DEPARTMENT OF
COMPUTER SCIENCE
計算機科學系

INTRODUCTION TO WEB DEVELOPMENT

ITEC2016 - DATA-DRIVEN VISUALIZATION FOR THE WEB

CHAPTER 1 - 2018 - HKBU

DR. MARTIN CHOY

Learning Objectives

- Able to create **HTML** pages using mark-up
- Able to style HTML pages using **CSS**.
- Able to use CSS to control the layout of a web page.

What is HTML?

- HTML stands for **Hyper Text Markup Language**.
- It is a markup language, but **not a programming language**.
- Note that a **markup language** is a system for annotating a text in a way that is distinguishable from that text.

Tags, Elements and Attributes

- The name of the **tag** appears between the **angle brackets**, like **<tag_name>**.
- This is a **start tag** and the name of an **end tag** is preceded by a **forward slash**, i.e., **</tag_name>**
- The marked-up content between a pair of start and end tags is known as an **element**.
  - Example: `<tag_name>Marked up content</tag_name>`

Tags, Elements and Attributes

- **Nested element** is allowed
  - Example:

```
<parent_tag>
    <child_tag>Marked up content</child_tag>
</parent_tag>
```

- Elements can have **attributes** that appear inside the start tag and consist of one or more **name-value pairs** with format

`attribute_name="attribute_value"` or `attribute_name='attribute_value'`

Example

```
<html>
<head>
    <title> This is a starting page </title>
</head>
<body>
    <h1 style="text-align:center"> This is a starting page </h1>
    Click <a href="http://www.comp.hkbu.edu.hk"> Here </a>
    to go to Computer Department of HKBU <br>
</body>
</html>
```

Standard Attributes

- Some **standard attributes** that are supported all HTML tags.
  - **class** – specifies a **class name** for an element
  - **id** – specifies a **unique id** for an element.
  - **style** – specifies **an inline style** for an element.
- Reference
  http://www.w3schools.com/tags/ref_standardattributes.asp

HTML Links

- Links are found in nearly all webpages.
- Links allow users to click their way from page to page.
- Links are specified in HTML using the **<a>** tag.
- The **href** attribute specifies the **destination**, which could be
  - another document, and/or
  - another element (specified by id)

HTML Links

- Examples
  - Create a named div inside an HTML document:
    <div **id="cp3"**> Chapter 3
  - Create a link to the “Chapter 3” inside the same document:
    <a **href="#cp3"**> Go to Chapter 3
  - Or, create a link to the “Chapter 3” from another page:
    <a **href="anchor1.html#cp3"**> Go to Chapter 3

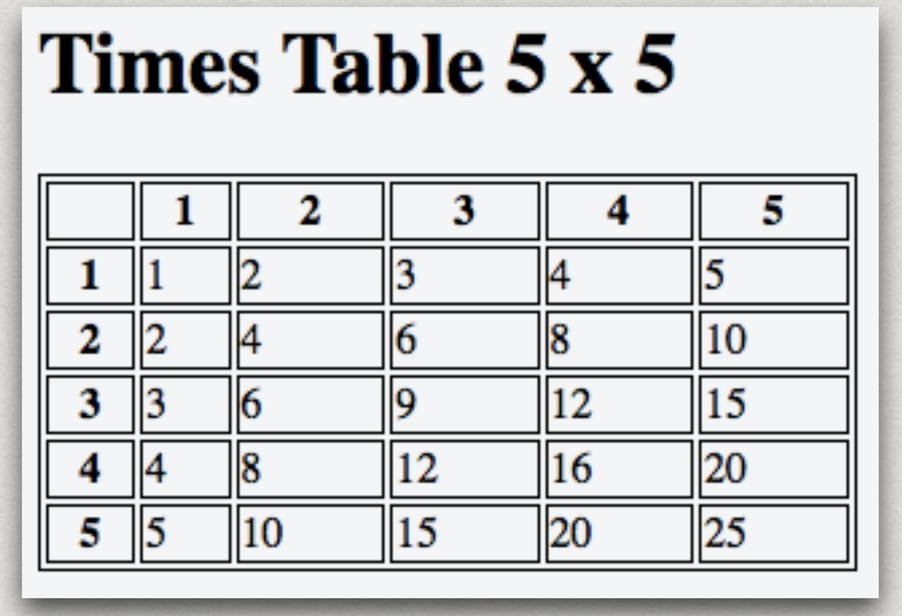
```html
<!DOCTYPE html>
<!-- anchor2.html -->
<html>
<head>
  <title>Anchor example 2</title>
</head>
<body>
  <h1>Anchor example 2: Link to another document</h1>
  <a href="anchor1.html#cp3">Go to Chapter 3 of Anchor example 1</a>
</body>
</html>
```

```html
<!DOCTYPE html>
<!-- anchor1.html -->
<html>
<head>
    <title>Anchor example 1</title>
</head>
<body>
    <h1>Anchor example 1: Link to the same document</h1>
    <p><a href="#cp3">Go to Chapter 3</a></p>
    <p><a href="anchor2.html">Go to Anchor example 2</a></p>
    <h2>Chapter 1</h2>
    <h2>Chapter 2</h2>
    <h2><a id="cp3">Chapter 3</a></h2>
</body>
</html>
```

