Lecture02: Cascading Style Sheet and HTML Validation

EGCI427

Topics

- Recap HTML
- CSS

CSS

▶ **CSS** stands for **C**ascading **S**tyle **S**heets

▶ Styles define **how to display** HTML elements

▶ Styles were added to HTML 4.0 **to solve a problem**

▶ **External Style Sheets** can save a lot of work

▶ External Style Sheets are stored in **CSS files**

Styles Solved a Big Problem

▸ HTML was never intended to contain tags for formatting a document

▸ HTML was intended to define the content of a document, like:

```
<h1>This is a heading</h1>
<p>This is a paragraph.</p>
```

▸ When tags like <font>, and color attributes were added to the HTML 3.2 specification, it started a nightmare for web developers
  ▸ Development of large web sites, where fonts and color information were added to every single page, became a long and expensive process
  ▸ To solve this problem, the World Wide Web Consortium (W3C) created CSS

▸ In HTML 4.0, all formatting could be removed from the HTML document, and stored in a separate CSS file.

Cascading Style Sheet – CSS

▶ CSSs provide the means to control and change presentation of HTML documents

▶ CSS is not technically HTML, but can be embedded in HTML documents

▶ Style sheets allow you to impose a standard style on a whole document, or even a whole collection of documents

▶ Style is specified for a tag by the values of its properties

Levels of Style Sheets

▸ There are three levels of style sheets

- ▸ Inline - specified for a specific occurrence of a tag and apply only to that tag
  – This is fine-grain style, which defeats the purpose of style sheets - uniform style
- ▸ Document-level style sheets - apply to the whole document in which they appear
- ▸ External style sheets - can be applied to any number of documents

▸ When more than one style sheet applies to a specific tag in a document, the lowest level style sheet has precedence

- ▸ In a sense, the browser searches for a style property spec, starting with inline, until it finds one (or there isn't one)

Levels of Style Sheets (cont.)

▸ Inline style sheets appear in the tag itself

▸ Document-level style sheets appear in the head of the document

▸ External style sheets are in separate files, potentially on any server on the Internet

    ▸ Written as text files with the MIME type text/css

Linking an External Stylesheet

▸ A `<link>` tag is used to specify that the browser is to fetch and use an external style sheet file

```
<link rel = "stylesheet"  type = "text/css"
 href = "http://www.wherever.org/termpaper.css">
</link>
```

▸ External style sheets can be validated

```
http://jigsaw.w3.org/css-validator/validator-
   upload.html
```

Style Specification Formats

▸ Format depends on the level of the style sheet

▸ Inline:

  ▸ Style sheet appears as the value of the style attribute
  ▸ General form:

```
style = "property_1: value_1;
         property_2: value_2;
         …
         property_n: value_n"
```

  ▸ Example:

```
<body style="background-color:#E9E9D1">
```

Format for Document-level

▶ Style sheet appears as a list of rules that are the content of a `<style>` tag

▶ The `<style>` tag must include the type attribute, set to `"text/css"`

▶ The list of rules must be placed in an HTML comment, because it is not HTML

▶ Comments in the rule list must have a different form - use C comments (`/*…*/`)

General Form, Document Level

▸ General form:

```
<style type = "text/css">
 <!--
   rule list
 -->
</style>
```

▸ Form of the rules:
selector {list of property/values}

　　▸ Each property/value pair has the form:
　　`property: value`

　　▸ Pairs are separated by semicolons, just as in the value of a `<style>` tag

General Form, External style sheets

▶ Form is a list of style rules, as in the content of a `<style>` tag for document-level style sheets

This is inside your html file

This is inside ex1.css

```
<html>
<head>
<link rel="stylesheet" type="text/css"
href="ex1.css" />
</head>
<body>
<h1>This header is 36 pt</h1>
<h2>This header is blue</h2>
<p>This paragraph has a left margin of
50 pixels</p>
</body>
</html>
```

```
body {background-color: yellow}
h1 {font-size: 36pt}
h2 {color: blue}
p {margin-left: 50px}
```

Selector Forms: Simple

▶ The selector is a tag name or a list of tag names, separated by commas

- ▶ `h1 {font-size:24pt}`
- ▶ `h2,h3 {font-size:20pt}`
- ▶ The property values in the rule apply to all occurrences of the named element.

