

# 03 - HTML & CSS

LBSCI 700 | Spring 2019

Queens College, CUNY

03-css.pdf

HTML

CSS

Lab Skills

Add Some Interactions

Images

FTP

More Exercises

Last Things

# HTML

# What is HTML?

- Standard markup language for creating Web pages
- Hypertext Markup Language
  - Describes the structure of Web pages
  - HTML Elements & Tags
- Browsers vs. HTML Tags

# HTML Elements

- The building blocks of HTML pages
- Represented by tags
  - usually consists of a start tag and end tag, with the content inserted in between

# HTML Tags

- Label pieces of content "heading", "paragraph", "table", and so on
- Two-sided tags  
e.g., `<i>Everything</i>` is on sale
- One-sided tags & attributes  
e.g., ``
- Hyperlinks  
e.g., `<a href="index.htm">Click Here</a>`

Browsers do not display the HTML tags, but use them to render the content of the page

# Mandatory HTML Elements

- DOCTYPE declaration
- html, head, and body elements

Template:

```
<!DOCTYPE Doctype goes here>
```

```
<html xmlns="http://www.w3.org/1999/xhtml">
```

```
  <head>
```

```
    <title>Title that shows in blue title bar of browser</title>
```

```
  </head>
```

```
  <body>
```

```
  </body>
```

```
</html>
```

Note: DOCTYPE not XHTML so has no closing tag.



# CSS

# What is CSS ?

- **Cascading Style Sheets**
- Existed since the 1970s
  - Created as a means of creating consistency
- Styles define **how to display** HTML elements
  - Moves presentation styles away from the html markup

# Advantages to Using CSS

- Less repetitive formatting
- Smaller and mobile files
- Easier maintenance!
- Better formatting control

# CSS Syntax

- The .CSS syntax is made up of three parts: a selector, a property and a value:
- `selector {property: value}`
- The selector is the HTML tag you wish to define
- The property is the attribute you wish to change, and each property can take a value.
  - The property and value are separated by a colon “:” and surrounded by curly braces “{}”

# Selectors

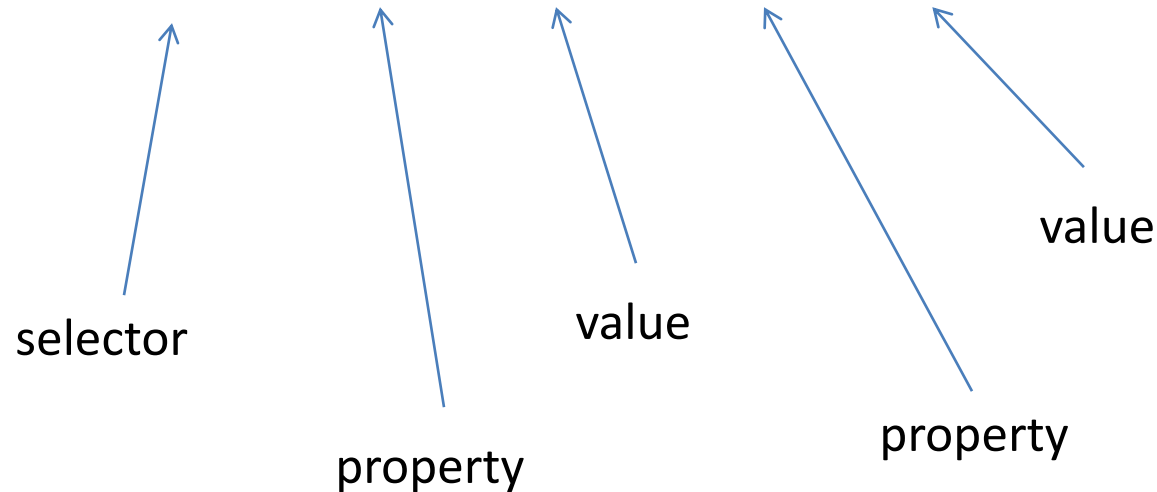
- The selector name creates a direct relationship with the HTML tag you want to edit. If you wanted to change the way a **paragraph tag** behaved, the CSS code would look like
- **p** { PROPERTY: VALUE }

# Selector Examples

- Want more than one property?

“.”  
;

– p {text-align: center; color: red}



# Selector Examples (cont.)

- Vertical Shift
  - In most external .css you will find that each property will be on its own line

```
p
{
text-align: center;
color: black;
font-family: arial
}
```

# Two Ways to Represent CSS

- Internal
  - Contained within the actual html code
- External
  - Placed in a separate file, that only contains CSS code, saved with a ".css" file extension.



# Internal CSS

```
<html>
<head>
  <style type="text/css">
</style>
</head>
<body>
  <p>Your page's
    content!</p>
</body>
</html>
```

- Code in **Red** does not do anything visually
- Place .CSS code between these tags

# Your Turn #1

```
<html>
<head>
<title>Success Recipe</title>
<style type="text/css">
p {color: red; }
body {background-color: green; }
</style>
</head>
<body>
<p>Prepare, study, do labs, read, have fun.</p>
</body>
</html>
```

# Walkthrough

- Select the HTML element we wanted to manipulate.
  - **p{ ; ; }**
- Select the CSS attribute color.
  - p { **color: ;** }
- Select the font color to be red.
  - p { color: **red;** }

\*Now all text within a paragraph tag will show up as red! Now an explanation of the CSS code that altered the <body>'s background:

