac.th โดยใช้ รหัสนักศึกษาเป็น UserId และใช้ password ที่แจกให้ (คนละตัวกับการใช้บริการอินเทอร์เน็ต) โดยใช้โปรแกรมสำหรับ FTP หรือใช้ความสามารถของ วินโดวส์ในการ upload อ่านได้ในเว็บวิชานี้

<http://www.bus.tu.ac.th/usr/wanchai/is216>

- เข้าเว็บเพื่อดูเอกสารที่ upload ไปแล้วได้ที่

<http://std.bus.tu.ac.th/userid/filename>

อ่านโจทย์ assignment HTML และ CSS ได้ที่

<http://www.bus.tu.ac.th/usr/wanchai/is216/>

หรือเข้าทางชื่อย่อ <http://bit.ly/tbsis216>

# การสอบทานเว็บไซต์

- ตรวจสอบว่าเราเขียนถูกต้องตามหลักที่ดีในการใช้ HTML หรือไม่

<https://validator.w3.org/>

- ตรวจสอบว่าเขียนถูกต้องตามหลัก CSS หรือไม่

<https://jigsaw.w3.org/css-validator/>

- ตรวจสอบว่าเขียนลิงค์ไม่เหมาะสมหรือไม่

<https://validator.w3.org/checklink>

- ตรวจสอบว่าเป็นมิตรกับอุปกรณ์พกพามากน้อยเพียงใด

<https://search.google.com/test/mobile-friendly>

# เครื่องมือในการเขียนเว็บ

## (Web Authoring Tools)

- Pure code-based editor
- Pure WYSIWYG (What You See Is What You Get) editor
- Compound editor (ผสมทั้ง 2 แบบ เข้าด้วยกัน)
- web generator (ใช้ซอฟต์แวร์ช่วยสร้างเว็บ ไม่ต้องเขียนเอง เช่น Microsoft Word สามารถแปลงไฟล์ word ให้เป็น html ได้)
- Web Content Management System

# Web Authoring Tools

*โปรแกรมสำหรับช่วยเขียนเว็บ* เพิ่มประสิทธิภาพในการเขียนเว็บ  
ได้มาก แต่มักต้องซื้อซอฟต์แวร์ราคาแพง บางโปรแกรมซับซ้อน  
มาก ต้องใช้เวลาในการเรียนรู้นาน เหมาะสำหรับผู้  
เขียนเว็บเป็นอาชีพ

*ตัวอย่างโปรแกรมช่วยเขียนเว็บเช่น*

- Adobe Dreamweaver      \$399
- Microsoft Expression      \$299
- Brackets      Free ดาวน์โหลดได้ที่ <http://brackets.io/>

## โปรแกรมสำหรับผลิตเว็บเฉพาะด้าน

มีโปรแกรมในท้องตลาดที่ช่วยผลิต(generate หรือ build)เว็บโดยไม่ต้องเขียนเอง ผู้ใช้เพียงแค่ให้ข้อมูลแล้วโปรแกรมจะผลิตหน้าเว็บให้ โดยที่ผู้ใช้โปรแกรมไม่จำเป็นต้องรู้ภาษา HTML ก็ได้ แต่ถ้าผู้ใช้ต้องการดัดแปลงแก้ไขหน้าเว็บที่โปรแกรมผลิตให้มักจำเป็นต้องมีความรู้ภาษา HTML จึงจะแก้ไขหน้าเว็บที่โปรแกรมผลิตให้ได้ ผู้ใช้โปรแกรมมักสามารถเลือกแม่แบบ (template) แบบ (style) ต่าง ๆ ที่มีเป็นตัวอย่างให้เลือกได้ ตัวอย่างโปรแกรมประเภทนี้ เช่น

- โปรแกรมสำหรับสร้างเว็บไซต์ขายของ (E-commerce)
- โปรแกรมสำหรับอัลบั้มรูปภาพ
- โปรแกรมสำหรับสร้างบทเรียนอิเล็กทรอนิกส์

## เว็บสำหรับสร้างหน้าเว็บ

- เว็บยุคแรกเป็นเว็บแบบนิ่ง ๆ (static web) ผู้ชมเว็บอ่านอย่างเดียวไม่สามารถโต้ตอบกับเว็บได้ ผู้เขียนเว็บต้องมีความรู้ภาษา HTML
- ต่อมามีการพัฒนาเว็บให้สามารถโต้ตอบกับผู้ชมได้ ผู้ชมสามารถใส่ข้อมูลของตนเองเข้าไปได้ทางเบราว์เซอร์โดยไม่จำเป็นต้องรู้ภาษา HTML โปรแกรมที่ทำงานอยู่ทางฝั่งเครื่องแม่ข่ายจะรับข้อมูลที่เรานำไปเก็บไว้ในเครื่องแม่ข่ายและนำข้อมูลนั้นมาแปลงให้เป็น HTML แล้วส่งไปให้เบราว์เซอร์แสดงผลเป็นหน้าเว็บ ความสามารถของเว็บที่ทำแบบนี้ได้นิยามเรียกว่า Web 2.0 ตัวอย่างของเว็บแบบนี้เช่น Facebook และ Wikipedia

# Web Application ใช้อะไรบ้าง

- ✓ • **HTML** (HyperText Markup Language)
- ✓ • style sheet
  - JavaScript
  - DHTML (Dynamic HTML)
  - CGI (Common Gateway Interface)
  - XML (Extensible Markup Language)
  - AJAX (Asynchronous JavaScript and XML)

# Web Fonts

- Font ฟรีสำหรับใช้กับเว็บที่ Google <https://fonts.google.com/>  
วิธีใช้ <https://developers.google.com/fonts/>