

Final Review for CMT111-01/-M1

***Most of the important key points we've covered
(I hope)***

6 May 2011

Writing an HTML Document

CMT111-01/-M1—6May2011—2

- **Sketch your Web page**
 - **Storyboarding**
 - Layout
 - content
- **Enter structuring tags for the file**
 - Create the document skeleton
- **Enter the head section elements**
 - hidden info about the page
 - Page title
 - meta tags: author, key words, page description, encoding
- **Enter content inside the *body* element**
 - Marking up paragraphs, lists, etc, as appropriate
- **Preview the Web page**
 - load page in your browser to see what it looks like
- **Test links**
 - If they don't work, fix them

Web Pages and Tags

CMT111-01/-M1—6May2011—3

- **Web page is a plain text file**
 - all content in file is labeled with tags, e.g. paragraphs, headers, etc
- **Tag is enclosed in brackets: <tag>**
 - Most tags come in pairs, an opening and closing tag surrounds each part of the content
 - closing tag denoted by slash, /
 - e.g. <p> sundry paragraph text </p>
- **Some tags have attributes: adds information to specific tag**
 - <tag attribute-property="attribute-value">
 - e.g. <div id="navigation"> sundry content </div>
- **Three essential tag pairs every page needs [plus Doctype]**
 - <html>...</html> [*First* and *Last* tags on page; identifies the document as being html]
 - <head>...</head> [contains title, information about page and processing instructions]
 - <body>...</body> [contains visible page content]

A Skeleton Web Page Has Three Essential Tag Sets... ...after a !DOCTYPE declaration

CMT111-01/-M1—6May2011—4

- **!DOCTYPE declaration goes first**

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

- **<html> First and last tags on page </html>**

- **<head> Descriptive content that's hidden from view</head>**

- Page title
- Meta tags
 - Author
 - Description
 - key words
 - encoding
- Styles and scripts (program snippets)
 - Each of these is wrapped in its own special tag set

- **<body> Web content </body>**

- Text
- Pictures
- Embedded media—audio, video, flash
- Web-form elements

The Basic Web Page

CMT111-01/-M1—6May2011—5

The code on the left produces the web page below

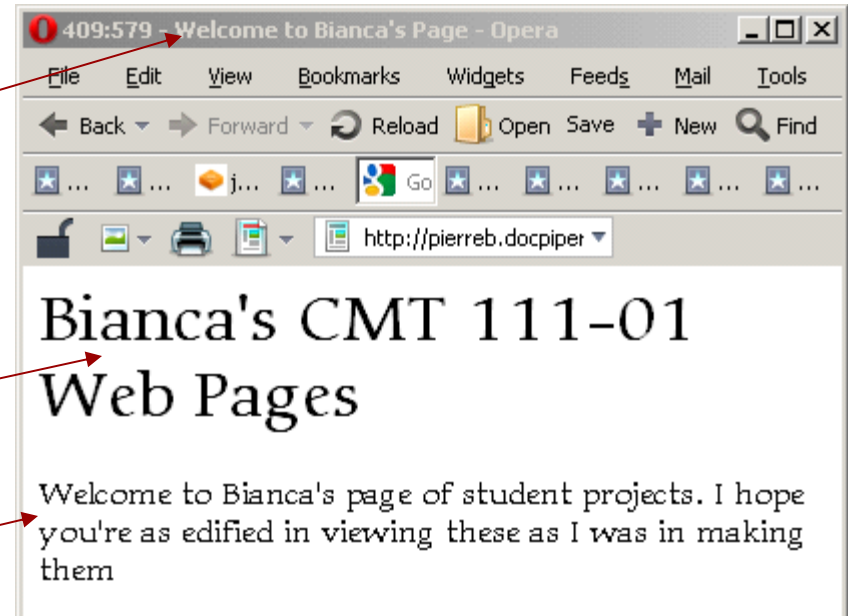
```
<!DOCTYPE html
  PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" lang="EN">

  <head>
    <title>Welcome to Bianca's Page</title>
    <link rel="stylesheet" href="style.css" type="text/css" />
  </head>

  <body>
    <h1>Bianca's CMT 111-01 Web Pages</h1>

    <p>Welcome to Bianca's page of student projects. I hope
    you're as edified in viewing these as I was in making them</p>

  </body>
</html>
```



Three Kinds of Tags

CMT111-01/-M1—6May2011—6

- **Block-level tags** are paired around content and define a box of some sort

- `<p>` this is a paragraph with some text!`</p>`
- the box can be styled and positioned as one sees fit
 - e.g. you can style the font, colors, etc inside the box
 - you can add borders—or not—around the box
 - you can position the box as you see fit
 - all this positioning and styling should be done via CSS [will cover soon]

This is a paragraph with some text!

- **In-line tags** are paired around content, but do not define a box, they only label something inside a containing box

- ``My website``
- ...let me ``emphasize`` one thing... `...let me emphasize one thing...`

- **Empty tags** are single tags that don't themselves contain content. They must, however, still be closed

- `
`: a line break
- `<hr />`: a horizontal rule
- ``

Marking up an HTML Document

CMT111-01/-M1—6May2011—7

- **Basic Mark up tags**

- Headlines [`<hn>headline</hn>` — n is a number from 1 to 6]
 - 1 is most important (largest) headline; 6 is least important (smallest)
- Paragraphs [`<p>paragraph text</p>`]
- Line breaks [`
` — “empty” tag; still *must be* closed]
- Lists
 - Ordered [``]
 - Unordered [``]

- **Tags should be all *lower case***

- **All tags should be closed**

- Tag pairs, like `<p> </p>`
- Single tags, like `
` or ``
- / denotes closing

Empty Elements Stand Alone

CMT111-01/-M1—6May2011—8

- **Formatting help**
 - `
` for line break
 - `<hr />` for horizontal rule
- **Insert objects into document**
 - `` to place an image on a page
 - `<input />` to place an input control on a page, e.g. radio button, text box, etc.
- **Provide specific information**
 - `<meta />` information about document in the `<head>` section
 - `<link />` links to supporting documents, like style sheets
- **Empty elements must be closed**
 - place a space after the tag, then `/>`

