HTML, CSS, JavaScript
Encoding Information: There’s more!

- Bits and bytes encode the information, but that’s not all

- **Tags** encode format and some structure in word processors

- **Tags** encode format and some structure in HTML

- Tags are one form of meta-data

- **Meta-data** is information about information
The Web uses http:// protocol

Its asking for a Web page, which usually means a page expressed in hyper-text markup language, or HTML

- Hyper-text refers to text containing LINKS that allow you to leave the linear stream of text, see something else, and return to the place you left
- Markup language is a notation to describe how a published document is supposed to look: what kinds of fonts, text color, headings, images, etc.
Basics of HTML #1

- Rule 1: Content is given directly; anything that is not content is given inside of tags
- Rule 2: Tags made of `<` and `>` and used this way:
  
  **Attribute&Value**

  ```html
  <p style="color:red">This is paragraph.</p>
  ```

  It produces: **This is paragraph.**

- Rule 3: Tags must be paired or “self terminated”
There are great resources out there

http://www.w3schools.com/html/default.asp

HTML Tutorial

With HTML you can create your own Web site.
This tutorial teaches you everything about HTML.
HTML is easy to learn - You will enjoy it.

Examples in Each Chapter
This HTML tutorial contains hundreds of HTML examples.
With our online HTML editor, you can edit the HTML, and click on a button to view the result.

Example

```html
<html>
<body>
<h1>My First Heading</h1>
<p>My first paragraph.</p>
</body>
</html>
```

Try it yourself »
Click on the "Try it yourself" button to see how it works

Start learning HTML now!
There are great resources out there

- Let’s do it.
Example

- Write HTML in text editor: notepad++ or TextWrangler
- The file extension is .html; show it in Firefox or your browser

Making a Bake

When do we get to eat it?!
Example: myfirst.html

```
<html>
<head>
<title>Fun in the Kitchen</title>
</head>
<body>
<h1>Making a Bake</h1>
<img src="cooking-ewan-isabel.jpg" alt="Kids Cooking" width="404" height="328" />
<p>When do we get to eat it?! </p>
</body>
</html>
```
Rule 4: An HTML file has this structure:

```html
<html>
  <head><title>Name of Page</title></head>
  <body>
    Actual HTML page description goes here
  </body>
</html>
```

Rule 5: Tags must be properly nested

Rule 6: White space is mostly ignored

Rule 7: Attributes (style="color:red") preceded by space, name not quoted, value quoted
Basics of HTML #3

- To put in an image (.gif, .jpg, .png), use 1 tag
  ```html
  <img src="cooking-ewan-isabel.jpg" alt="Kids Cooking"/>
  ```

- To put in a link, use 2 tags
  ```html
  <a href=http://users.soe.ucsc.edu/~charlie>Charlie’s page</a>
  ```

- More on HTML (including good tutorials) at
  ```html
  http://www.w3schools.com/html/default.asp
  ```
HTML Cheat Sheet:
http://www.simplehtmlguide.com/cheatsheet.php

Basic Tags
<html></html> Creates an HTML document
<head></head> Sets off the title and other information that isn’t displayed on the web page itself
<body></body> Sets off the visible portion of the document

Body Attributes
<body style="background-color:pink"> Sets the background color, using name or hex value
<body style="color:black"> Sets the text color, using name or hex value

Text Tags
<h1></h1> Creates the largest headline
<h6></h6> Creates the smallest headline
<b></b> Creates bold text
<i></i> Creates italic text
<tt></tt> Creates teletype, or typewriter-style text
<em></em> Emphasizes a word (with italic or bold)
<strong></strong> Emphasizes a word (with italic or bold)

Links
<a href="URL"></a> Creates a hyperlink; anchor between tags
<a href="URL"><img src="URL"> </a> Creates hyperlink with image anchor

Formatting
<p></p> Creates a new paragraph
<p style="text-align:left"></p> Aligns a paragraph to the left (default), right, or center.
<br/> Inserts a line break
<blockquote> Indents text from both sides
<hr /> Inserts a horizontal rule
<hr size="3" /> Sets size (height) of rule
<hr width="80%" /> Sets width of rule, in percentage or absolute value