▶ Contextual selectors

- ▶ Style only applies to elements in certain positions
- ▶ `body b em {font-size:14pt}`
  - ▶ *Applies to the content of emphasis elements that are descendants of bold element in the body of the document*

Class Selectors

▸ Used to allow different occurrences of the same tag to use different style specifications

▸ A style class has a name, which is attached to a tag name
  ▸ `p.narrow {property/value list}`
  ▸ `p.wide {property/value list}`

▸ The class you want on a particular occurrence of a tag is specified with the `class` attribute of the tag
▸ For example,
```
<p class = "narrow">
...
</p>
...
<p class = "wide">
...
</p>
```

Generic Selectors

▸ A generic class can be defined if you want a style to apply to more than one kind of tag

▸ A generic class must be named, and the name must begin with a period

▸ Example,
  ▸ `.really-big { … }`
▸ Use it as if it were a normal style class
```
<h1 class = "really-big"> … </h1>
...
<p class = "really-big"> … </p>
```

`id` Selectors

▸ An id selector allow the application of a style to one specific element

▸ General form:

```
#specific-id {property-value list}
```

▸ Example:

```
#section14 {font-size: 20}
```

In your html:

```
<h2 id="section14"> 1.4 Calico Cats </h2>
```

Pseudo Classes

▸ Pseudo classes are styles that apply when something happens, rather than because the target element simply exists

▸ Names begin with colons

▸ `hover` classes apply when the mouse cursor is over the element

▸ `focus` classes apply when an element has focus

Pseudo Class Example

```
<!-- pseudo.html -->
<html xmlns = "http://www.w3.org/1999/xhtml">
  <head> <title> Checkboxes </title>
    <style type = "text/css">
      input:hover {color: red;}
      input:focus {color: green;}
    </style>
  </head>
  <body>
    <form action = "">
      <p>
        Your name:
        <input type = "text" />
      </p>
    </form>
  </body>
</html>
```

Properties

▶ There are 60 different properties in 7 categories:

- ▶ Fonts
- ▶ Lists
- ▶ Alignment of text
- ▶ Margins
- ▶ Colors
- ▶ Backgrounds
- ▶ Borders

Property Values

▸ Keywords - left, small, …
  ▸ Not case sensitive

▸ Length - numbers, maybe with decimal points

▸ Units:
  ▸ px - pixels
  ▸ in - inches
  ▸ cm - centimeters
  ▸ mm - millimeters
  ▸ pt - points
  ▸ pc - picas (12 points)
  ▸ em - height of the letter 'm'
  ▸ ex-height - height of the letter 'x'
  ▸ No space is allowed between the number and the unit specification e.g., 1.5 in is illegal!

Property Value Forms (continued)

▸ Percentage - just a number followed immediately by a percent sign

▸ URL values
  ▸ url(protocol://server/pathname)

▸ Colors
  ▸ Color name
  ▸ rgb(n1, n2, n3)
    ▸ Numbers can be decimal or percentages
  ▸ Hex form: #XXXXXX

▸ Property values are inherited by all nested tags, unless overridden

Font Properties

▸ `font-family`

  ▸ Value is a list of font names - browser uses the first in the list it has

  ▸ `font-family: Arial, Helvetica, Courier`

  ▸ Generic fonts: serif, sans-serif, cursive, fantasy, and monospace (defined in CSS)

    ▸ Browser has a specific font for each

  ▸ If a font name has more than one word, it should be single-quoted

    ▸ `Font-family: 'Time New Roman'`

Font Properties (continued)

- `font-size`
  - Possible values: a length number or a name, such as smaller, xx-large, etc.
  - `font-size:10pt`

- `font-style`
  - italic, oblique (useless), normal

- `font-weight` - degrees of boldness
  - bolder, lighter, bold, normal
    - Could specify as a multiple of 100 (100 – 900)

- `font` (shorthand)
  - For specifying a list of font properties
    `font: bolder 14pt Arial Helvetica`
  - Order must be: style, weight, size, name(s)

Font Properties (continued)

▶ ✍ **try:** fonts.html and display

▶ ✍ **try:** fonts2.html and display

▶ The text-decoration property
  ▶ line-through, overline, underline, none
  ▶ `letter-spacing` – value is any length property value

List properties

▸ list-style-type

▸ Unordered lists
  ▸ Bullet can be a disc (default), a square, or a circle
  ▸ Set it on either the <ul> or <li> tag
    ▸ On <ul>, it applies to list items

```
<h3> Some Common Single-Engine Aircraft </h3>
  <ul style = "list-style-type: square">
      <li> Cessna Skyhawk </li>
      <li> Beechcraft Bonanza </li>
      <li> Piper Cherokee </li>
  </ul>
```

List properties (cont.)