- Select the HTML element Body
  - **body { ; ; }**
- Select the CSS attribute.
  - body { **background-color: ;** }
- Select the background color to be green.
  - body { background-color: **green;** }

# External CSS file

- Typically its better to keep your .CSS sheet in a separate place from your HTML
- External CSS is a file that contains only CSS code
- It is saved with a “.css” file extension
- Your CSS file is then linked to your HTML page using the **<link>** instead of **<style>**

# External CSS file (cont.)

```
body {  
    background-color: yellow;  
}  
h1 {  
    color: red;  
}  
p {  
    color: red;  
}
```

## Your Turn #2

- Open your text editor, type the following CSS code.

```
body{ background-color: grey;}
```

```
h1{ color: red; }
```

```
p { color: blue; }
```

# Your Turn #3

- Now save file with a .css extension
- Now open a .html file within text editor
- You will add the following code to your header tag

```
<html>
<head>
<link rel="stylesheet" type="text/css" href="default.css" >
</head>
<body>
<h1> Test </h1>
<p>This is a normal paragraph.</p>
</body>
</html>
```

# Your Turn #4

- Note: Your .css file and html file **MUST** be in the same directory if not specifying additional subdir in path.
  - In other words, save them both to your desktop



# CSS Elements

- Background (colors, images, position,
- Font (color, font family, size, style, weight)
- Text (decoration, indent, align, spacing)
- Padding (space between an element's border and the content within it)
- Margin (border, direction)
- Border (styles, width, color, direction)
- List (list style, w/ images, position)

[More Information](#)

# Lab Skills

- Open Text Editor
- See there is HTML code and the Basic Tags

HTML <HTML>

HEAD <HEAD>

Title <TITLE>

Body <BODY>

# Software Installation

- Check that you have or get:
  - Text Editor
  - Multiple web browsers
  - FTP Software

# Saving the HTML File

- Create appropriate directory for course
- Create appropriate directory for website
- Save as
- File type = .htm or .html
- Save

# Viewing the file

- Open Browser
- Open file (browse to file)

# Entering Content

```
<html>  
  <head>  
    <title>Success Recipe</title>  
  </head>  
  <body>  
    Recipe for Success
```

Prepare, study, do labs, read, have fun.

These are the ingredients for success in LBSCI 700.

```
</body>  
</html>
```

# Debugging Errors

1. Open in browser
  2. Review
  3. View Source
  4. Open file in editor
  5. Correct errors \*
  6. Save
- (Repeat by Starting at #1)

\*It is at Step 5 where knowing how to hand tag helps you correct errors in web pages. Many WYSIWYG editors add extemporaneous code that results in errors when viewed through a browser. Once you know how to hand tag you should be able to correct most errors with careful review of code.



# Create and edit a Webpage - HTML tags (cont.)

HN(H1 – H6) `<h1>` `</h1>`

P `<p>` `</p>`

UL `<ul>` `</ul>`

LI `<li>` `</li>`

OL `<ol>` `</ol>`

# Body Tags

```
<html>
  <head>
    <title>Success Recipe</title>
  </head>
  <body>
    <h1>Recipe for Success</h1>
    <p>Prepare, study, do labs, read, have fun.</p>
    <p>These are the ingredients for success in LBSCI
    700.</p>

  </body>
</html>
```

# Self-closing tags & background color

## Self-closing tags

- Line break `<br>`
- horizontal rule `<hr>`

`<hr>`

`<hr size=5>`

`<P> This page last updated on January 30,  
2010.</P>`

## Background color

- `<body bgcolor=#FFFF10>`

For color codes see <http://html-color-codes.info/>

# Simple design element - Horizontal Rule

- The horizontal rule allows a developer to divide information on a webpage. Frequently page developers will place a horizontal rule to break up main page content and copyright, date created info, headers/title, document footer, and/or page developer information.

# Horizontal Rule Example

```
<html>
<head>
<title>Success Recipe</title>
</head>
<body bgcolor="#FFFF10">
<h1><u>Recipe for Success</u></h1>
```

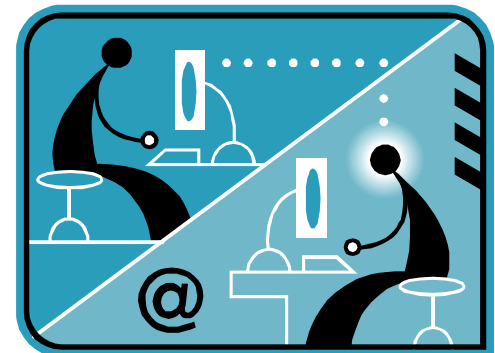
```
<p align="center"> <b>Prepare</b>, <b>study</b>, <b>do labs</b>, read,
  have fun. These are the ingredients for success in LBSCI 700.</p>
```

```
<hr>
<hr size=5>
<P> This page last updated on January 30, 2017.</P>
</body>
</html>
```

# Adding Some Interactions

# Adding some interactions

One of the beauties of the Internet is its ability to allow us to communicate with others. One way to open communication is to provide an e-mail address.