Character References

CMT111-01/-M1—6May2011—9

- **How do you put non-keyboard characters on your page?**
 - things like © [copyright], ™ [registered trade mark], ° [degree sign]
- **use character references**
- **Two kinds of character references**
 - HTML references: *&name;*
 - must begin with **&** and end with **;**
 - **•** makes a bullet: •
 - **©** makes copyright: ©
 - numerical references: *&#nnnn;*
 - must begin with **&#** and end with **;**
 - much more extensive—can add Kanji, Sanskrit, etc
 - **•** makes a bullet: •
 - **©** makes copyright: ©
- **Partial list:** http://www.w3schools.com/tags/ref_symbols.asp
- **More extensive discussions:**
 - Jukka Korpela: <http://www.cs.tut.fi/~jkorpela/chars.html>
 - Alan Wood: <http://www.alanwood.net/unicode/index.html>
 - Simple chart on Evolt.org: <http://evolt.org/node/21234/>

Creating Links

CMT111-01/-M1—6May2011—10

- **Four basic types of links**
 - **Internal** links
 - page-to-page within web site
 - page-to-page within web site, but to a different directory
 - **External** links
 - website-to-website
 - **Jump** (or anchor) links
 - section-to-section within single page
 - **e-mail** links
 - To e-mail program
 - unreliable—a contact form works better
- **Additions to links**
 - Open links in new windows or tabs [target=“_blank”]
 - Add link titles (accessibility issue) [title=“some descriptive title”]
 - e-mail link enhancements (adding subject, bcc, etc.)

Creating Internal Links

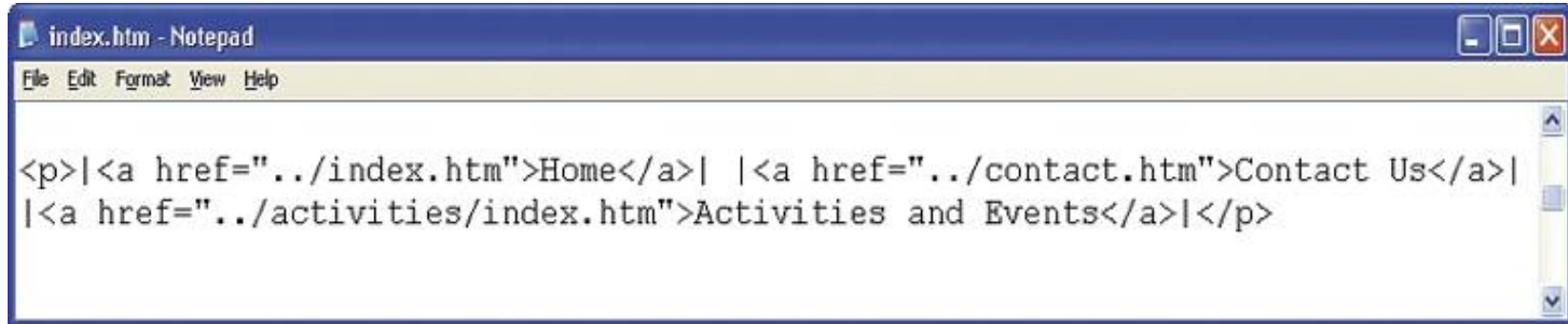
CMT111-01/-M1—6May2011—11

- An *anchor tag* or link tag surrounds information that users click to navigate to another location
- *Attributes* and *values* specify the tag's behavior or function
 - href
 - title
 - target
- **Internal links link to files within the same site**
 - usually relative links
- **Tag pattern:**
 - `text related to url`
 - url *must* be enclosed in quotation marks
 - url is link to
 - another page [*internal* link]
 - another part of the page [*jump* link]
 - another website [*external* link]
 - e-mail client [*mailto* link]

- a stands for *anchor*
- href stands for *Hypertext Reference*

Creating Links to Other Directories

CMT111-01/-M1—6May2011—12



```
index.htm - Notepad
File Edit Format View Help
<p>|<a href=" ../index.htm">Home</a>| |<a href=" ../contact.htm">Contact Us</a>|
|<a href=" ../activities/index.htm">Activities and Events</a>|</p>
```

- **Current directory either has nothing (or './') before file or directory name**
 - href="index.html" or href="./index.html"
- **Directory one step up is denoted ../**
 - href="../index.html"
- **Directory one step down is /down**
 - href="down/index.html" or href="./down/index.html"

- one-step up called *parent* directory
- one-step down called *child* directory

Creating External Links

CMT111-01/-M1—6May2011—13

- **Create links to files on *external web sites* to provide additional resources and information to your site visitors**
- **Absolute URL is the complete URL**
 - Protocol [http://]
 - subdomain or server name [www, blog, forum, ...—sometimes unneeded]
 - domain name [mySite.com]
 - Path [/dogs —sometimes unneeded]
 - Filename [myDogs.html]
 - not needed if file is index.html, index.php, or a similar default
- **Examples:**
 - http://www.google.com
 - http://www.mySite.com/dogs/myDogs.html

Jump Links

CMT111-01/-M1—6May2011—14

- **Jump links connect one section on a page to another section on that same page**
- **Two things required for a jump link**
 - The link itself
 - A link *target*, an element on the page with an *id*
 - (usually an `<hx>` or `<div>`)
- **Jump links reference the *id* of the element demarking the page section**
- **For example:**
 - The target, `<h1 id="top">`
 - is reached via `to top`

e-mail Links

CMT111-01/-M1—6May2011—15

- **E-mail links will (allegedly) open your e-mail program with the “to:” field filled in**
 - Not all users have browser configured properly (e.g. my brother)
 - Many users might be on someone else’s computer
 - University
 - Public Library
 - Parents’ house(s)
- **me@mySite.com**
 - Produces something like
contact me: *me@mySite.com*
- **Can enhance your e-mail links to automatically** [won’t be tested on this]
 - Add a subject line
[?subject=free chocolate]
 - Carbon copy another recipient
[?subject=free chocolate&cc=joe@mySite.com]

Other Things About Links

CMT111-01/-M1—6May2011—16

- **You can add a title to the link**

- A little pop-up appears when you hover link with mouse giving link title
- Add the attribute: *title*="my link title" after the href attribute

```
<a href="http://mysite.com" title="a link to my awesome site">my awesome site</a>
```