HTML Tables

- Tables are defined with **<table>** tag.
- A table is divided into **rows** with **<tr>** tag.
- Each row is divided **data cells** with **<td>** tag.
- **<td>** tag can contain text, links, images, lists, forms, other tables, etc.
- **<th>** tag stands for **table header** in which text element is displayed as **bold and centered**.

```
<!DOCTYPE html>
<html>
<head>
    <title>Times Table</title>
    <style>
        table, th, td {
            border: 1px solid black;
        }
    </style>
</head>
<body>
    <h1>Times Table 5 x 5</h1>
    <table style="width:50%">
        <tr><td></td><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th></tr>
        <tr><th>1</th><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td></tr>
        <tr><th>2</th><td>2</td><td>4</td><td>6</td><td>8</td><td>10</td></tr>
        <tr><th>3</th><td>3</td><td>6</td><td>9</td><td>12</td><td>15</td></tr>
        <tr><th>4</th><td>4</td><td>8</td><td>12</td><td>16</td><td>20</td></tr>
        <tr><th>5</th><td>5</td><td>10</td><td>15</td><td>20</td><td>25</td></tr>
    </table>
</body>
</html>
```



Times Table 5 x 5

	1	2	3	4	5
1	1	2	3	4	5
2	2	4	6	8	10
3	3	6	9	12	15
4	4	8	12	16	20
5	5	10	15	20	25

HTML Lists

- Unordered list
  - Defined with **<ul>** tag
  - Each item starts with **<li>** tag
- Ordered list
  - Defined with **<ol>** tag
  - Each item starts with **<li>** tag
- List can be nested

```
<!DOCTYPE html>
<html>
<head>
    <title>HTML lists</title>
</head>
<body>
    <ul>
        <li>Item 1</li>
        <li>Item 2: Nested order list
            <ol>
                <li>Nested item 1</li>
                <li>Nested item 2</li>
            </ol>
        </li>
        <li>Item 3</li>
    </ul>
</body>
</html>
```

- Item 1
- Item 2: Nested order list
  1. Nested item 1
  2. Nested item 2
- Item 3

Special Characters

- HTML character references are **numeric or symbolic names** that can be used instead of literal characters in an HTML document.

HTML character reference	Equivalent character	Meaning
&lt;	<	Less than
&gt;	>	Greater than
&quot;	"	Quotation mark
&amp;	&	Ampersand
	(a space)	Non-breaking

- Reference
  http://www.w3schools.com/tags/ref_charactersets.asp

What is CSS?

- CSS stands for **Cascading Style Sheets**
- Styles defined how to display HTML elements

CSS and its Benefits

- **HTML was intended to define the content of a document.**
- **CSS defines the style and formatting**:
  - Specify display details once for any element.
  - Styles can be saved in external .css files.
  - Change presentation of all pages in **one single file**.

Where to put CSS?

- **External** style sheet
  - Style applies to many pages, each page must link with **<link>** tag inside the head section
- **Internal** style sheet
  - For a single document has a unique style, specified using **<style>** tag
- **Inline** style
  - Style tag using **style attribute**

CSS Linkage

- How CSS is inserted:
  - External
  - Internal
  - Inline

```
<html>
<head>
    <link rel="stylesheet" href="external.css">
    <style>
        p { color:#ff0033; }
    </style>
</head>
<body>
    <p style="color:#ff0033;"> Some text. </p>
</body>
</html>
```