# The “mail to” command

The mail to tag line should fall within the body section of the html document. The command line will look like this:

**`<a href="mailto:qxu@qc.cuny.edu">Email me at qxu at qc dot cuny dot edu.</a>`**



# Link to another webpage

```
<html>
<head>
  <title>Success Recipe</title>
</head>
<body bgcolor=yellow>
<h1>Recipe for Success</h1>
<h3>In order of importance</h3>
<p>Prepare, study, do labs, read, have fun.</p>
  <ol type="I">
    <li>Prepare</li>
    <li>study</li>
    <li>do labs</li>
    <li>read</li>
    <li>have fun</li>
  </ol>
<p>These are the ingredients for success in LBSCI 700.</p>
  <ul>
    <li>Prepare, study, do labs, read, have fun</li>
  </ul>
<p>To see my university <a href="http://www.ua.edu">
click here</a></p>
<br>
<br>
<hr size=5>
<P> This page last updated on February, 2019.</P>
<a href="mailto:qxu@qc.cuny.edu">Email me at qxu at qc dot cuny dot edu.</a>
</body>
</html>
```

Absolute Path

# Providing a download link

Often times providing information via a webpage view is not preferred. For instance, you may want a prospective employer to be able to download your resume as a .pdf file or share a Word or PowerPoint document.

# Providing the file for download

You will need to create the file and save it.  
This file will need to be transferred (by ftp)  
to a directory on the server.

# Providing the link (within same directory)

To provide the link to the .pdf document you simply are making a link to the actual document just as you would create a link to another page.

```
<a href="resume.pdf">Download Joan's  
resume</a>
```

The resume file will need to sit in the same directory as the webpage from which it is being linked.

# Link to file in same dir

```
<html>
<head>
<title>Success Recipe</title>
</head>
<body bgcolor=yellow>
<h1>Recipe for Success</h1>
```

```
<p>To see my work so far see <a href="work.docx">click here</a></p>
```

```
<a href="mailto:qxu@qc.cuny.edu">Email me at qxu at qc dot cuny dot
  edu.</a>
```

```
</body>
</html>
```

**\*\*Understand the difference between absolute and relative paths**

# Providing the link (in a different dir)

```
<a href="resume.pdf">Download Joan's  
resume</a>
```

```
<a  
  href="professional/resume.pdf">Download  
Joan's resume</a>
```

The .pdf file is sitting in a subdirectory (on the same machine / computer) called “professional.” The file will need to be ftp’ d to the “professional” directory if using this code. The path must match exactly. Be cautious of extra “/” or the use of “..”

# Link to file in a sub-directory

```
<html>
<head>
<title>Success Recipe</title>
</head>
<body bgcolor="#FFFF10">
<h1>Recipe for Success</h1>
```

```
<p>To see my work so far see <a href="professional/work.docx">click
  here</a></p>
```

```
<a href="mailto:qxu@qc.cuny.edu">Email me at qxu at qc dot cuny dot
  edu.</a>
```

```
</body>
</html>
```

Be cautious of extra "/"

# Images



# Images

- `<img>` defines an image in an HTML page.
- Requires attributes `src` and `alt`.
- **Note:** The actual image file is not inserted into an HTML page. A link is made to the file from the HTML page. The `<img>` tag creates a holding space for the referenced image.
- **Tip:** To link an image to another document, simply nest the `<img>` tag inside `<a>` tags

# HTML Upgrade

- Layout attributes: align, border, hspace, and vspace were deprecated after HTML 4.01.  
That means they are not supported in HTML5
- HTML 5 supported attributes:
  - src
  - alt
  - height
  - width

# HTML5 <img> And src Attribute

Required. Specifies the URL of an image

Sample code line:

If image is stored in same directory as html file, then the path to the image file would be:

```

```

If image is stored in a subfolder named "images", then the path to the image file would be:

```

```

# HTML5 <img> alt Attribute

Required. Specifies an alternate text for an image

Sample code line:

```

```

```

```

# HTML5 <img> height and width Attributes

Not required. Specifies the height and width of the image (in pixels)

Can be used together or individually

Can be problematic ...

Sample code line:

```

```

# Image file types

3 Most common for use in web pages:

JPEG uses **lossy** JPG compression. Use for higher quality and larger files, or lower quality and smaller files. 24 bit color only.

GIF is a **lossless** (Internet original) indexed file (256 colors maximum), Best for web graphics but not photos. 8 bit indexed only.

PNG is lossless and best for photos. 24 or 48 bit.

# Interpreting image data for file size

A.

Image size =  $3000 \times 2000$  pixels, then this = 6 million pixels (6 megapixels).

B.

6 megapixel image data that is RGB color and 24 bits, then the size of this image data is 6 million  $\times$  3 bytes RGB = 18 million bytes.