Lists
<dl> Defines a definition list
<dt> Identifies each definition term
<dd> Identifies each definition
<ol> Creates a numbered list
<ul> Creates a bulleted list
<li> Encloses each list item, and adds a number or symbol depending upon the type of list selected

Images
<img src="name" alt="description"/> Places an image
<img src="name" alt="description" style="float:right"/> Aligns an image: right

Tables
<table> Creates a table
<tr> Sets off each row in a table
<td> Sets off each cell in a row
<th> Sets off the table header (a normal cell with bold, centered text)
<table border="1"> Sets width of border around table cells
<table width="500"> Sets width of table, in pixels
<td colspan="2"> Sets number of columns a cell should span (default=1)
<td rowspan="4"> Sets number of rows a cell should span (default=1)
Which does not apply to HTML?

A. It is the language used for the web browser and the web server to communicate over the Internet.

B. It is the language used to express how a document should be displayed.

C. It is a language that allows for “documents” to be created that are not linear. (A book with chapters is linear – you normally read from start to end in order.)

D. All of A-C apply to HTML.
CSS – separating style from content
Content vs Style

- `<h1>`This is a heading`</h1>`
- `<em>`emphasize this`</em>`
- `<b>`Make this bold face`</b>`
- `<p style="color:red">This is paragraph is red.</p>`
CSS Syntax

A CSS rule has two main parts: a selector, and one or more declarations:

Selector  | Declaration  | Declaration
-----------|--------------|--------------
h1          | color: blue; font-size: 12px; |        

The selector is normally the HTML element you want to style.

Each declaration consists of a property and a value.

The property is the style attribute you want to change. Each property has a value.

From http://www.w3schools.com/css/css_syntax.asp
Adding CSS to your html file

```html
<!DOCTYPE html>
<html>
<head>
<style>
p {color:red;text-align:center;}
body {background-image:url("images/ComputerSlug.gif");}
</style>
</head>
<body>
<p>Hello World!</p>
<p>This paragraph is styled with CSS.</p>
</body>
</html>
```
Using an external CSS

<!DOCTYPE html>
<html><head>
<link rel="stylesheet" type="text/css" href="mystyle.css">
</head>
<body>
<p>Hello World!</p>
<p>This paragraph is styled with CSS.</p>
</body></html>
mystyle.css

p {color:red;text-align:center;}
body {background-image:url("images/ComputerSlug.gif");}
Using inline style annotations

<!DOCTYPE html>
<html><head>
</head>
<body>
<p>Hello World!</p>
<p style="color:red;text-align:center;">This paragraph is styled with CSS.</p>
</body></html>
A. css allows you to separate the specific formatting information from the main body of the document.

B. css allows you to change how many html documents will be displayed by changing just a single file

C. A & B

D. neither A nor B
JavaScript
Puts code right in the web page

- Syntax similar to Java
- Has its own set of predefined functions you need to discover (like Processing’s drawing functions).
- How do we put it in the web page?
<!DOCTYPE html>
<html>
<head>
</head>
<body>
<h1>My Web Page</h1>
<p id="demo">Hello!</p>
<button type="button" onclick="myFunction()">Click Me!</button>
<script>
function myFunction() {
    document.getElementById("demo").innerHTML="Goodbye";
}
</script>
</body>
</html>
<h1>My Web Page</h1>

<p id="demo">Hello!</p>

<button type="button" onclick="myFunction()">Click Me!</button>

<p><strong>Note:</strong> myFunction is stored in an external file called "goodbye.js".</p>

<script src="goodbye.js"></script>
function myFunction()
{
document.getElementById("demo").innerHTML="Goodbye!";
}
Click the button to get a time-based greeting.

<button onclick="myFunction()">Try it</button>

<p id="demo"></p>

<script>
function myFunction() {
  var x="";
  var time=new Date().getHours();
  if (time<20) {
    x="Good day";
  } else {
    x="Good evening";
  }
  document.getElementById("demo").innerHTML=x;
}
</script>
What are you supposed to learn?

- HTML lets you programmatically indicate how a particular content should be displayed.
- It can be served up by any HTTP server anywhere in the world.
- CSS lets you partially separate content from presentation.
- JavaScript puts full power of computing in a web page.