- **Force the link to open in a new window or tab**

- Nominally keeps visitors on your site, but generally annoys people
- Add the attribute: *target*="_blank" after the href attribute
 - can use other attributes, e.g. *target*="new", *target*="sam", etc.

```
<a href="http://mysite.com" title="a link to my awesome site, which will open in a new window" target="_blank">my awesome site</a>
```


Basic HTML Formatting Methods

CMT111-01/-M1—6May2011—17

- **The text on HTML pages can be formatted using the following methods:**
 - text-align [align text inside the element]
 - Left
 - Right
 - Center
 - justified
 - Font Formatting
 - font-family: serif, sans serif, ...
 - font-style: bold, italic, underlined...
 - Block-Level Formatting
 - Break page into boxes, then format each “box” differently
 - Lists
 - Ordered [code> . . .]
 - Unordered [code> . . .]
 - Definition [probably won't get to these, but I use them all the time]

```
<p style="text-align:right;">paragraph  
text of some sort.</p>
```

Formatting Page Elements with HTML

CMT111-01/-M1—6May2011—18

- **Should be using *semantic* markup**
 - Separate content from styling
 - Tags should tell what the specific element is, *not* how to style it
 - e.g. paragraphs, ordered lists, headlines, etc.
- **Only use `` and ``**
 - `` and `<i>` change presentation, but don't convey meaning to mark up [NOTE: I will take points off if you use `<i>` or ``]
 - `<u>` is confusing because, in general, underline indicates a link
 - If it's not a link, *DON'T* underline
- **Style choices, e.g. font-family, font-size, color, etc., should be done using css rules**

Controlling the Default Font in a Document

CMT111-01/-M1—6May2011—19

- In olden days, people placed a `<basefont>` tag in the `<head>` of a document to declare the default font.
 - No longer works—*only* IE still recognizes the tag. <50% of browsers
- Instead, specify font information in a body style-declaration
 - Best to do it in an external .css file
 - OK to do it *embedded* inside the `<head>...</head>` part of each document
- How it looks in the `<head> ... </head>`

```
<head>
... other stuff, like <title> <meta>, etc.
<style type="text/css">
  body{
    font-family: Verdana, Arial, Helvetica, sans-serif;
    font-size: 1em;
    color: #333366;
  }
</style>
</head>
```

Note: opening and closing curly braces

Also, semicolons after each rule

To understand the syntax, please see discussion on CSS styling further along.

Suggestions Regarding Font Selection

CMT111-01/-M1—6May2011—20

- **Selecting and specifying page fonts**
 - Typeface selection rules apply to both print and screen media
 - Use one family for headings, and another family for body text
 - Not a hard-and-fast rule
 - Perhaps it makes things more readable or interesting
 - Do not mix fonts from the same family
 - Maintain consistency
 - Use fonts that are commonly available
 - If the font isn't on the user's computer, the computer makes stuff up, which may or may not be good.
 - *always* provide a default, i.e. generic family
 - serif
 - sans-serif
 - other options: monospace, cursive, fantasy

Note: font names are case sensitive!
Arial and arial are *not* the same!

font-family: Georgia, "Times New Roman", Times, serif;
or
font-family: Verdana, Arial, Helvetica, sans-serif;

About Lists

CMT111-01/-M1—6May2011—21

- **Three kinds of lists**

- Ordered—items numbered consecutively

```
<ol>
  <li>item one </li>
  <li>item three</li>
  <li>item two</li>
</ol>
```

```
1. Item one
2. Item three
3. Item two
```

- Unordered—items have bullet in front of them

```
<ul>
  <li>item one </li>
  <li>item three</li>
  <li>item two</li>
</ul>
```

```
• Item one
• Item three
• Item two
```

- Definition—we didn't study these very much, but I use them all the time

```
<dl>
  <dt>a term </dt>
  <dd>a definition</dd>
  <dd>2nd definition</dd>
</dl>
```

```
a term
a definition
2nd definition
```

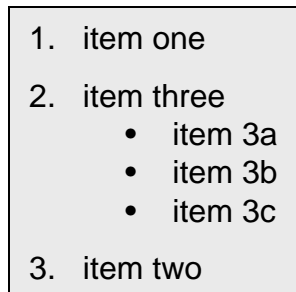
You Can Nest Lists

CMT111-01/-M1—6May2011—22

- **Nesting lists**

- It may be necessary to create sub items within a parent list
 - This process is known as **nesting** a list
- You can nest lists of any type

```
<ol>
  <li>item one </li>
  <li>item three
    <ul>
      <li>item 3a </li>
      <li>item 3b</li>
      <li>item 3c</li>
    </ul></li>
  <li>item two</li>
</ol>
```

- 
- 1. item one
 - 2. item three
 - item 3a
 - item 3b
 - item 3c
 - 3. item two

Planning Graphics Use

CMT111-01/-M1—6May2011—23

- **Use supported file formats**
 - .gif — graphics image format
 - Good for simple drawings with *limited* colors
 - .jpg or .jpeg — joint photography experts group
 - Best for photographs or lots of colors
 - .png — portable network graphics
 - Relatively new, but even recent IE versions recognize it
 - Lots of colors
 - Transparent backgrounds
- **Add alternate text** [alt=“descriptive text for accessibility”]
 - Descriptive text for accessibility
- **Keep file size small**
 - Rarely need anything larger than 600x400px or 50 kB
 - Generally need to edit images
 - Resize
 - crop
- **Use graphics wisely**
 - Too much gets confusing

Adding Images to Web Page

CMT111-01/-M1—6May2011—24

- **Use the tag**
 - A single tag, so *must* be closed at back end
- **One required attribute**
 - The file source (**src**), i.e. where to find the image file
 - In images we use **src**
 - In links we use **href**
 - In css, we use **url**
- **One semi-required attribute**
 - alt: it provides a description for screen readers
- **Several useful attributes**
 - height=, width= : size of figure; their use improves load times considerably
 - vspace=, hspace= : add space around the picture [improve readability]
 - align= ; describe how text flows around picture
 - Whether or not there is a border around the picture
 - border="0" is *very* useful for keeping image links from having ugly blue borders

More about Image Tags and their Attributes

CMT111-01/-M1—6May2011—25

- **Because pictures differ one from another, often need to specify attributes for each img tag**
 - Could make up standard sizing and then use css for all images
 - Can still use css for some attributes, like space around the figure
- **Attributes in a tag use old-style coding**
 - property="value"
 - don't need to separate property/value pairs with commas
 - don't need to specify px for attributes with units, pixels are assumed
- **Basic format**

```