▸ On <li>, list-style-type applies to just that item

```
<h3> Some Common Single-Engine Aircraft </h3>
 <ul>
      <li style = "list-style-type: disc">
            Cessna Skyhawk </li>
      <li style = "list-style-type: square">
            Beechcraft Bonanza </li>
      <li style = "list-style-type: circle">
            Piper Cherokee </li>
 </ul>
```

Some Common Single-Engine Aircraft

- Cessna Skyhawk
- Beechcraft Bonanza
- Piper Cherokee

Some Common Single-Engine Aircraft

- Cessna Skyhawk
- Beechcraft Bonanza
- Piper Cherokee

List properties (cont.)

▸ Could use an image for the bullets in an unordered list
  ▸ Example:
    ```
    <li style = "list-style-image:url(bird.jpg)">
    ```

▸ On ordered lists - list-style-type can be used to change the sequence values

Property value	Sequence type	First four
Decimal	Arabic numerals	1, 2, 3, 4
upper-alpha	Uc letters	A, B, C, D
lower-alpha	Lc letters a, b, c, d	
upper-roman	Uc Roman	I, II, III, IV
lower-roman	Lc Roman	i, ii, iii, iv

▸ ✎ try: sequence_types.html and display

▸ CSS2 has more, like lower-greek and hebrew

Colors

▸ Color is a problem for the Web for two reasons:
  1. Monitors vary widely
  2. Browsers vary widely

▸ There are three color collections
  1. There is a set of 16 colors that are guaranteed to be displayable by all graphical browsers on all color monitors

black	000000	green	008000
silver	C0C0C0	lime	00FF00
gray	808080	olive	808000
white	FFFFFF	yellow	FFFF00
maroon	800000	navy	000080
red	FF0000	blue	0000FF
purple	800080	teal	008080
fuchia	FF00FF	aqua	00FFFF

  2. There is a much larger set, the Web Palette
     ▸ 216 colors
     ▸ Use hex color values of 00, 33, 66, 99, CC, and FF

Colors (cont.)

3.   Any one of 16 million different colors

▸ The `color` property specifies the foreground color of elements

```
<style type = "text/css">
  th.red {color: red}
  th.orange {color: orange}
</style>
…
<table border = "5">
  <tr>
    <th class = "red"> Apple </th>
    <th class = "orange"> Orange </th>
    <th class = "orange"> Screwdriver </th>
  </tr>
</table>
```

▸ The `background-color` property specifies the background color of elements
▸ SHOW back_color.html and display

Alignment of Text

▸ The `text-indent` property allows indentation

  ▸ Takes either a length or a % value

▸ The `text-align` property has the possible values, `left` (the default), `center`, `right`, or `justify`

▸ Sometimes we want text to flow around another element - the `float` property

  ▸ The `float` property has the possible values, `left`, `right`, and `none` (the default)

  ▸ If we have an element we want on the right, with text flowing on its left, we use the default `text-align` value (`left`) for the text and the `right` value for `float` on the element we want on the right

Alignment of Text (cont.)

```
<img src = "c210.jpg"
        style = "float: right" />
```

