MIS3690 WEB TECHNOLOGIES

BABSON COLLEGE

TOIM DIVISION

FORMS
SERVER-SIDE PROCESSING

FORMS

- Why forms?
  - Let visitors interact with your site
  - Change the nature of your site from a presentation or brochure oriented site to an interactive site
  - Collect important information
- Form Functions
  - Input
  - Action
    - Usually taken by server, but can also be done on browser by JavaScript
    - Occurs when user clicks on "Submit" button

FORM EXAMPLE

```
<form method="post" action=mailto:zli@babson.edu>

    Form text and input tags go here

</form>
```

FORM TAG ATTRIBUTES

- `action` – identifies a server (URL) and the program to be run on it.
  - The only required attribute
  - Can also be `mailto:address` (only works if the user has a mail program such as Outlook installed. It will not work with webmail.)
- `method` – describes how form's data is sent to server (see next slide)
- `enctype` – describes how the form input is encoded before being sent

FORM METHODS

POST	GET*
Form data are encoded into a message sent to the server.	Form data is encoded by the browser into a URL.
Better functionality: Can retrieve, store, and update data, modify databases, etc.	Best used for retrieving data.
No restriction on amount or type of data sent	URL max length of 100 and ASCII characters only
More secure	Less secure

* GET is the default method if no method is specified.

FORM INPUT WITH <INPUT> TAG

- Text line: `<input type="text" />`
- Password: `<input type="password" />`
- Radio buttons: `<input type="radio" />`
- Check boxes: `<input type="checkbox" />`
- Files: `<input type="file" />`
- Hidden data: `<input type="hidden" />`
- HTML5 New Input Types:
  - https://www.w3schools.com/html/html_form_input_types.asp

INPUT TAG ATTRIBUTES

- `type` – the type of input
- `name` – names the input tag. Names and the user-entered data are passed as pairs to the server
    - Name is mandatory for each `<input>` tag
- Other attributes depend on the type

INPUT TAG CONTEXT

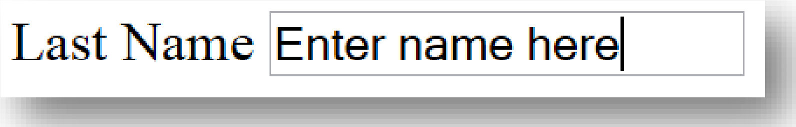
- Input tags must be placed **inside** the <form> tags and if necessary, **inside** the same types of tags that text is placed:
  - <p> - use this to space elements evenly
  - <h#>– use this to format the text
    - where # is one of 1…6
  - <td> - when you need a form inside a table!

INPUT: TYPE = "TEXT"

- Example:

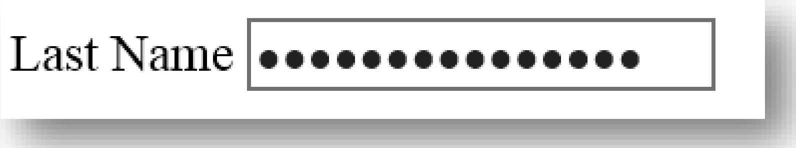
```
<p> Last Name

<input type= "text" name="lname" maxlength="100" size="20"
value="Enter name here"  /> </p>
```



Last Name Enter name here

- `password` type works the same way, except that value appears asterisked out as it is entered
  - `type= "password "`



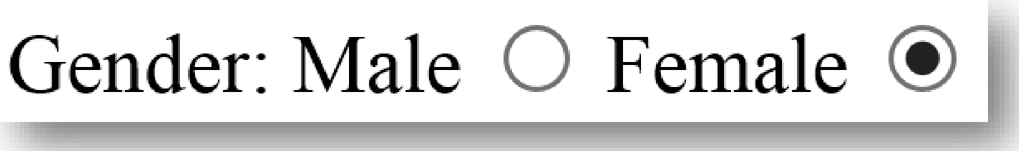
Last Name ●●●●●●●●●●●●●●●

INPUT: TYPE = "RADIO"

Note: Same name implies that only one of these radio buttons can be true at a time.