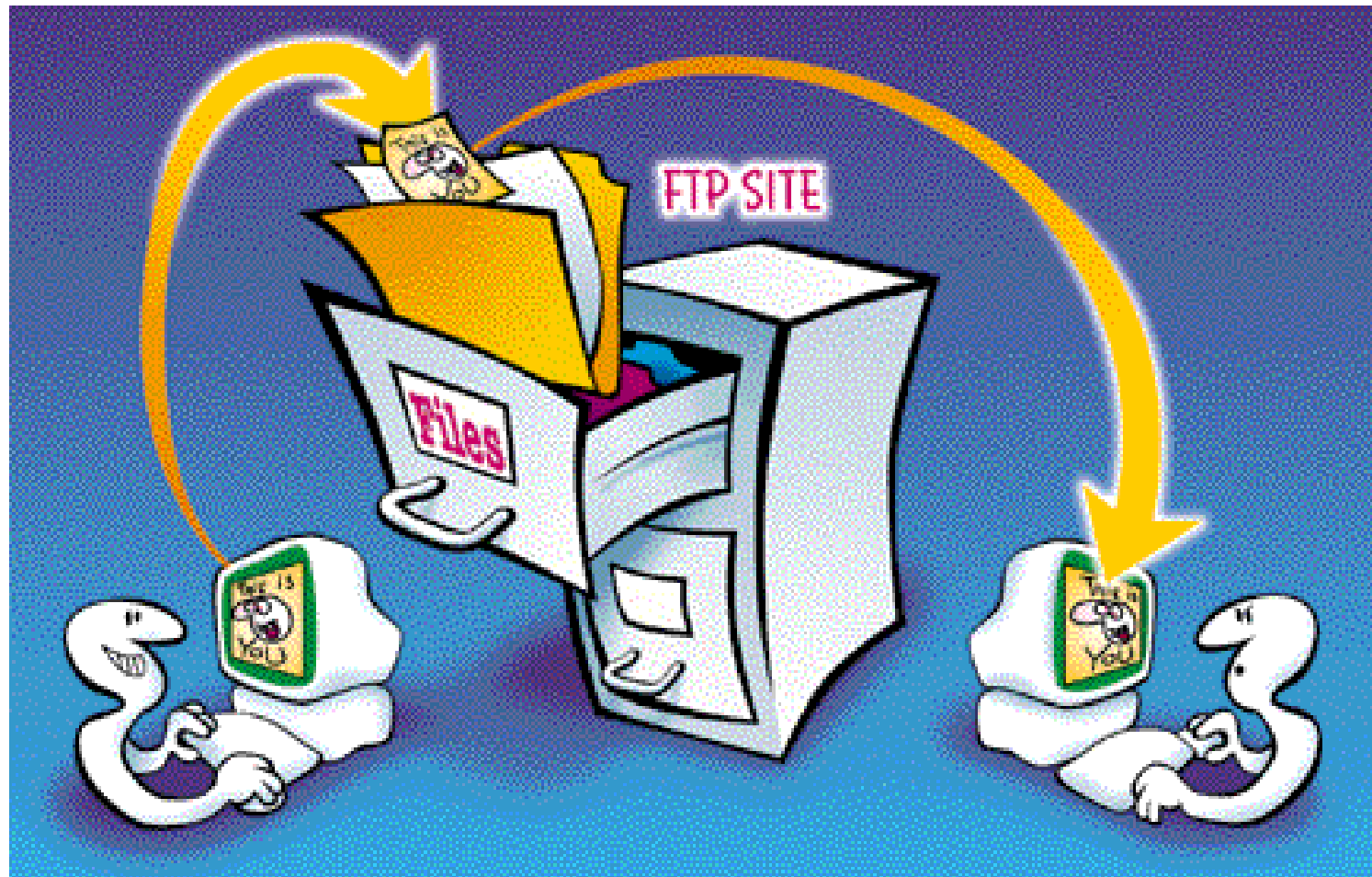
C.

File compression – use JPG to make the file smaller (note some image quality loss)