▸ Some text with the default alignment - left
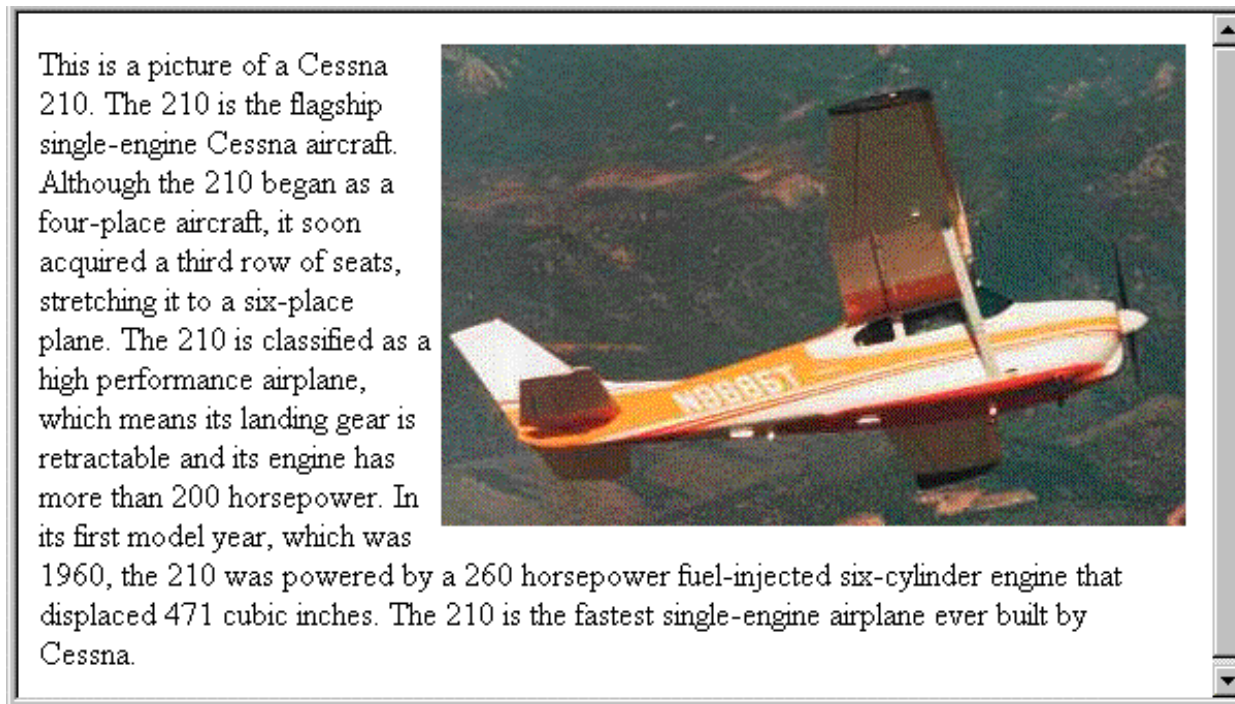
The Box Model

▸ Borders – every element has a `border-style` property

  ▸ Controls whether the element has a border and if so, the style of the border

  ▸ `border-style` values: `none`, `dotted`, `dashed`, and `double`

  ▸ `border-width` – `thin`, `medium` (default), `thick`, or a length value in pixels

  ▸ Border width can be specified for any of the four borders (e.g., `border-top-width`)

  ▸ `border-color` – any color

  ▸ Border color can be specified for any of the four borders (e.g., `border-top-color`)

  ▸ ☞ try: `borders.html` and display

The Box Model (continued)

▸ Margin – the space between the border of an element and its neighbor element

▸ The margins around an element can be set with margin-left, etc. - just assign them a length value

```
<img src = "c210.jpg " style = "float: right;
    margin-left: 0.35in;
    margin-bottom: 0.35in" />
```
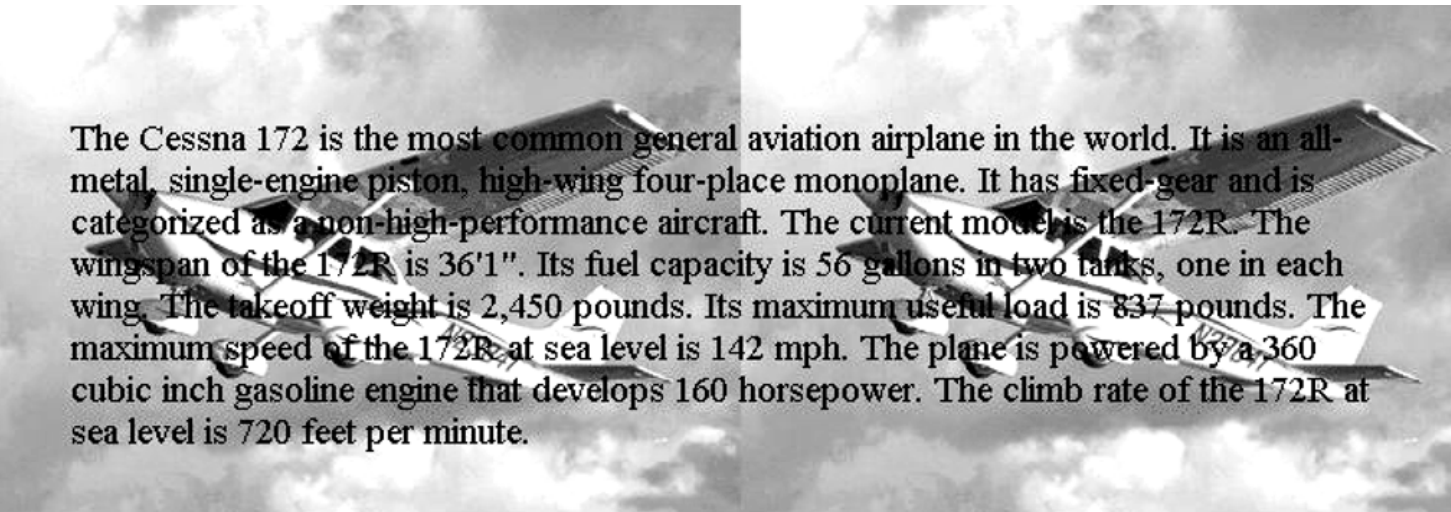
The Box Model (continued)