- Example:

```
<p>Gender:
Male <input type="radio" name="gender" value="male" />
Female <input type="radio" name="gender" value="female"
checked="checked" /></p>
```



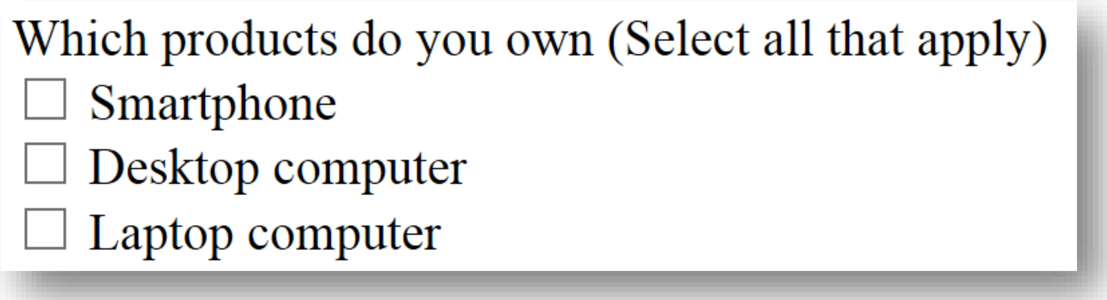
Gender: Male ○ Female ●

Note: optional checked attribute indicates if one of the buttons should be checked initially.

INPUT: TYPE = "CHECKBOX"

- Example:

```
<p>Which products do you own (Select all that apply)<br />
<input type="checkbox" name="Q1" value="Smartphone" />
   Smartphone<br />
<input type="checkbox" name="Q1" value="DeskC" /> Desktop
   computer<br />
<input type="checkbox" name="Q1" value="LapC" /> Laptop
   computer</p>
```



Which products do you own (Select all that apply)

□ Smartphone

□ Desktop computer

□ Laptop computer

- Optional attribute: `checked="checked"`

INPUT: TYPE = "HIDDEN"

- A hidden field is one that is not visible to the user, however it can be seen when viewing the source code.
- Why use hidden fields:
  - Server-side program can collect data from one form to another
  - This makes it easier for the users.
- Example:

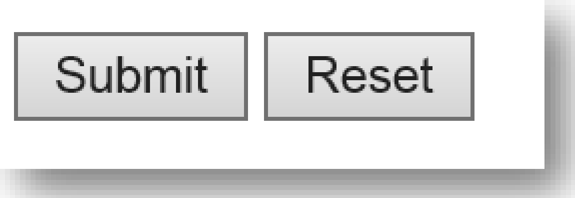
```
<input type="hidden" name="name" value="value"/>
```

BUTTONS WITH INPUT TAG

- Example:

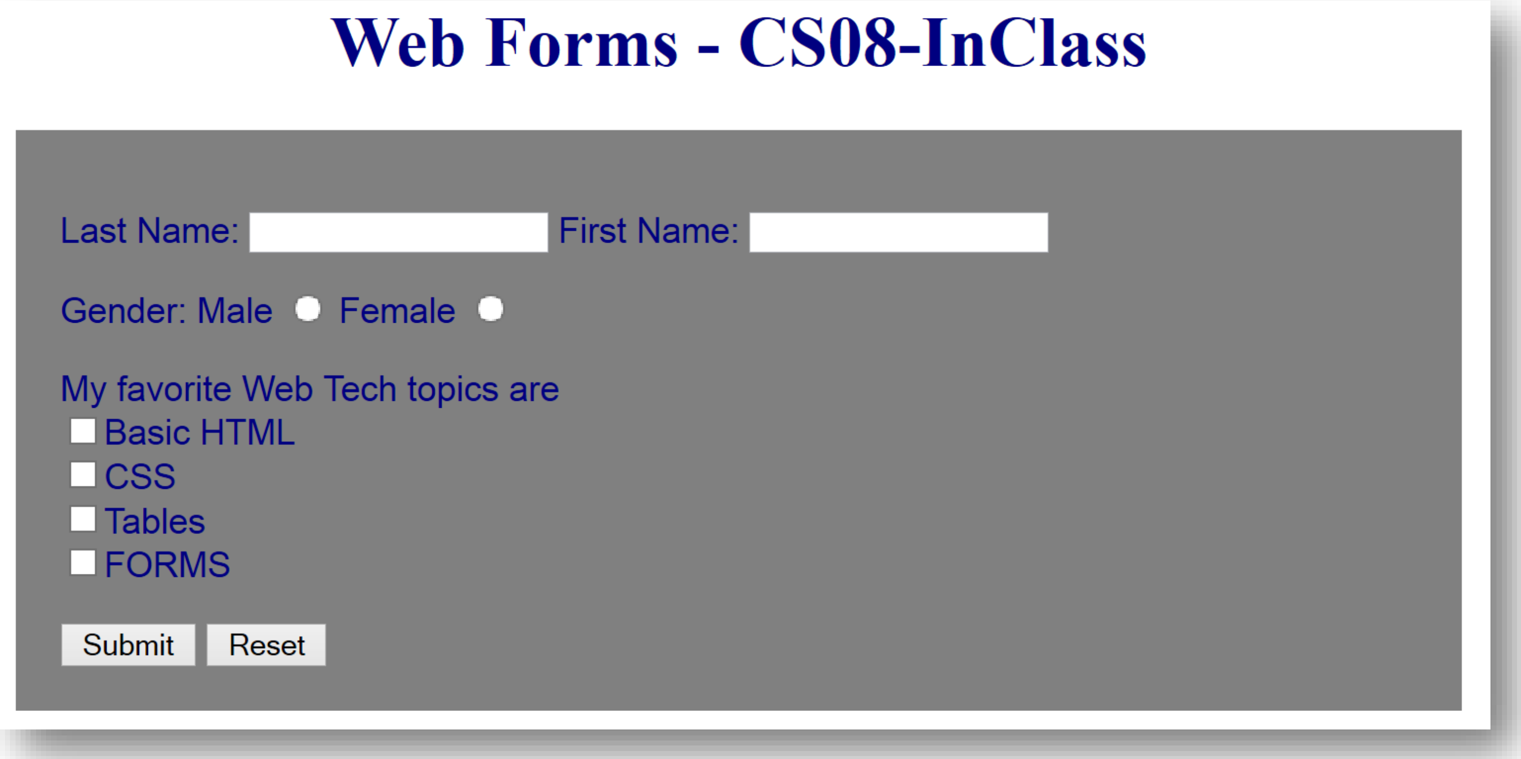
```
<input type="submit" value="Submit" name="Submit"  />
<input type="reset" value="Reset" name="Reset" />
```

- These two are standard buttons. The "type" value should not be changed!



CS08-INCLASS.HTM

- Create a form that looks like what is shown here.



Web Forms - CS08-InClass

Last Name: ________ First Name: ________

Gender: Male ○ Female ○

My favorite Web Tech topics are
□ Basic HTML
□ CSS
□ Tables
□ FORMS

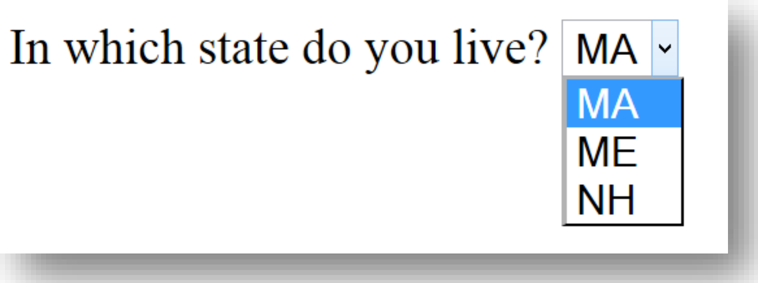
MORE FORM TAGS FOR USER INPUT

- `<select>` – Allow users to select from a menu
- `<textarea>` – Let user enter multi-line text with optional wrapping
- `<button>` – Let users select an action

THE SELECT TAG (WITH OPTION TAGS)

- Example:

```
<p>In which state do you live?
<select name="state">
    <option value="MA">MA</option>
    <option value="ME">ME</option>
    <option value="NH">NH</option>
</select></p>
```