CSS Syntax

- Two main parts: `Selectors { declarations }`
- **Selectors**
- Specify the HTML elements to be styled.
  - Multiple selectors are separated with a comma.
- **Declarations**
  - Each declaration consists of **a property and a value**.
  - **Multiple declarations** are separated with a **semi-colon**.
  - Comment enclosed between /* and */

Matching of Selectors

- Selects all elements by **element name** `p {...}`
- Selects all elements by **class name** `.marked {...}`
- Selects element by **id** `#color {...}`
- Specify all elements. `* {...}`

Matching of Selectors

- Some CSS properties
  - **background-color:** specifies background color to be used.
  - **color:** specifies color of text.
  - **text-align:** specifies the horizontal alignment of text in an element
  - **text-transform:** controls the capitalization of text
  - **text-decoration:** specifies the decoration added

CSS Properties

- Some CSS properties, cont'
  - **font-family:** specifies the font for an element.
  - **font-weight:** sets how thick or thin characters in text should be displayed.
  - **font-style:** specifies the font style for a text.
  - **font-size:** sets the size of a font.

```
body {
    background-color: black;
    color: white;
    font-family: times, arial, serif;
}
h1 {
    text-align: center;
    text-transform: uppercase;
    text-decoration: underline;
}
h2 {
    font-weight: bold;
    font-style: oblique;
}
```