▶ Padding – the distance between the content of an element and its border

   ▸ Controlled by `padding`, `padding-left`, etc.

   ▸ 🛍 try: `marpads.html` and display

Background

▶ **The** `background-image` **property**

▶ **Repetition can be controlled**

　▶ `background-repeat` **property**

　▶ **Possible values:** `repeat` **(default),** `no-repeat`**,** `repeat-x`**, or** `repeat-y`

　▶ `background-position` **property**

　　▶ **Possible values:** `top`**,** `center`**,** `bottom`**,** `left`**, or** `right`

The Cessna 172 is the most common general aviation airplane in the world. It is an all-metal, single-engine piston, high-wing four-place monoplane. It has fixed-gear and is categorized as a non-high-performance aircraft. The current model is the 172R. The wingspan of the 172R is 36'1". Its fuel capacity is 56 gallons in two tanks, one in each wing. The takeoff weight is 2,450 pounds. Its maximum useful load is 837 pounds. The maximum speed of the 172R at sea level is 142 mph. The plane is powered by a 360 cubic inch gasoline engine that develops 160 horsepower. The climb rate of the 172R at sea level is 720 feet per minute.

The `<span>` and `<div>` tags

▶ One problem with the font properties is that they apply to whole elements, which are often too large

  ▶ Solution: a new tag to define an element in the content of a larger element - `<span>`

  ▶ The default meaning of `<span>` is to leave the content as it is

```
<p>
Now is the <span> best time </span> ever!
</p>
```

  ▶ Use `<span>` to apply a document style sheet to its content

```
<style type = "text/css">?
    bigred {font-size: 24pt;
            font-family: Ariel; color: red}
</style>
<p>
    Now is the
        <span class = "bigred">
    best time </span> ever!
</p>
```

The `<span>` and `<div>` tags (cont.)

Now is the **best time** ever!

Done   My Computer
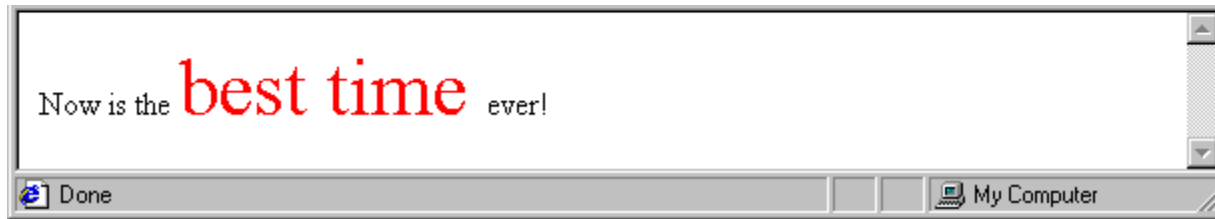
▶ The `<span>` tag is similar to other HTML tags, they can be nested and they have `id` and `class` attributes

▶ Another tag that is useful for style specifications: `<div>`
  ▶ Used to create document sections (or divisions) for which style can be specified
    ▶ e.g., A section of five paragraphs for which you want some particular style

Conflict Resolution

▸ When two or more rules apply to the same tag there are rules for deciding which rule applies