```

Background Images and Using Images for Links

CMT111-01/-M1—6May2011—26

- **You can place a background image on a web page**

- be careful it doesn't make your page unreadable

```
body{  
    background-image: url(path/to/pics/nameOfPic.gif);  
}
```

- **You can place an image inside an anchor**

- makes the image act like a link
- clicking image takes you to new page, web site, etc.
- To make a logo image take you to the home page:

```
<a href="index.html"></a>
```

Styling Web Pages with CSS

CMT111-01/-M1—6May2011—27

- **CSS stands for cascading style sheets**
- **A style sheet contains rules for how you want html elements to be rendered**
 - how they will look on the page
 - How they will be laid out on the page
- **If you have multiple rules applying to the same item, the last defined rule will take precedence. This is the *cascade* part.**
- **There are *three ways* to specify (or implement) style using css**
 - **In-line**
 - Applied to specific element
 - **Embedded**
 - A set of rules inside the <head> tag
 - Applies *only* to page in which it's embedded
 - **External**
 - Link in the <head> tag to a file that contains style rules
 - Far and away the best method
 - A single .css file can apply to *all* pages on a site, if linked properly

Basics of CSS Rules—1

CMT111-01/-M1—6May2011—28

- **CSS rules have three parts**
 - **Selector**: the name of the tag being modified
 - **Property**: the specific property you wish to modify
 - **Value**: the value you wish to have for the property
- **The rules are written slightly differently for in-line styling than for the other two methods**
- **In-line**
 - Property and value placed inside the opening tag as a *style* attribute
`<tag style="property: value;">`
 - Each *property:value* pair
 - *must be* inside quotation marks
 - *must be* separated by a colon (:)
 - *must be* terminated by semicolon (;)
 - Can have multiple *property:value* pairs inside one set of quotation marks, each separated by semicolons.
`<h1 style="font-size: 1.75 em; text-align: center; color: #006600;">`

Basics of CSS Rules—2

CMT111-01/-M1—6May2011—29

- **embedded and external style rules have same syntax**

```
Selector      {  
                property1: value1;  
                property2: value2;  
                etc.  
            }
```

- **Each property:value pair**

- *must be* between curly braces
- *must be* separated by a colon (:)
- *must be* terminated by semicolon (;)

- **Many good on-line resources available**

- W3C Schools: <http://www.w3schools.com/css/default.asp>
- HTML Dog: <http://www.htmldog.com/>

Basics of CSS Rules—3

CMT111-01/-M1—6May2011—30

- **Embedded CSS rules in document head**

```
<head>
```

```
...
```

```
<style type="text/css">
```

```
  body    {text-align: center;
           background-color: #ffffdd;
           color: #660000;
           }
```

Styling the *body* element

```
  p      {text-align: left;
           font-family: Arial, Helvetica, sans-serif;
           ...;
           }
```

Styling *paragraph* elements

```
...
```

```
</style>
```

```
...
```

```
</head>
```

Basics of CSS Rules—4

CMT111-01/-M1—6May2011—31

- **Linking to an external CSS file**

```
<head>
```

```
...
```

```
<link rel="stylesheet" href="../../lgp.css" type="text/css" />
```

```
...
```

```
</head>
```

- **The *rel*, *href* and *type* attributes are required**

- Can have others, e.g. *title*
- Can have multiple stylesheets for printing, mobil media, etc.
- Can have alternate stylesheets that apply in certain instances
 - e.g. might have a javascript that tests something then chooses an alternate style sheet to suit the result of the test

A Word about Color

CMT111-01/-M1—6May2011—32

- **Color is generally specified in terms of red/green/blue**
 - hexadecimal most common: #rrggbb
 - 1st two digits—amount of red; 2nd—amount of green; 3rd—amount of blue
 - digits from 0 to F
 - each hexadecimal digit represents 1–15, so need a–f above 9
 - If the two digits are the same, can drop the second from each group
 - e.g. #ffaacc and #fac give the same color
 - rgb(rrr, ggg, bbb) [no idea why the book uses this; no one else does]
 - digits run 0 up to 255
- **Maximum amount of color is ff or 255 (ff in hex=255 in decimal)**
- **minimum color is 0, i.e. no color at all**
- **In some cases, can use color words**
 - white, black, brown, green, etc.
 - generally a bad idea because you don't know how any given browser defines a given key-word color

Basic Colors

CMT111-01/-M1—6May2011—33

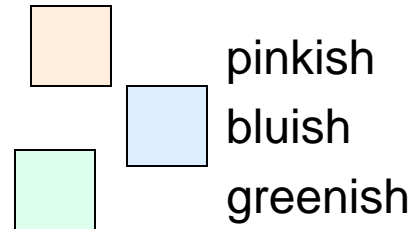
- **Basic colors (almost certain to be on a test)**
 - #ff0000 will be red
 - #00ff00 will be green
 - #0000ff will be blue
 - #ffffff will be white
 - #000000 will be black
- **All other colors are a mixture of the three**
 - use a color picker on line
 - <http://johndyer.name/lab/colorpicker/>
 - http://images.my-addr.com/online_color_picker_tool_pick_color_html_hex_rgb.php
 - <http://www.pagetutor.com/colorpicker/index.html>
 - or one in Photoshop/Fireworks/Illustrator [also in DW]

Colors with CSS

CMT111-01/-M1—6May2011—34

- **background color**

- background-color: #ffeedd will be
- background-color: #ddeeff will be
- background-color: #ddffee will be



- **color means color of text**

- color: #ff0000 is red text
- color: #00aa00 is green text
- color: #0000ff is blue text

- **Can also have color on things like borders**

And a Short Mention of Fonts

CMT111-01/-M1—6May2011—35

- **A font is a specific way to render text characters**
- **There are five basic font families, although only two matter much**
 - serif (e.g. Times New Roman):
 - has little lines or serifs hanging on various characters.
 - Allegedly serif fonts make reading printed matter easier
 - sans-serif (e.g. Arial)
 - no little serifs hanging on the text
 - allegedly sans-serif makes reading on a monitor easier
 - monospace (e.g. Courier)
 - like an old typewriter. every character takes up the same amount of space
 - *cursive (e.g. Lucida Handwriting)*
 - *supposed to mimic fancy handwriting*
 - **FANTASY (E.G. ALGERIAN)**
 - **A BONE FOR SCI-FI GEEKS, I SUPPOSE**