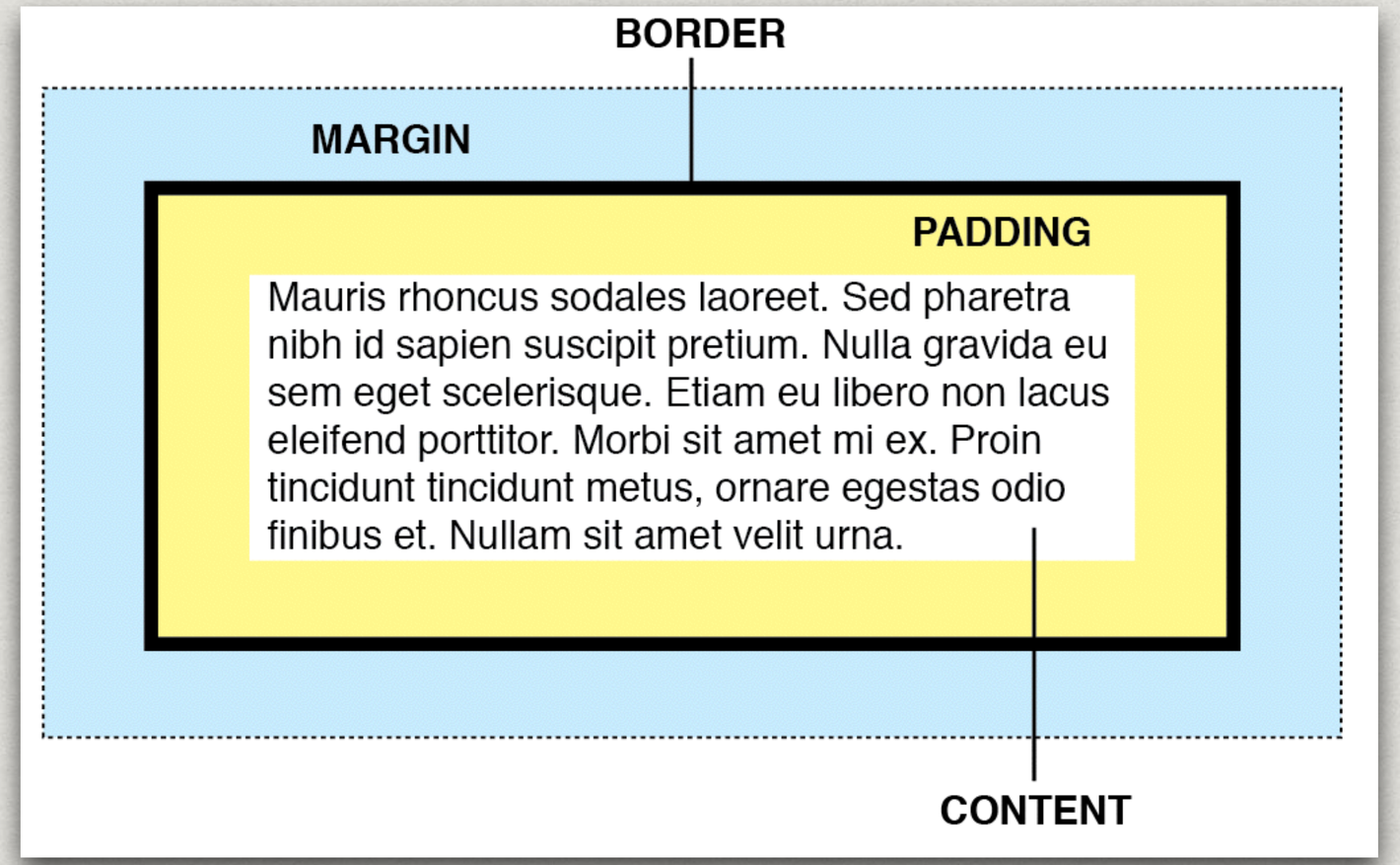
Using CSS for Page Layout

- What you should know first?
  - CSS Box Model
  - Block and inline elements of HTML
  - Float and Clear

- CSS reference: http://www.w3schools.com/cssref/default.asp

CSS Box Model

- All HTML elements can be considered as **boxes**.



CSS Box Model

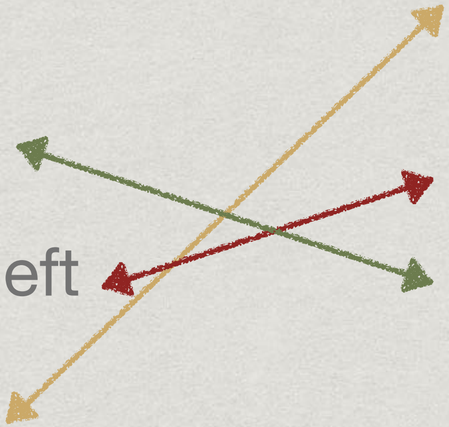
- **Margin** - Clears an area around the border. The margin does not have a background color, it is completely transparent
- **Border** - A border that goes around the padding and content. The border is affected by the background color of the box
- **Padding** - Clears an area around the content. The padding is affected by the background color of the box
- **Content** - The content of the box, where text and images appear

CSS Box Model

- When you set the **width** and **height** properties of an element with CSS, you **just set the width and height of the content area**.
- CSS units
  - % - relative to container width
  - em & rem - relative to current font size

CSS Box Model

- margin, padding
  - top right bottom left
  - top right_and_left bottom
  - top_and_bottom right_and_left
  - all_four

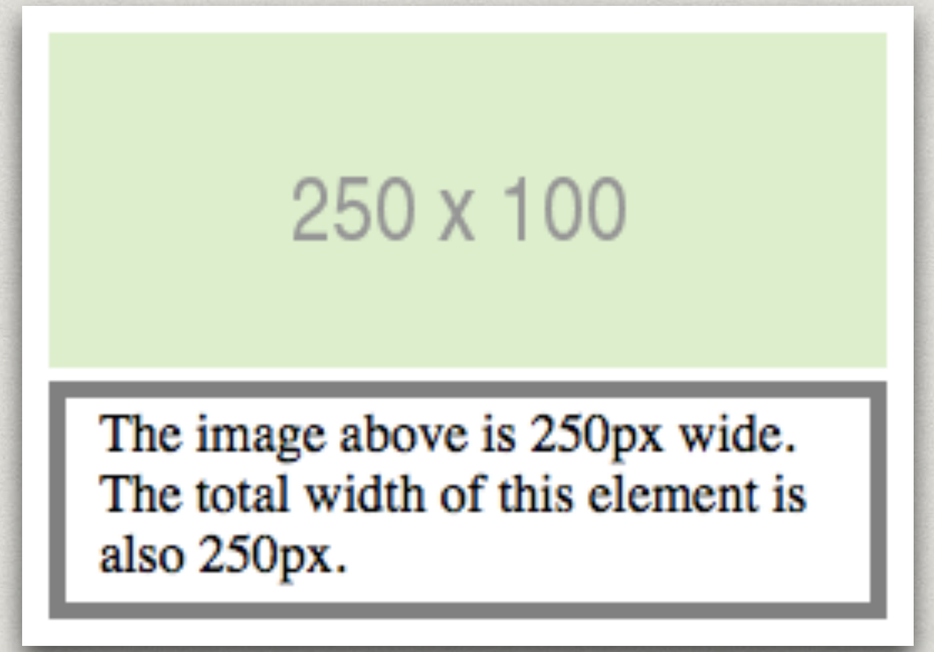


- Example
  - margin:0px;
  - padding:2px 10px;
  - padding:2px 10px 5px;