▸ Document level

  ▸ In-line style sheets have precedence over document style sheets

  ▸ Document style sheets have precedence over external style sheets

▸ Within the same level there can be conflicts

  ▸ A tag may be used twice as a selector

  ▸ A tag may inherit a property and also be used as a selector

▸ Style sheets can have different sources

  ▸ The author of a document may specify styles

  ▸ The user, through browser settings, may specify styles

▸ Individual properties can be specified as important

Precedence Rules

‣ From highest to lowest

1. Important declarations with user origin
2. Important declarations with author origin
3. Normal declarations with author origin
4. Normal declarations with user origin
5. Any declarations with browser (or other user agent) origin

Tie-Breakers

▶ Specificity

1. id selectors
2. Class and pseudo-class selectors
3. Contextual selectors
4. General selectors

▶ Position

▶ Essentially, later has precedence over earlier

Exercise

▶ Go to *Exercise/img_tree.png* to obtain an image

▶ Use CSS (document level) to create a page that looks like this:

Markup Validation

Introduction

▸ Best viewed with Internet Explorer

▸ Best viewed with Netscape

▸ Best viewed with …

Best Viewed

This website is best viewed using Internet Explorer 6.x or higher, Netscape Communicator 7.x or higher, or their equivalent browser software on PC's. On MAC's it is best viewed with Internet Explorer 5.x or higher or Netscape 7.x or higher. If your browser is older, it may not present JavaScript, Cascading Style Sheet, colors, text formatting and other features properly. You can determine the version of your browser by choosing Help at the top of your browser and About (your browser).
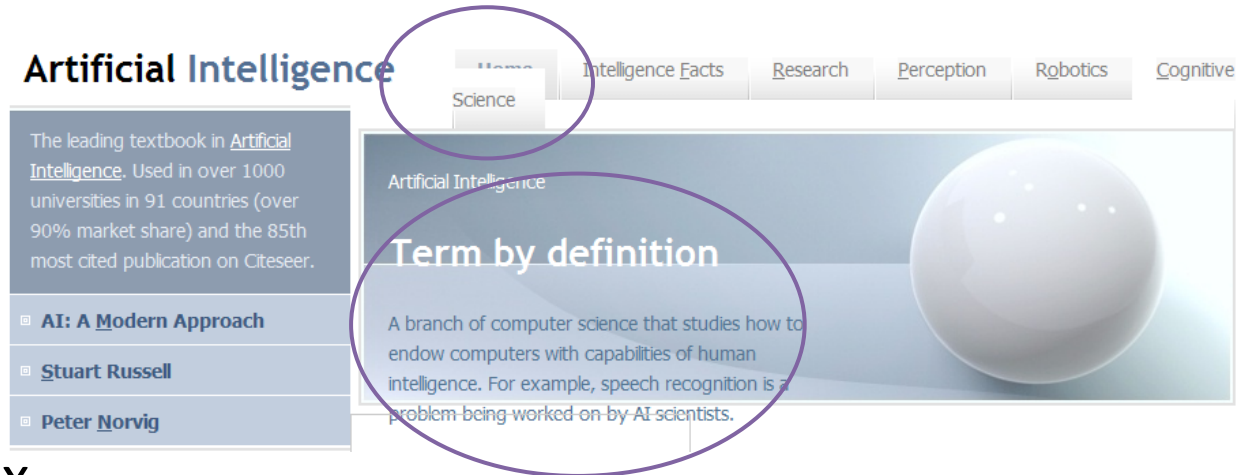
You may download the latest version of a browser for free by clicking on a link below:

- Microsoft
- Mozilla Firefox

Example (1)
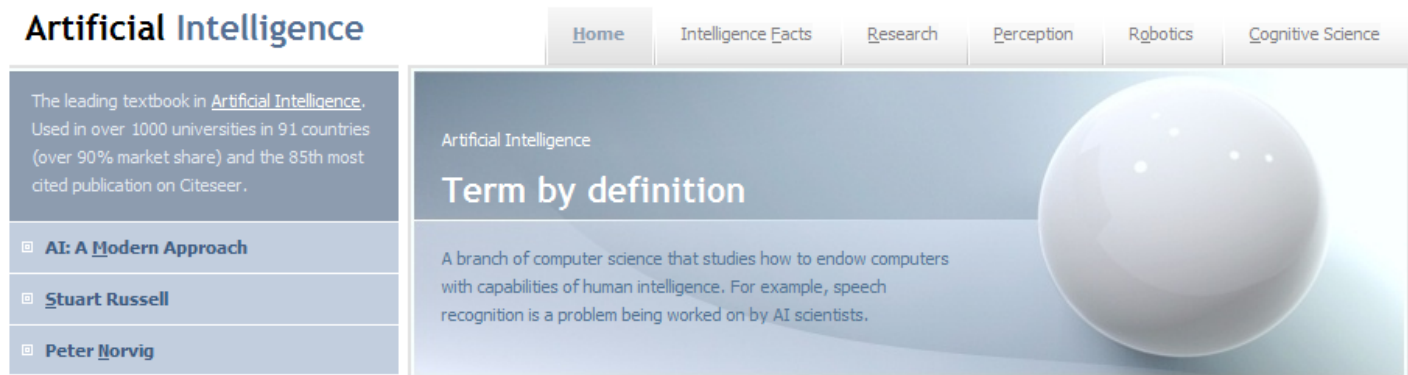
▸ http://www.solucija.com/preview/Artificial_Intelligence

▸ IE

▸ Firefox

Example (2)

- http://www.bangkoktourist.com/index_thai.php
- IE
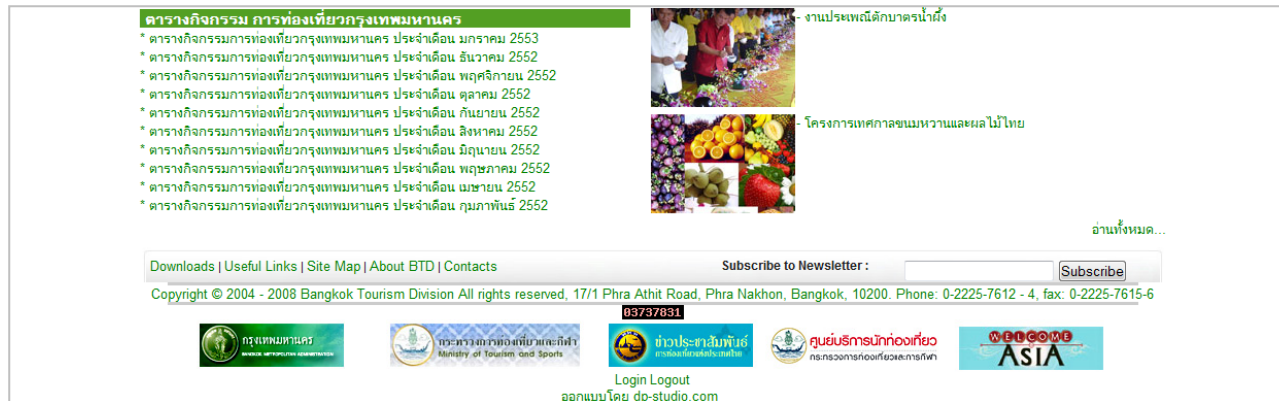
- Firefox
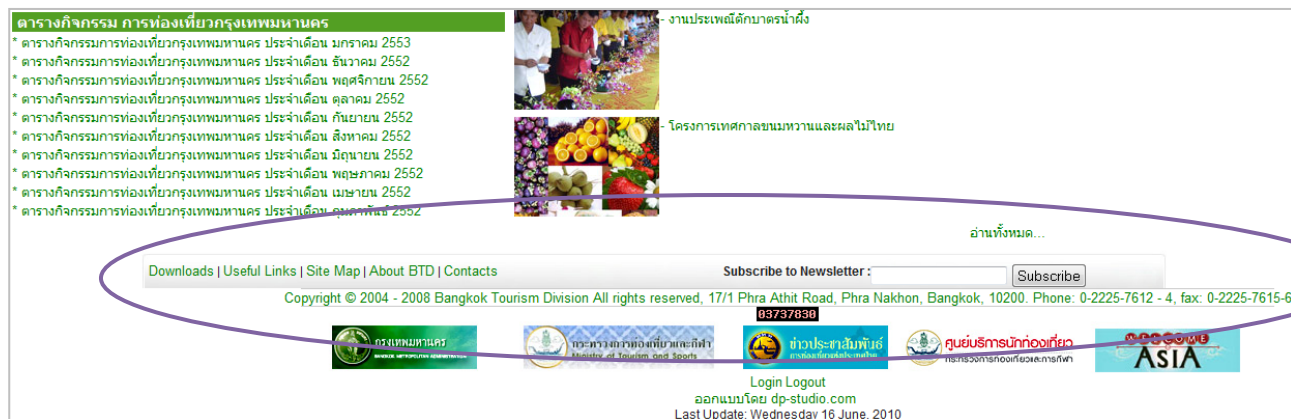
W3C Standard and Validation