Fonts Come in Various Sizes

CMT111-01/-M1—6May2011—36

- **Absolute Sizes**
 - no rational person uses these on the web
 - points and picas are for print
- **Relative Sizes**
 - per cent
 - em—basically the size of an upper-case M in whatever size initially chosen
- **Most web people size in pixels (px)**
 - not such a good idea because older IE can't resize for people needing it
- **Can also use key words (7): e.g. medium, x-small, xx-large, etc.**
- **Default for most browsers is ~16px**
 - too large for normal use
 - size at 80–85% in body
 - then resize individual elements with em

You Can Change all Kinds of Things

CMT111-01/-M1—6May2011—37

- ***Kerning* is letter spacing**
 - letter-spacing: 0.4em;
- ***Tracking* is word spacing**
- ***Leading* is space between lines**
 - line-spacing: 2em; would be double spacing
 - line-spacing: 1.2 em; makes it easier to read than 1 em
- **Learn about various typography issues on line**
 - Web Style Guide:
 - <http://webstyleguide.com/wsg3/index.html>
 - Especially Ch. 8:
 - <http://webstyleguide.com/wsg3/8-typography/index.html>

Setting Link Properties

CMT111-01/-M1—6May2011—38

- **Style link properties with *pseudo classes***

```
a:link          {
    font-weight: bold;
    color: #0000ff;
    text-decoration: underline;
}
a:visited      {
    color: #8800ff;
    text-decoration: underline;
}
a:hover        {
    font-style: italic;
    font-weight: bold;
    color: #009900;
    text-decoration: underline;
}
a:active       {
    color: #ff6633;
    text-decoration: underline;
}
```

- **The `:link`, `:visited`, `:hover` and `:active` are called *pseudo-classes***
 - `:link` — normal view of a link; generally dark blue (`#0000ff`)
 - `:visited` — view of a link one has visited; generally violet (`#8800ff`)
 - `:hover` — view of link when one hovers the mouse over it: generally green (`#009900`), sometimes with change in background color
 - `:active` — view of a just-clicked link; generally red (`#ff6633`). Also generally of little use.
- **Pseudo-classes must be listed in specific order to work properly**
 - mnemonic: LoVeHAte

The Box Model

CMT111-Q1/-M1—6May2011—39

- **The box model describes how block-level content is displayed**
- **Each block is essentially a box**
- **Each box is described by four characteristics**
 - **width**: the width of the content in the box, i.e. the text
 - **padding**: the space between the content and the box's border
 - **border**: the line around the box
 - can have width
 - or can be invisible
 - **margin**: the distance between boxes
- **Boxes can have other characteristics**
 - position
 - color
 - visibility
 - etc.

What is a Box?

CMT111-01/-M1—6May2011—40

- **Any block-level element can define a box**
 - <p>
 -
 -
 - <address>
 - <blockquote>
 - <div>
 - <h1>
- **You can style a box as you see fit**
 - position
 - size
 - background image or color
 - font-family, font-size, color, etc.
 - borders, margins, borders...any or none
 - etc.
- **Modern practice manipulates *boxes* for formatting and layout**
 - tables are so last century

Box Dimensions

CMT111-01/-M1—6May2011—41

- **Four dimensions for box**

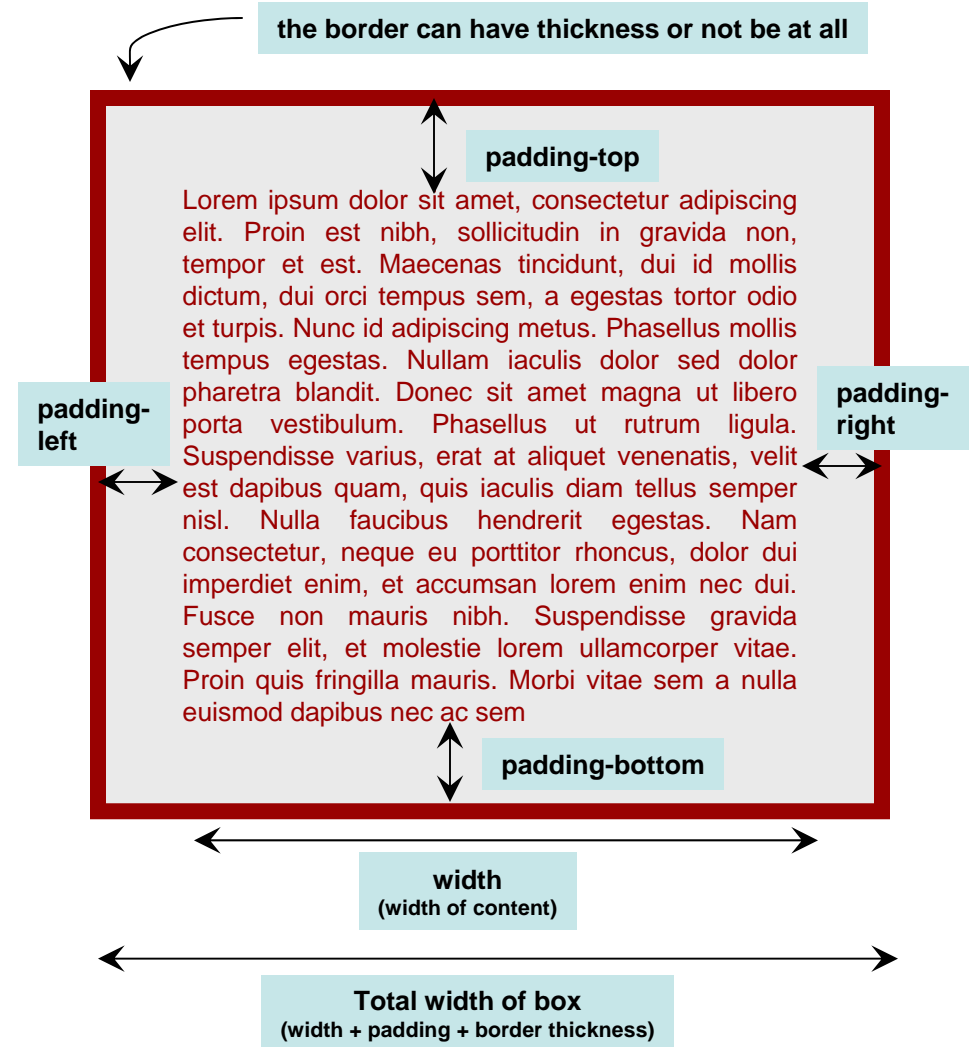
- **width** of content
- **padding** between content and border
- **border**
 - can have width
 - can be invisible
- **margin**: spacing from other boxes

