

CS 2033

Multimedia & Communications II

LECTURE 4 – HTML AND WEB FORMS

1

Announcements

2

- ▶ Quiz 1 will be open next Wed-Thurs (Feb 5-6).
- ▶ Make sure you do the quiz while it is open! It's a 0 if not completed by 11:55pm on Thursday.
- ▶ It is open-book so feel free to refer to the lecture slides.

Announcements

3

- ▶ Assignment 1 is due this Friday (Jan 31).
- ▶ Get it done early and submitted properly on time.
- ▶ Our TAs are holding consulting hours this week. The schedule is posted on OWL. They will likely be held in MC16A.

Websites from scratch!

4

- ▶ No more Kompozer or BlueGriffon!
- ▶ I'll teach you how to create a website using code.
- ▶ What you will need:
 - ▶ Plain text editor like Brackets
 - ▶ Do NOT use Word for this!
 - ▶ Internet browsers...

Which browsers?

5

- ▶ Chrome is the most popular one.
- ▶ Ideally, websites should work on ALL platforms and browsers.



- ▶ Most browsers render code very similar to one another. Sometimes Edge behaves a little differently.

Website languages

6

- ▶ The primary trinity of languages:
 - ▶ HTML5
 - ▶ CSS3
 - ▶ JavaScript
- ▶ Other languages:
 - ▶ PHP
 - ▶ ASP



HTML

7

- ▶ HTML: Hypertext Markup Language.
- ▶ Consists entirely of tags.
- ▶ Tons of elements, some of which we'll go over in detail.
- ▶ Standard language for making static websites.
- ▶ Files usually end in .html or .htm

Files and folders

8

- ▶ The homepage or splash page must be named `index.html` to be loaded first.
- ▶ Other files should be named appropriately and:
 - ▶ All lowercase (fruit.html)
 - ▶ No spaces (myfamily.html)
 - ▶ Not too long (uwo.html)

Files and folders

9

- ▶ Folder structure is also important.
- ▶ Have one `root` folder which will contain all the website files.
- ▶ Within that, have a sub-folder for `images`, and sub-folders for `css`, and `js` (once we get there).
- ▶ HTML files should generally stay in root folder, not in sub-folders.

Filepaths

10

- ▶ Remember the horror of links and images breaking when you uploaded your CS1033 websites?
- ▶ **Absolute filepaths** point directly to a file using a full path to its location.
- ▶ **Relative filepaths** indicate how to get to the file relative to the HTML file containing that link.

Filepaths

11

- ▶ Use relative paths exclusively!
- ▶ This way, they will work on your computer AND on the server because the relative location does not change.
- ▶ A file's full location path changes when you upload it, so do not use absolute paths!

Filepaths

12

- ▶ `C:/Users/tomsmith/Desktop/website/images/apple.jpg`
 - ▶ Works on Tom Smith's laptop
 - ▶ Does NOT work ANYWHERE else!
- ▶ `images/apple.jpg`
 - ▶ Works on Tom Smith's laptop
 - ▶ Works on server too ☺

HTML tags

13

- ▶ Most tags need to be closed, so that content is surrounded by the opening and closing tags.
 - ▶ i.e. `<p>This is a <b>great</b> paragraph.</p>`
- ▶ Some tags are self-closing.
 - ▶ i.e. `<br />`

Attributes

14

- ▶ HTML tags often have attributes which provide extra information about the content or design of the element.
 - ▶ `<img src='picnic.jpg' width='200' />`
 - ▶ `src` and `width` are two attributes in `img`
 - ▶ You can use either single or double quotations but open and close a string (text) with the same type!

Doctype

15

- ▶ Tells the browser which HTML version is being used.
- ▶ This must be the very first line.
- ▶ `<!doctype html>`
- ▶ Most browsers will assume HTML 5 if not specified but it is best to include this line regardless!

Head and Body

16

- ▶ Main two sections:
 - ▶ Head
 - ▶ `<head></head>`
 - ▶ Meta data, setting title, linking external files, internal CSS or JS, etc.
 - ▶ Body
 - ▶ `<body></body>`
 - ▶ The main part of the website.