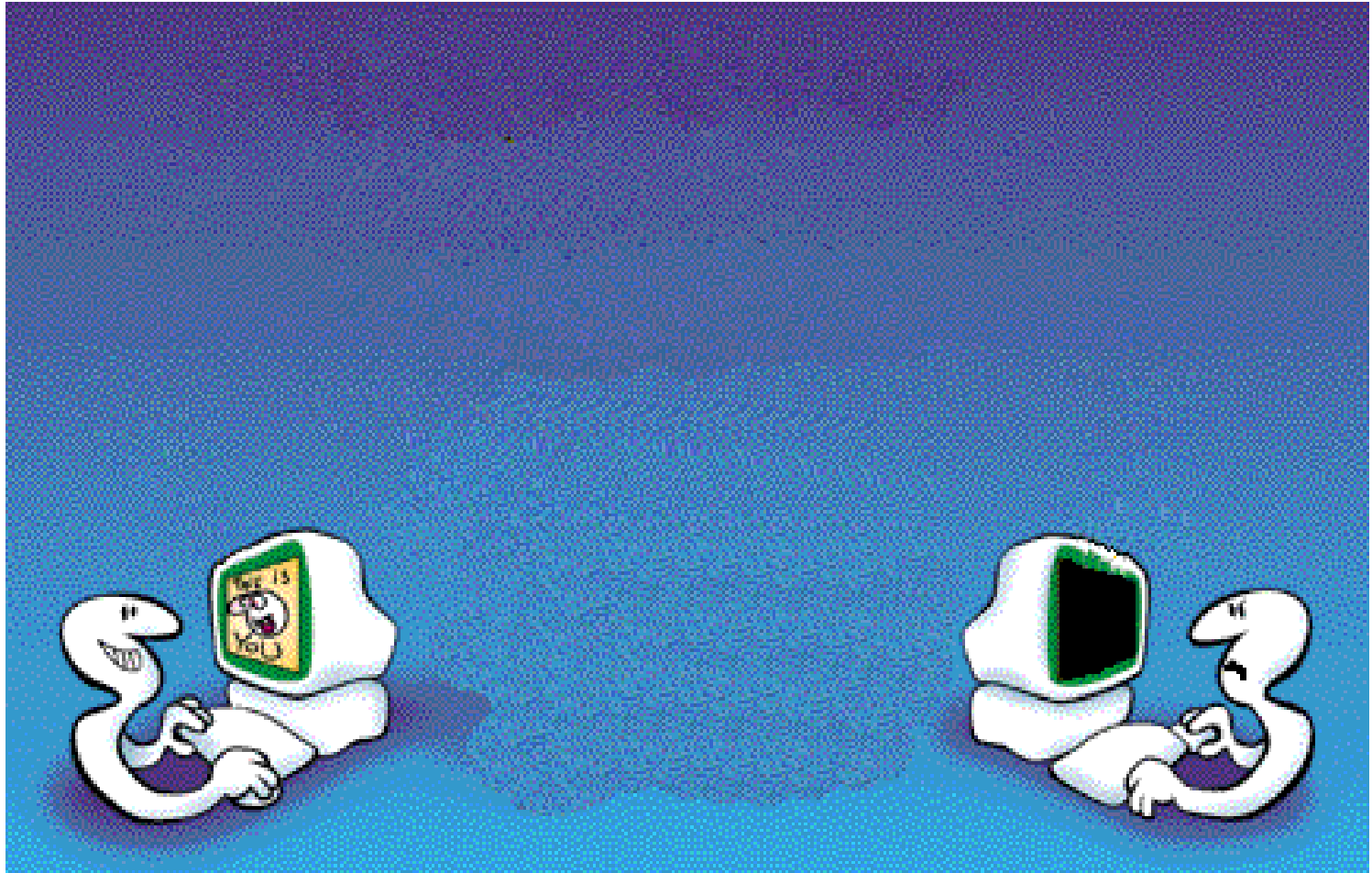
# FTP



# FTP = File Transfer Protocol



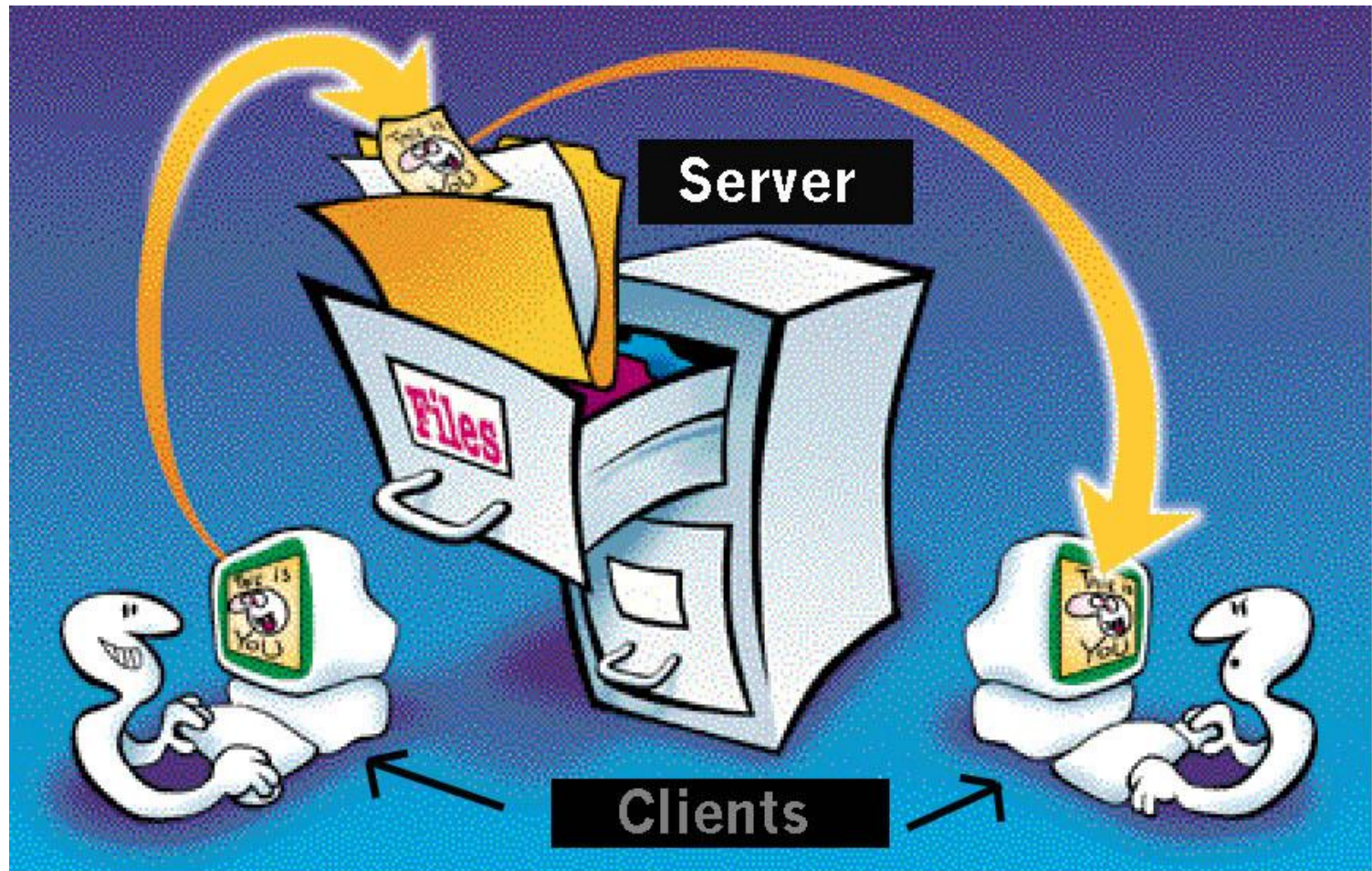
# Without FTP



# Do I use FTP already?

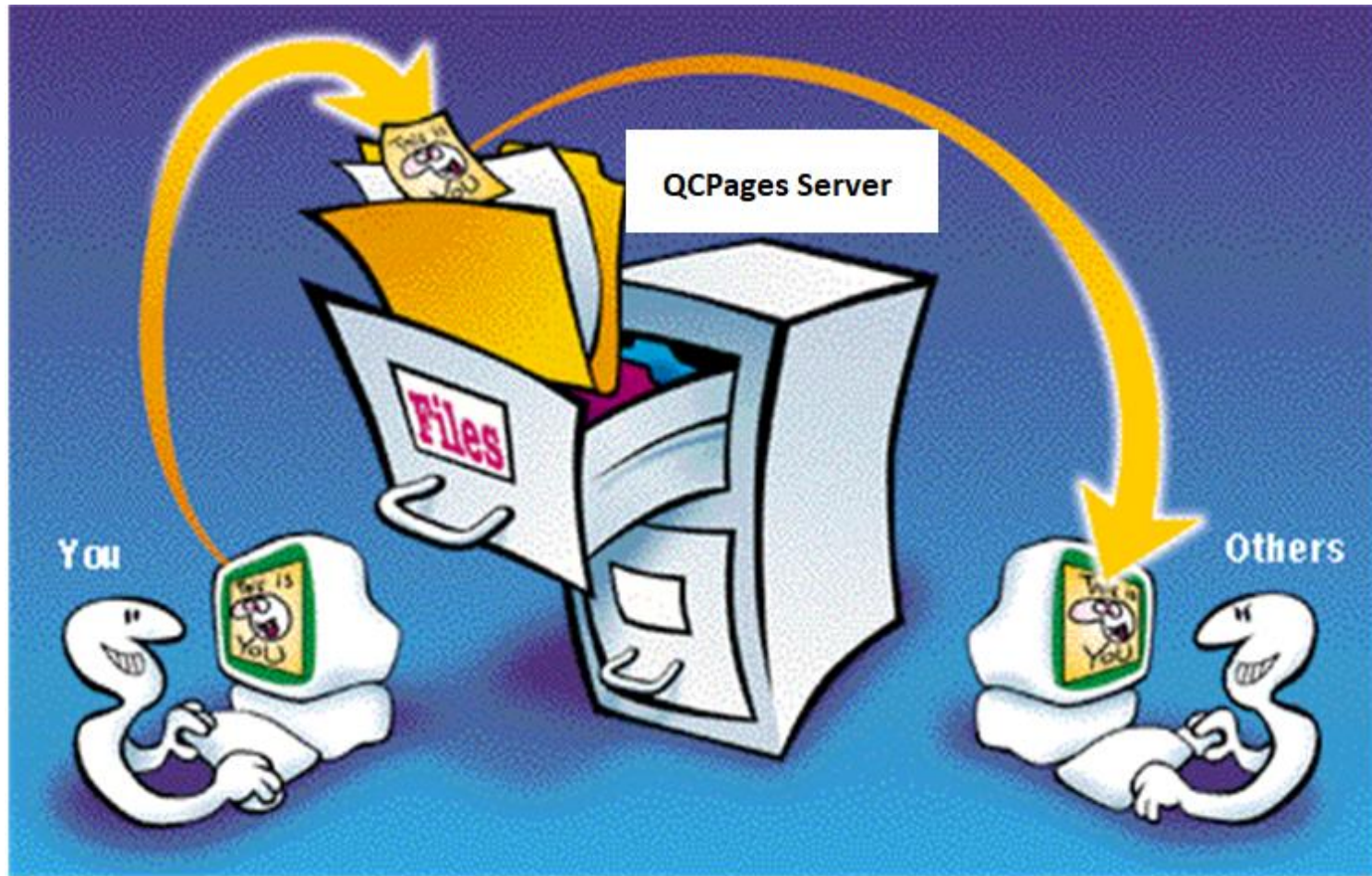
- Attach files to emails
- Add pictures to Facebook albums
- Upload videos to YouTube
- Submit tax forms online
- Any time you “send” something from your computer or device so that someone else using another computer or device can see it