- **Total dimension of a box:**

width + padding + border

- **Older versions of IE used a different box model**

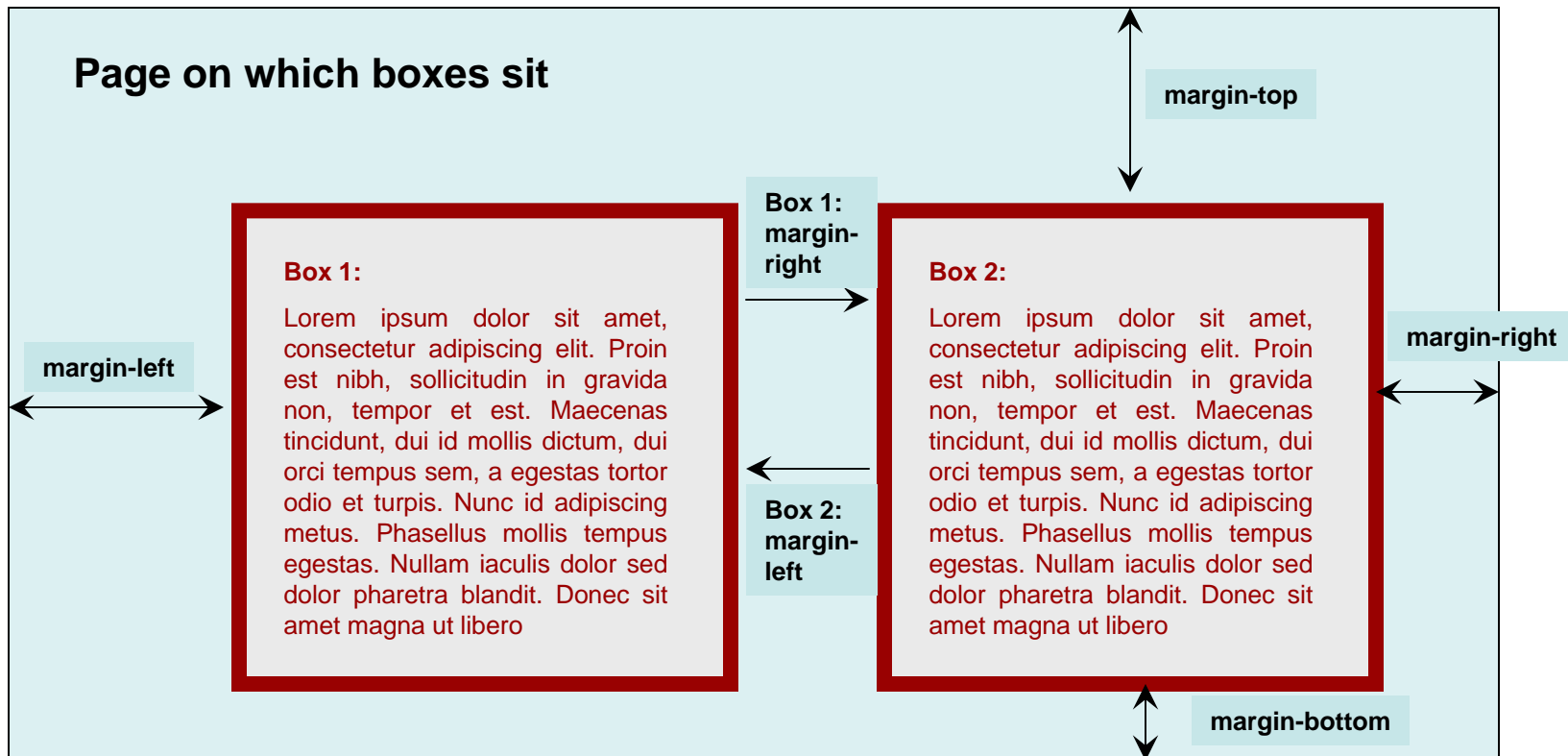
- created much angst in web-dev community
- contributed to poor practices
- fortunately, old IE versions are mostly dead



Boxes Relative to Each Other and the Page

CMT111-01/-M1—6May2011—42

- Margins affect how closely one box sits relative to another or to the page



Applying Margins

CMT111-01/-M1—6May2011—43

- Can apply margin to text by wrapping it inside the *body* element or a *div* element and applying margin to that element:

```
div {  
  margin-left: 40px;  
  margin-right: 40px;  
}
```

- `<div>` is essentially a way to divide web-pages into blocks or boxes
- a basic building block on well-designed pages

- Apply a margin on the right side of an image:

```
img {  
  margin-right: 25px;  
}
```

Using CSS Class Rules

CMT111-01/-M1—6May2011—44

- **Can define a class which will apply each time you specify it inside a tag:**

```
.smallText {  
    font-size: 80%;  
}
```

- **The class declaration is denoted by the ‘.’ before the name**
- **Classes are basically a form of in-line styling—*use sparingly***
- **Show class with class=“className” attribute inside tag**
- **Classes can apply to more than one element**

```
<p class="smallText">This paragraph uses small text</p>  
<h3 class="smallText">This subheading uses small text</h3>
```

Using CSS id Rules

CMT111-01/-M1—6May2011—45

- **Can define an id which applies to *one and only one* tag per page:**

```
#navBar {  
    text-align: center;  
    font-size: 80%;  
}
```

- **The id declaration is denoted by the ‘#’ before the name**
- **use id to demark different major parts of a web page**
 - nav, header, footer, main content, etc.
 - used for css styling and positioning
 - provide anchor points for jump links
- **Show id with id=“idName” attribute inside tag**

```
<h3 id="colleen">Colleen O'Rarry, C.D.</h3>
```

id is Restricted to Only One Element on Each Page

CMT111-01/-M1—6May2011—46

- **only one element on each page can have a given id**

```
<div id="navBar">
  |<a href="index.html">home</a>| |<a href="about.html">about</a>|
  |<a href="products/index.html">products</a>| |<a href="contact.html"> contact</a>|
</div>

<h3 id="dogs">Dogs in My Life</h3>
```

What is a Table?