http://validator.w3.org/

http://validator.w3.org/feed/

http://jigsaw.w3.org/css-validator/

http://www.validome.org/xml/

```xml
<?xml version="1.0"
<quiz>
 <question>
 Who was the forty-second
 president of the U.S.A.?
 </question>

 <answer>
 William Jefferson Clinton
 </answer>
<!-- Note: We need to add
    more questions later.-->
</quiz>
```
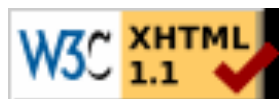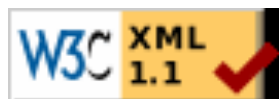XML

```xml
<?xml version="1.0"
<!DOCTYPE html PUBLI
<html xmlns="http://www.w3
 <head><title>XYZ</title>
 </head>
 <body>
 <p>
  voluptatem accusantium do
  totam rem aperiam eaque
 </p>
 </body>
</html>
```
XHTML

```xml
<?xml version="1.0"/>
<xsl:stylesheet xmln
<!-- created 2005-12-12-->
 <xsl:include href="xslt_
 <xsl:output method="xml"
 <xsl:template match="/">
 <root>
 Heuristic:<xsl:value-of
 <p>The leading manufact
 </root>
 </xsl:template>
</xsl:stylesheet>
```
XSLT

XHTML

▸ XHTML, a W3C Recommendation, is basically HTML written to XML rules, without sloppiness found in many HTML pages on the Web.

▸ Browsers are forgiving of errors, which encourages posting of less-than-perfect Web pages.

▸ Even if tags are missing or a document is badly structured, Internet Explorer and other browsers try their best to display the content.

▸ XHTML, by using a more perfect structure, ensures quality documents are posted and will display as expected in the browsers.

Authoring XHTML

▶ The major rules for conforming the XHTML standard:

  ▶ Start your XHTML documents with an XML declaration (optional).

  ▶ Include a DOCTYPE declaration to one of these document types: *strict*, *transitional*, or *frameset*.

  ▶ One root tag must surround the rest of the tags in your XHTML document. The <html> begin tag is above your first tag, and the </html> end tag is after everything.

Authoring XHTML (cont.)

▸ Use HTML 4.0 tags but keep all begin and end tags lowercase (h2, not H2).

▸ Every begin tag has an end tag, and the begin and end tags match in name and are lowercase.

▸ Any empty tags must include an appropriate ending slash, as in <br />. You can alternatively replace the empty tag format with begin and end tags, thus
</br>.

▸ Attributes must have a name and value pair.

▸ Attributes values must be enclosed in quotes.

XHTML DOCTYPE Declarations

▸ ***Transitional***: General Rule

    ▸ <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional //EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

▸ ***Strict***: Takes more time and effort in exchange for very little progress toward XML

▸ ***Frameset***: Only use if your Web layout includes a frameset.

XHTML DOCTYPE "Strict"

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
 <html xmlns="http://www.w3.org/1999/xhtml" lang="en" xml:lang="en">
    <head>
            <title>Title</title>
     </head>
    <body>
            <!-- ... body of document ... -->
    </body>
</html>
```

XHTML DOCTYPE "Transitional"

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" lang="en" xml:lang="en">
    <head>
            <title>Title</title>
     </head>
    <body>
            <!-- ... body of document ... -->
    </body>
</html>
```

Validating XHTML – Sample Files

▶ http://validator.w3.org

▶ Find the website that can pass the XHTML transitional and strict validation

▶ *Try the sample files*

Example

- Transitional
  - o http://www.w3.org/XML/
  - o http://www.acm.org/crossroads/

- Strict
  - o http://canada.gc.ca/home.html

- http://www.w3schools.com/xHTML/xhtml_validate.asp

- Another Example
  - o http://www.w3schools.com/site/site_validate.asp