# Client/Server

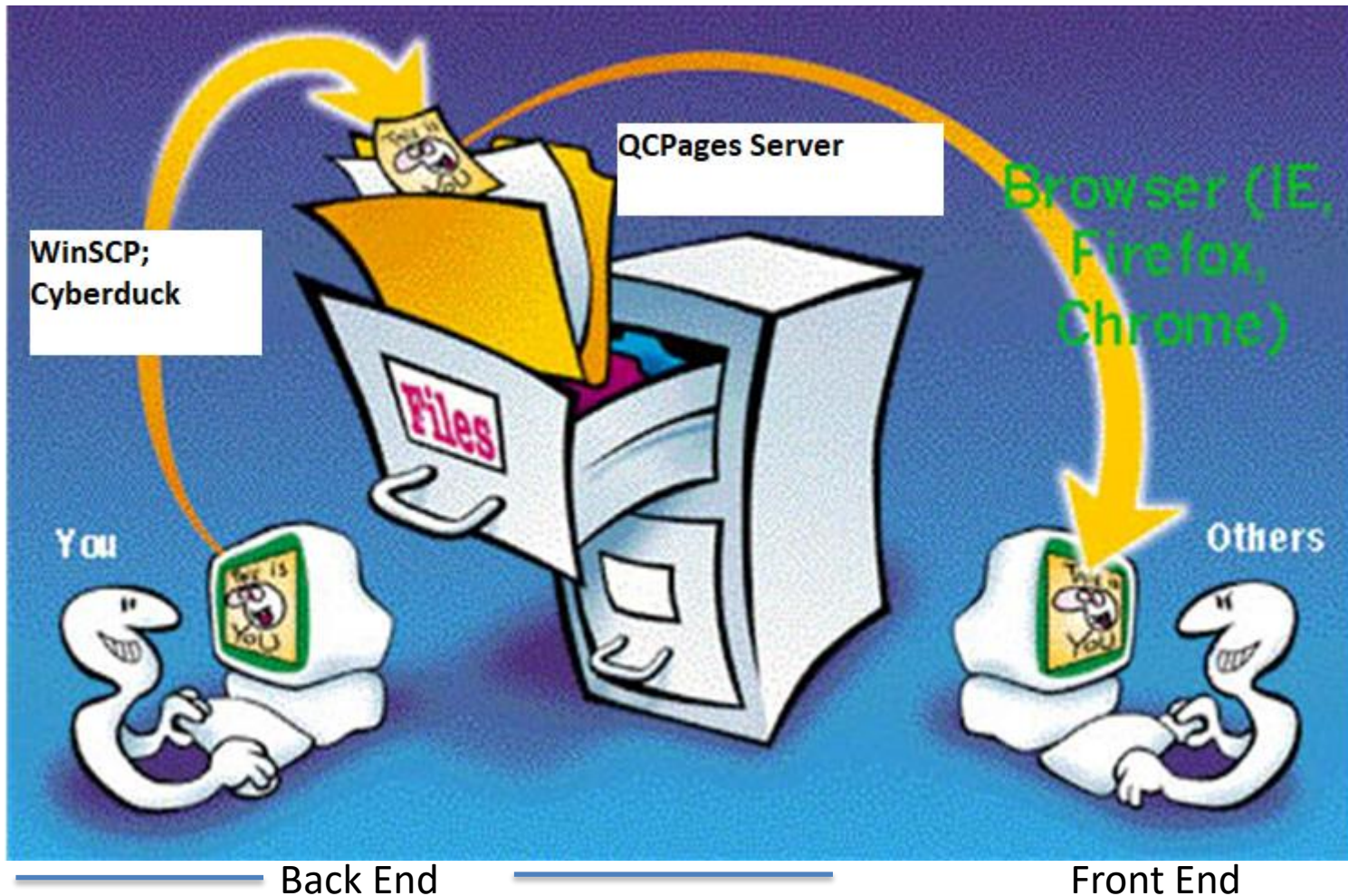




# QCPages Server

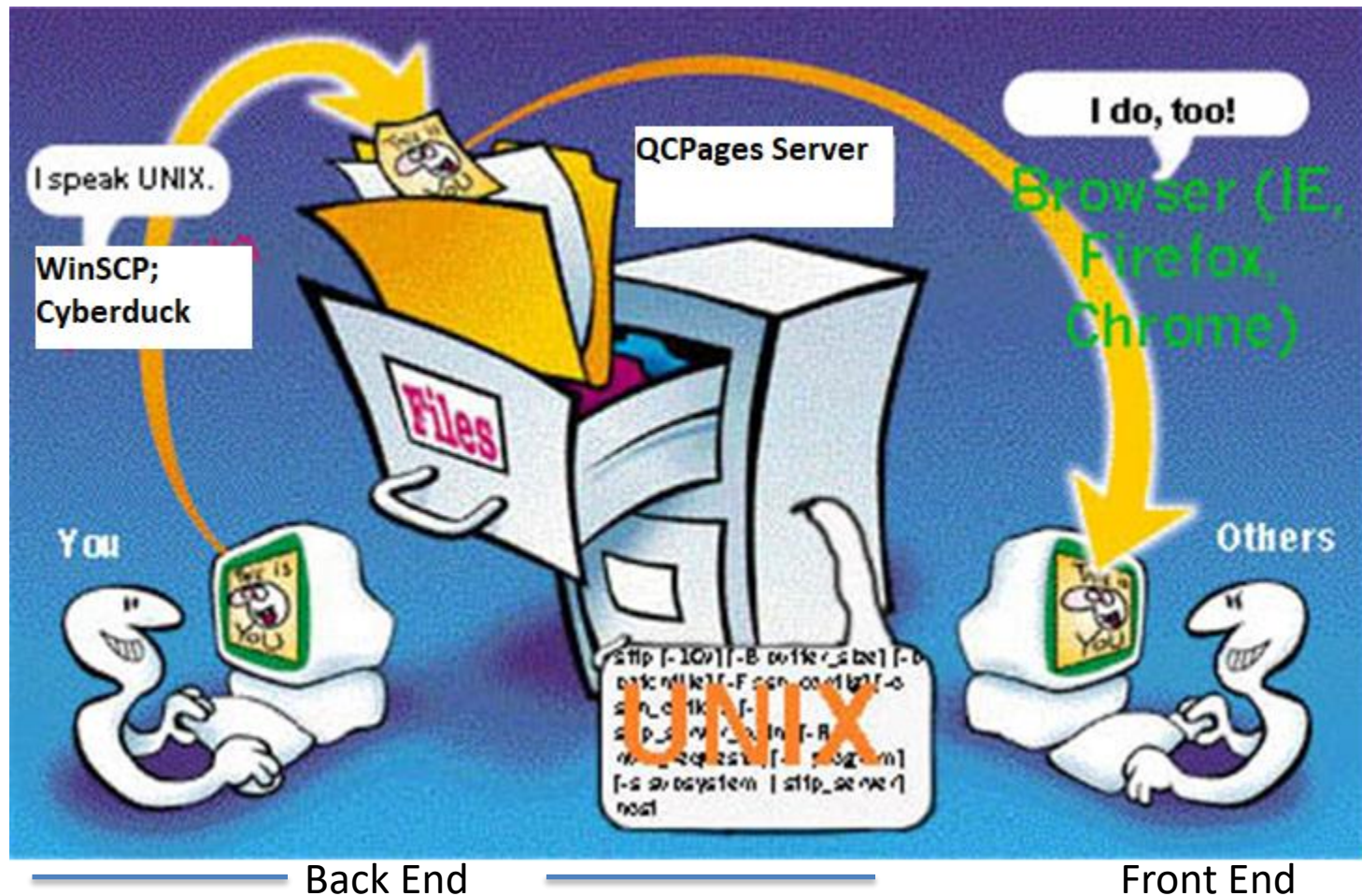


# QCPages Server (cont.)





# QCPages Sever (cont.)



Back End

Front End

# Now you should be able to:

- Understand the purpose of a server and your relationship to it as a client.
- Log on to and off of the `qcpages.qc.cuny.edu` server.
- Navigate to the website files on your computer and to the website folder and files on the QCPages server.
- Copy your website files from your computer to the QCPages server.
- Delete files from the QCPages server and replace them with updated files from your computer when you update your website.



# How to see your webpages

- Place web sites in the public\_html subdirectory of home directory, That is your web site document directory is /home/QCUsername/ public\_html
- Your website will be:  
<https://qcpages.qc.cuny.edu/~QCUsername>.
- If a user does not have an index.html file in /home/QCUsername/ public\_html, a 403 error will be returned.

# More Exercises on HTML

# Ordered Lists

```
<h3>This list is numbered</h3>
```

```
<ol type="1">
```

```
  <li>Read Book</li>
```

```
  <li>Do lab</li>
```

```
</ol>
```

# Your Turn #5: Ordered Lists exercise

```
<html>
<head>
<title>Success Recipe</title>
</head>
<body bgcolor="#FFFF10">
<h1><u>Recipe for Success</u></h1>

<h3>In order of importance</h3>
<ol type="I">
  <li>Prepare</li>
  <li>study</li>
  <li>do labs</li>
  <li>read</li>
  <li>have fun</li>
</ol>
<p>These are the ingredients for success in LBSCI 700.</p>

<hr size=5>
<P> This page last updated on January 30, 2010.</P>
</body>
</html>
```

# Unordered Lists

```
<h3>This list is unordered</h3>  
<ul type="square">  
  <li>Read Book</li>  
  <li>Do Lab</li>  
</ul>
```

# Your Turn #5: Unordered Lists exercise

```
<html>
<head>
<title>Success Recipe</title>
</head>
<body bgcolor="#FFFF10">
<h1><u>Recipe for Success</u></h1>

<h3>In no particular order</h3>
<ul type="square">
  <li>Prepare</li>
  <li>study</li>
  <li>do labs</li>
  <li>read</li>
  <li>have fun</li>