CMT111-01/-M1—6May2011—47

- **A table is a way to organize data**
 - Rows—student names
 - Columns—quiz grades
 - In theory tables should only be used for tabular data
- **Tables are also used to lay out web pages**
 - Easy to do
 - Easy to control position precisely
 - *Very difficult* to maintain
 - Now there are *much better* ways to do layout using css
 - Table-based layout invented before css was adequately mature
 - Modern designers need to learn css

Essential tag sets for defining a table

CMT111-01/-M1—6May2011—48

tag(s)	function
<code><table>...</table></code>	Defines start and end of table contents
<code><tr>...</tr></code>	Marks contents of each table row
<code><th>...</th></code>	Marks contents of a table heading cell and applies table heading format
<code><td>...</td></code>	Marks contents of a table data cell
<code><caption>...</caption></code>	When coded within a table, formats text to appear as a table caption

- **Must wrap contents of table in `<table> ... </table>` tag set**
- **Wrap each row in `<tr> ... </tr>` tag set**
- **Each column within a row defined by `<td> ... </td>` tag set**
 - Generally have multiple `<td> </td>` tag sets inside a row—*table cells*
 - Table headers [`<th>...</th>` tag sets] tell what each column represents
 - If used, they go on the top row and define the columns

Standard Table Attributes

CMT111-01/-M1—6May2011—49

- **Tables generally use old-style attributes**
 - Attribute=“property”
- **Standard attributes of <table>**
 - width—width of whole table, in pixels
 - cell spacing—spacing between cells in pixels
 - cell padding—spacing between text and borders of cells in pixels
 - align—how to align the table, i.e. center, right, left
- **Standard attributes of <td>**
 - colspan—number of columns the cell spans
 - width—width of cell in pixels
 - height—height of cell in pixels
 - valign—vertical alignment: top, middle, baseline
 - align—center, left or right

Table Example

This code

```
<table border="2" cellspacing="4" cellpadding="4">
  <caption>Table of Dogs</caption>
  <tr>
    <th>Name</th>
    <th>Breed</th>
  </tr>
  <tr>
    <td>Bridget</td>
    <td>Irish Terrier</td>
  </tr>
  <tr>
    <td>Colleen</td>
    <td>Irish Terrier</td>
  </tr>
  <tr>
    <td>Cider Anne</td>
    <td>Golden Retriever</td>
  </tr>
  <tr>
    <td>Brewster</td>
    <td>Golden Retriever</td>
  </tr>
</table>
```

CMT111-01/-M1—6May2011—50

Produces this HTML table

Table of Dogs

Name	Breed
Bridget	Irish Terrier
Colleen	Irish Terrier
Cider Anne	Golden Retriever
Brewster	Golden Retriever

Form Basics

CMT111-01/-M1—6May2011—51

- **All form objects are wrapped inside `<form>...</form>` tag pair**
 - attributes inside the `<form>` tag tell how to process the form
 - id
 - action—what to do with form data, e.g. send it to a php page for processing
 - method—the means to send the data to where it is processed
 - **get**—through the browser (used by Google searches)
 - data appended to a URL as ascii data (not secure)
 - less secure and amount of data are limited to ~8K
 - **post**—a more secure way to send data to your web server
 - sends id/value data for *each* form object in form of associative array to target of the action, inside an encrypted binary file
 - either way, data sent as *name/value* pairs
 - actual form field denoted by name or id
 - value is the data entered in the form field
 - **Example—opening tag of contact form:**

```
<form id="contact" action="./ContactThanks.php" method="post"
enctype="multipart/form-data" onsubmit="return validateContact();" >
```

Key Features of Contact Form

CMT111-01/-M1—6May2011—52

field set—organize the inputs into groups to make form easier to understand.

label—can label form inputs to make them easier to understand

Can also pre-fill in some fields to make them easier to understand

form buttons—
"submit" and "reset"
most common

Text field—enter (or display) a line of text

Text area field—
enter multiple lines of text

The diagram shows an "E-mail Form" with the following fields and annotations:

- To:** myProf@docPiper.com (Text field)
- From:**
 - Name:** your name (Text field)
 - e-mail:** your e-mail address (Text field)
 - Web site:** http:// (Text field)
- Subject:** doc P Query (Text field)
- Message:** Please leave me a message. Thanks! (Text area field)
- Buttons:** send, cancel (Form buttons)

Annotations include arrows pointing from text boxes to the form fields and buttons, and a red box highlighting the "From" section.

<input> Form Object

CMT111-01/-M1—6May2011—53

- **<input> is most common form object**
- **attributes select the <input> type**
 - type="text box"
 - type="radio button"—select just one member of a group
 - each member of group must have same "name" (only time name matters anymore)
 - type="check box"—select any or all members in group
 - type="button"
 - and some others (file, hidden, password, ...)
- **other important attributes**
 - id and/or name
 - value
 - set the initial text, check, ... in the element
 - changes get processed when form submitted
 - javascript calls
 - onfocus="someScript();"
 - onclick="someScript();"
 - onsubmit="someScript();"
 - etc.