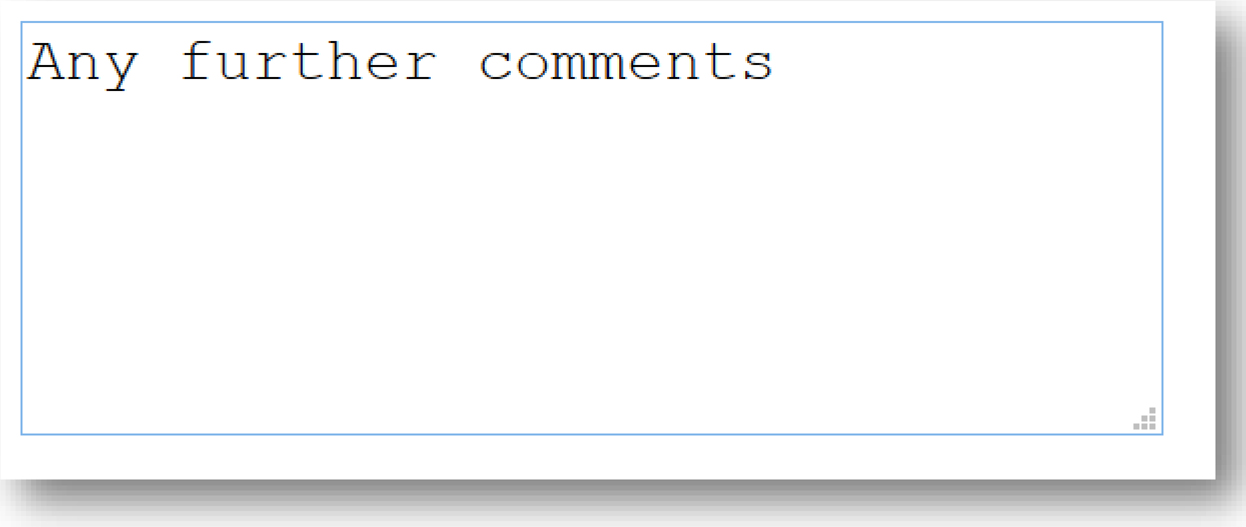
In which state do you live? MA

- Optional attribute for select: `multiple="multiple"`
- Optional attribute for select: `size="#"` – to set number of choices visible
- Optional attribute for option: `selected="selected"`

TEXT AREA

- Example:

```
<textarea name="comments" rows="5" cols="29">Any further
comments</textarea>
```

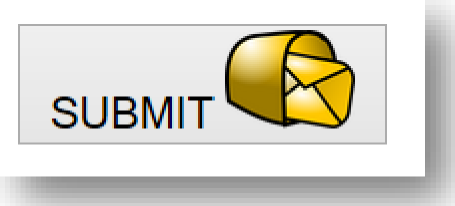


```
Any further comments
```

IMAGE BUTTONS

- Example:

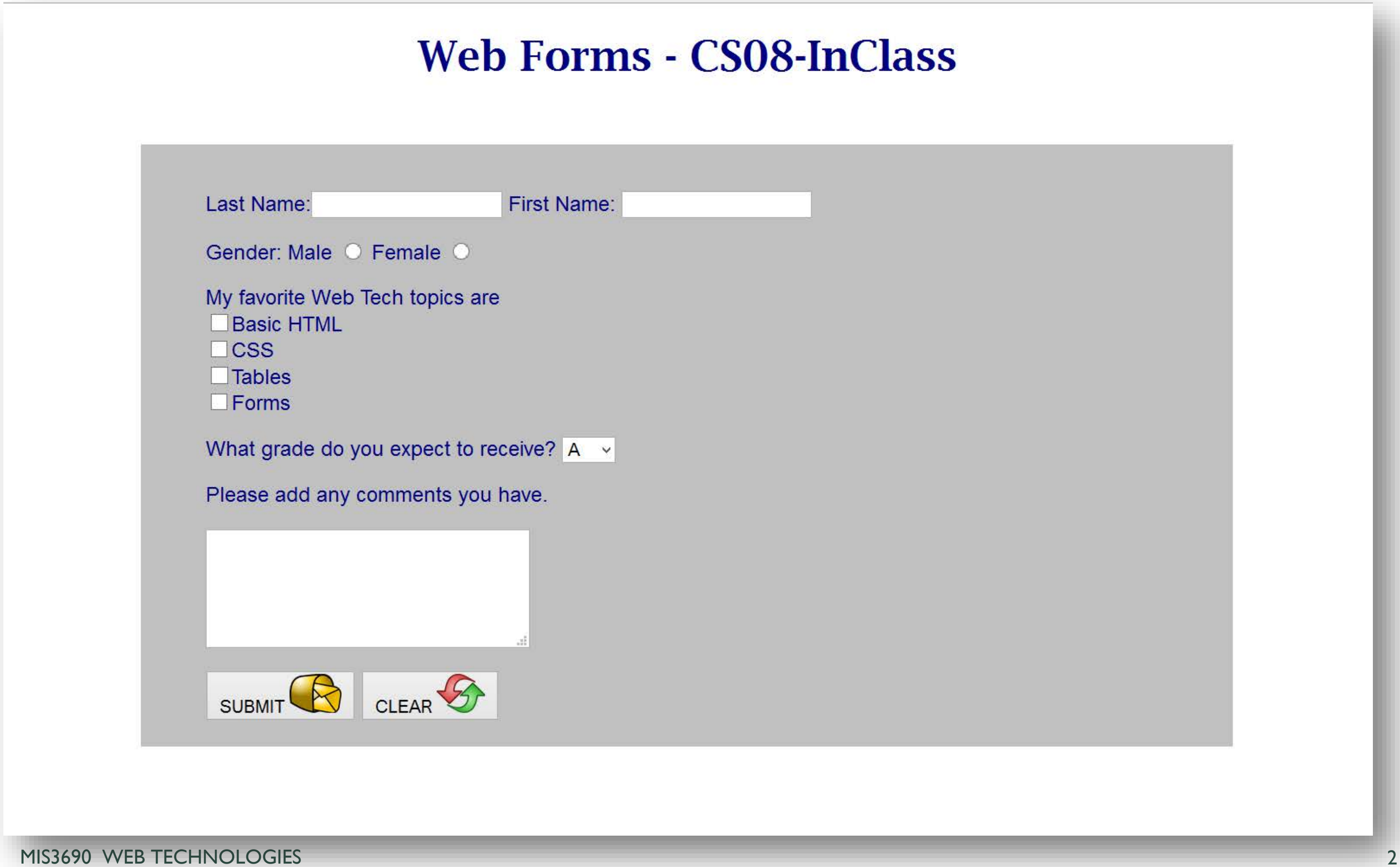
```
<button type="submit" name="submit" value="submit">
   SUBMIT <img src="mailbox.png"  height="30"
   alt="submit_button" />
</button>
```



CS08-INCLASS.HTM (CONT.)

- Add a drop-down menu.
- Add a text area for comments.
- Check it out. Then style it as follows:
  - For the form:
    - Set the margins as 50px top and bottom; 100px left and right; set the padding as 20px top and bottom; 50px right and 100px left;
    - Set the font to Arial, color to Navy
    - Set background to Silver
  - For the heading, center it, use Lucida Bright font in Navy color

CS08-INCLASS.HTM (CONT.)



Web Forms - CS08-InClass

Last Name: First Name:

Gender: Male ○ Female ○

My favorite Web Tech topics are

□ Basic HTML
□ CSS
□ Tables
□ Forms

What grade do you expect to receive? A

Please add any comments you have.