</ul>
<p>These are the ingredients for success in LBSCI 700.</p>

<hr size=5>
<P> This page last updated on January 30, 2010.</P>
</body>
</html>
```

# Nested Lists

- `<h4>A Nested List:</h4>`
- `<ul>`
- `<li>Coffee</li>`
- `<li>Tea`
- `<ul>`
- `<li>Black tea</li>`
- `<li>Green tea</li>`
- `</ul>`
- `</li>`
- `<li>Milk</li>`
- `</ul>`

# Nested Lists - Results

## **A Nested List:**

- Coffee
- Tea
  - Black tea
  - Green tea
- Milk



# Your Turn #6: Nested Lists exercise

```
<html>
  <head>
    <title>My Exercise</title>
  </head>
  <body>
    <h4>A Nested List</h1>
    <ul>
      <li>Coffee</li>
      <li>Tea
        <ul>
          <li>Black Tea</li>
          <li>Green Tea</li>
        </ul>
      </li>
      <li>Milk</li>
    </ul>
  </body>
</html>
```

# Another Nested List - Results

## Another Nested List:

- Coffee
- Tea
  - Black tea
  - Green tea
- Milk
  1. Strawberry milk
  2. Chocolate milk
    - Naturally sweetened
    - Artificially sweetened

# Your Turn #7: Another Nested Lists Exercise

```
<html>
  <head>
    <title>Drink Menu</title>
  </head>
  <body>
    <h4>Another Nested List</h4>
    <ul>
      <li>Coffee</li>
      <li>Tea
        <ul>
          <li>Black Tea</li>
          <li>Green Tea</li>
        </ul>
      </li>
      <li>Milk</li>
      <ol>
        <li>Strawberry Milk</li>
        <li>Chocolate Milk
          <ul>
            <li>Naturally Sweetened</li>
            <li>Artificially Sweetened</li>
          </ul>
        </li>
      </ol>
    </ul>
  </body>
</html>
```

# Tables

- Tags
  - `<table>`
  - `<tr>`
  - `<td>`
- Great way of structuring content on your webpage
- But can get confusing to keep track of rows and columns in code view

# Table Attributes

- `<table>`
  - border
- `<tr>`
- `<td>`
  - colspan
  - rowspan

# More HTML Coding Exercises

<http://www.w3schools.com/>

<http://www.codecademy.com/#!/exercises/>

# Last Things

# ToDo

- Start homework
  - Look for email
  - Check Bb weekly folder
- Note any questions from reading and homework
- Class 4 on Feb 25 (look for email)