CSS Box Model

- border
  - border-width, border-style, border-color
- Some values of **border-style**
  - none, dotted, dashed, solid, double
- Example
  - **border: 5px solid gray;**

```html
<!DOCTYPE html>
<html>
<head>
<title>CSS box model</title>
    <style>
        .ex {
            width: 220px;
            padding: 2px 10px 5px;
            border: 5px solid gray;
            margin: 0px;
        }
    </style>
</head>
<body>
    <img src="http://via.placeholder.com/250x100/dec.png" width="250" height="100" />
    <br>
    <div class="ex">
        The image above is 250px wide.<br>
        The total width of this element is also 250px.
    </div>
</body>
</html>
```



Block and Inline elements of HTML

- HTML elements can be either **block** level or **inline**.
  - A **block** element is an element that takes up the **full width** available, and has a **line break** before and after it.
  - Example: <h1>, <p>,
- An **inline** element only **takes up as much width as necessary**, and **does not force line breaks**.
  - Example: , <a>

Display Property

- Changing an **inline element to a block element**, or vice versa, can be achieved using **display** property of CSS.
  - Example:

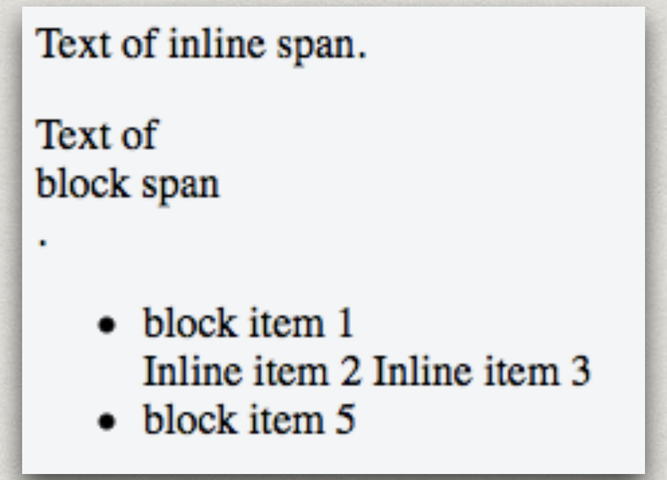
```
li {display:inline;} /* display li as inline element */

span {display:block;} /* display span as block element */
```

- To **hide** an element, we can set its CSS **display property** to **none**

```
span {display:none;} /* this element will not be displayed */
```

```
<!DOCTYPE html>
<html>
<head>
    <title>CSS example of display</title>
</head>
<body>
    <p>Text of <span>inline span</span>.</p>
    <p>Text of <span style="display:block">block span</span>.</p>
    <ul>
        <li>block item 1</li>
        <li style="display:inline">Inline item 2</li>
        <li style="display:inline">Inline item 3</li>
        <li style="display:none">hidden item 4</li>
        <li>block item 5</li>
    </ul>
</body>
</html>
```



CSS Float & Clear

- An element can be pushed to the **left or right**, allowing other elements to wrap around it
- How elements float
  - Elements are **floated horizontally**.
  - A floated element will move as far to the left or right as it can.
  - The elements after the floating element will **flow around it**. **To avoid this**, use the **clear** property.
  - The elements before the floating element will not be affected.

CSS Float & Clear

- The **clear property** specifies which sides of an element other floating elements are not allowed.
  - **clear:left** – No floating elements allowed on the left side.
  - **clear:right** – No floating elements allowed on the right side.
  - **clear:both** – No floating elements allowed on either the left or the right side.
  - **clear:none** – Default. Allows floating elements on both sides.

```
<!DOCTYPE html>
<html>
<head>
    <title>CSS example of float</title>
    <style>
        #logo {float: left;}
        #p4 {clear: both;}
    </style>
</head>
<body>
    <div id="logo">
        <img src="http://via.placeholder.com/150x150/dec.png" alt="Logo"/>
    </div>

    <p id="p1">Text area 1; Text area 1; Text area 1; Text area 1;</p>
    <p id="p2">Text area 2; Text area 2; Text area 2; Text area 2;</p>
    <p id="p3">Text area 3; Text area 3; Text area 3; Text area 3;</p>
    <p id="p4">Text area 4; Text area 4; Text area 4; Text area 4;</p>
    <p id="p5">Text area 5; Text area 5; Text area 5; Text area 5;</p>
</body>
</html>
```