FORM PROCESSING

- Determined by "action" attribute of the form tag
- May be
  - `mailto:address`
    - Works only if user has a mail program, such as Outlook, installed; does not work with webmail
  - Name of program file on server
    - Program processes the data submitted by form

ENCODING TYPE

- Tells browser how to format data when method is post
- Example:

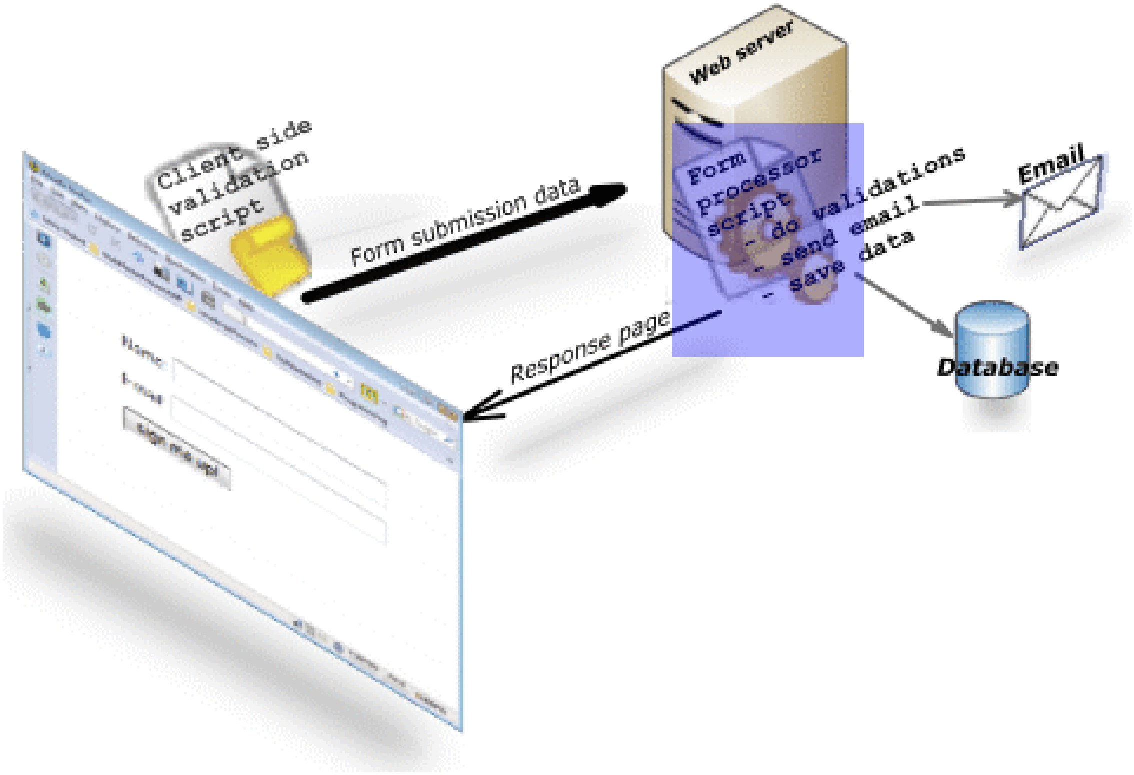
```
enctype="value"
```

- Use for value:
  - text/plain for mailto
  - multipart/form-data for non-ASCII text or large binary data
  - application/x-www-form-urlencoded for most other form applications

SERVER SIDE PROCESSING

- When you submit, a server-side program takes over.
  - Perl or CGI Scripts
  - ASP or DOT-NET Scripts
  - Java Servlets
  - Server-side JavaScript
  - Python program
- Complex form processing requires programming

PROCESSING….



Source: http://javascript-coder.com/html-form/html-form-tutorial-p4.phtml

PROCESSING WITHOUT A BACK-END

- Can use the mailto as part of the form action

```
<form method="post" action=mailto:zli@babson.edu
enctype="text/plain">
```

- Sends the content of the form fields to an email address

USING A BACK-END HOSTING SERVICE

- There are many services on the Web that will help you send and process your data.
- Some are free, others charge.
- Most hosting services send you data via email.
- Using "mailer.php" on your form.

TEST CS08-INCLASS.HTM (OPTIONAL)

- Download the attached mailer.php (right-click and Save As into your folder).
- Use Notepad to modify the fourth line (`"$headers = "From: zli@babson.edu";`), replacing my address with yours.
- Modify the <form> tag of your CS08-InClass.htm file so that the action is "mailer.php" and the enctype attribute is deleted.
- FTP both files to the class FTP server.
  - NOTE: The mailer.php program will not work on your laptop. It works only on a server that has the php language installed.