Additional Form Features

CMT111-01/-M1—6May2011—54

- **Selection lists, option lists, or drop-down menus**
 - wrap `<select>` tag set around a number of `<option>` tag sets
- **textarea fields**
 - input multiple lines of information, as in an email message
- **Organize groups of form objects by wrapping in `<fieldset>` tags**
- **Form-object labels**
 - provide extra information
 - larger click target
- **Add behaviors (snippets of javascript) to form objects**
 - form validation
 - change web page content/presentation
- **Most forms are processed via server-side scripts after clicking *submit* button**

Four (or Five) Ways to Put Audio and Video on a Web Page

CMT111-01/-M1—6May2011—55

1. Provide a link to the media file

1. in most browsers, clicking on the link will cause the file to play
2. some browsers just offer you an option to download the file

2. embed media file in <object> tag set

1. this the official method

3. embed media file inside <embed> tag set

1. everyone uses this even though it's not official (deprecated actually)

4. embed media file inside <audio> or <video> tag set

1. The future, but today <audio> only works in a few browsers and only for a few formats
 1. Chrome, Safari and Opera—some, but not all formats
 2. Firefox—no formats worked for me; FF is alleged to support <audio>
 3. allegedly will work in IE9, but not current/earlier versions
2. I couldn't get <video> to work in anything...yet

5. Can also embed a flash audio player you or someone else made

Audio 101

CMT111-01/-M1—6May2011—56

- **Only two or three audio formats worth one's time**
- **.mp3**
 - The standard form for music files
 - this is what's on your iPod and what you get from iTunes
- **.midi**
 - created by music synthesizers
 - limited, but easy to create from sheet music
 - ...or from which to create sheet music
- **.wav**
 - basically an unprocessed (or little processed) sound file
 - maintains (most) of fidelity when copied from vinyl records
 - My son and I convert old tapes and records to .wav, and from there to .mp3

Video

CMT111-01/-M1—6May2011—57

- **As with audio, there is a plethora of file formats, most of which can be ignored**
 - stick with .mpeg, .mov, and flash-based video
- **Same options for placing on your web page**
 - plain link and let them figure it out
 - `<object>`
 - `<embed>`
 - `<video>`
 - new tag with HTML 5
 - Someone's flash player

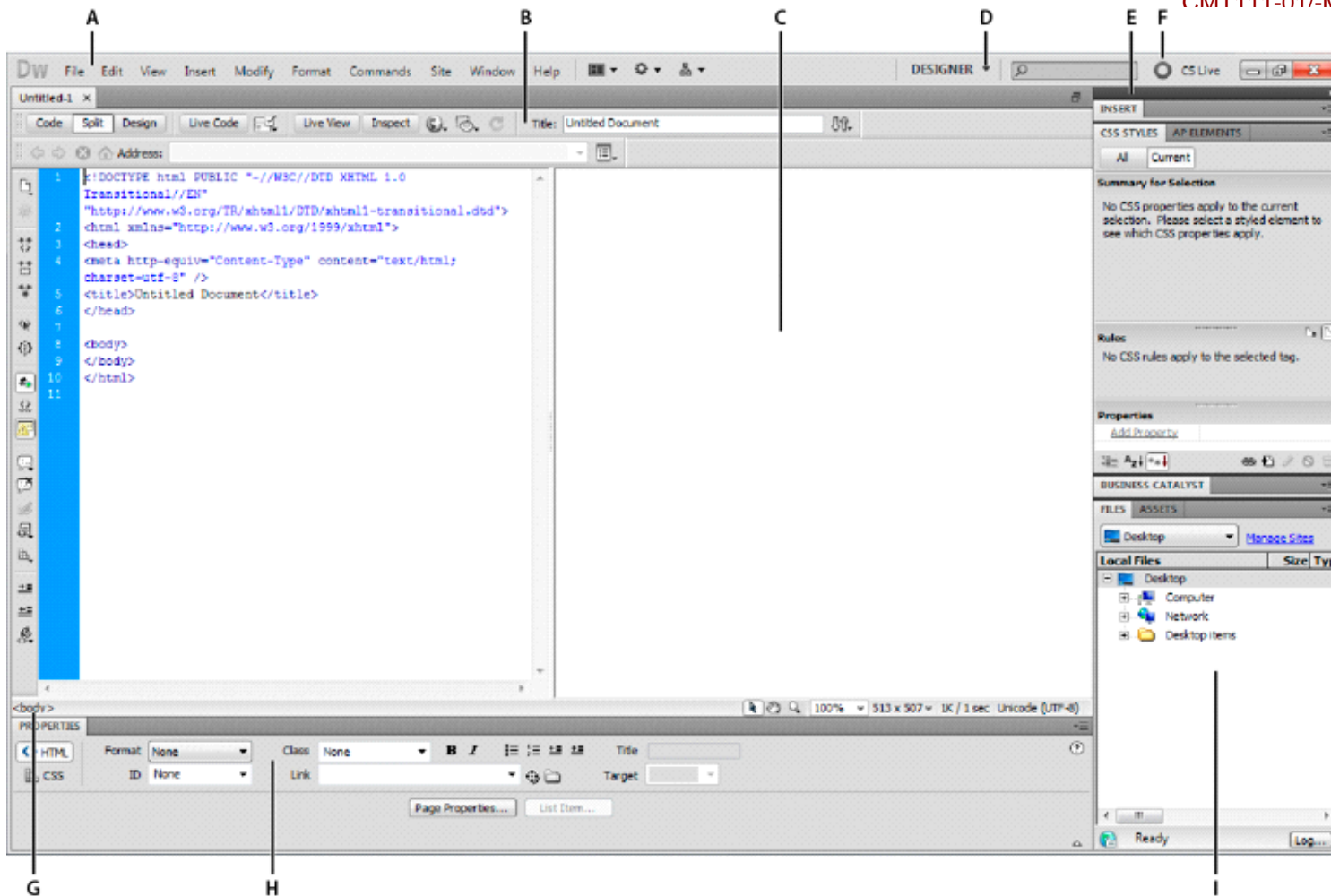
Browser Plug-ins

CMT111-01/-M1—6May2011—58

- **Most browsers play audio and video by way of “plug-ins”**
 - bits of code added to browser
- **Need different plug-ins to play the different formats**
 - Most plug-ins will play .mp3 and .midi
- **My browsers all seem to have Quicktime and Flash plug-ins installed through no effort on my part**
 - Then things work automagically...most of the time
- **Avoid RealPlayer, it’s intrusive crap**
- **Although most Windoz computers have Media Player, browsers other than IE have a difficult time finding the MediaPlayer plug-in (at least mine do)**
 - best to avoid wma for audio and wmv for video (not reliable)
 - Abel and I had fun with this one at the last minute

The Dreamweaver Workspace

C:MT111-01/-M1—6May2011—59



Key

- A. Application bar
- B. Document toolbar
- C. Document window
- D. Workspace switcher
- E. Panel groups
- F. CS Live
- G. Tag selector
- H. Property inspector
- I. Files panel