Shell of a website

17

```
<!doctype html>
<html>
  <head></head>
  <body></body>
</html>
```

Title Property

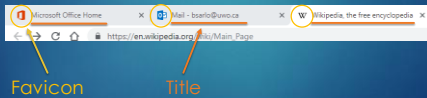
18

- ▶ Remember the title property talked about repeatedly in CS1033?
- ▶ This is the text that shows up in the browser tab.
- ▶ `<title>Website – Webpage</title>`
- ▶ This is **meta** data about the page, so it goes in the **head** section.

Favicon

19

- ▶ Have you noticed the little icons beside the title property in a tab?
- ▶ This is called the favicon
- ▶ Small images, usually 16x16 pixels.



Favicon

20

- ▶ `<link rel="shortcut icon" href="favicon.ico" type="image/x-icon">`
 - ▶ Notice the image is .ico format.
 - ▶ Some browsers allow .png or .gif formats too.
 - ▶ www.favicon.cc
- ▶ This is **meta** data about the page, so it goes in the **head** section.

What else goes in head?

21

- ▶ In addition to the title and favicon, the head often directly contains, or has links to, CSS and JavaScript.
 - ▶ We will talk more about that in coming weeks!
- ▶ Sometimes meta tags are used for SEO purposes. This isn't as popular as it used to be.

Now to the body!

22

- ▶ The **body** is typically much larger than the head.
- ▶ This contains content and layout design elements that are displayed in the actual webpage.
- ▶ Some head data (i.e. CSS or JavaScript) can be done in the body as well.

Common body elements

23

- ▶ `<h1>`Largest header`</h1>`
- ▶ `<h6>`Smallest header`</h6>`
- ▶ `<p>`Paragraph of text`</p>`
- ▶ `<a href="index.html">`Link`</a>`
- ▶ ``
- ▶ `<b>`Bold text`</b>`
- ▶ `<i>`Italicized text`</i>`

Common body elements

24

- ▶ `<ul>` creates an unordered list.
- ▶ `<ol>` creates an ordered list.
- ▶ `<li>` adds a list item to a `<ul>` or `<ol>`

```
<ul>
  <li>Apples</li>
  <li>Grapes</li>
  <li>Peaches</li>
</ul>
```

- Apples
- Grapes
- Peaches

```
<ol>
  <li>Apples</li>
  <li>Grapes</li>
  <li>Peaches</li>
</ol>
```

1. Apples
2. Grapes
3. Peaches

Common body elements

25

- ▶ `<table>` creates a table.
- ▶ `<tr>` creates a row in the table.
- ▶ `<th>` creates a header/title cell.
- ▶ `<td>` creates a data cell.

```
<table border="1">
  <tr>
    <th>Fruit</th>
    <th>Amount</th>
  </tr>
  <tr>
    <td>Apples</td>
    <td>4</td>
  </tr>
  <tr>
    <td>Peaches</td>
    <td>2</td>
  </tr>
</table>
```

Fruit	Amount
Apples	4
Peaches	2

Turning the tables

26

- ▶ You previously used tables to create the layout of your webpages. They make it easy.
- ▶ However, tables are only supposed to be used for tabular data, not for webpage layouts.
- ▶ In this course, we will move away from table layouts. We will use the HTML elements instead.

The best HTML element?

27

- ▶ One element that hasn't been taught yet is called the `div` (short for division/divider).
- ▶ This is essentially a panel that may contain other divs or different HTML elements, mainly used for creating layouts without a table.
- ▶ When in doubt, use a `div`!

The best HTML element?

28

- ▶ We will learn about CSS over the next two weeks. Then we can apply styles to our divs to form a layout.
- ▶ For now, our divs won't do very much until we add styles.
- ▶ You can create one anyway!
- ▶ `<div>`I'm so excited about my first div!`</div>`

More on layouts

29

- ▶ In CS1033, your websites contained 5 or 6 individual pages.
- ▶ Have you noticed a lot of modern websites have 1 very long page?
- ▶ Look at these sites:
 - ▶ <http://www.thebeet.ca/>
 - ▶ <https://www.mysteryescaperooms.com/>

More on layouts

30

- ▶ Both formats are fine!
- ▶ Individual short page format:
 - ▶ PRO: Keeps content separated.
 - ▶ CON: More files and links.
- ▶ Scrolling long page format:
 - ▶ PRO: One file, everything loads at once, clean on mobiles.
 - ▶ CON: May take longer for users to find the info they need.

Break

31

- ▶ Take a 10 minute break.
- ▶ We will discuss web forms after the break.

Forms

32

- ▶ What is a form?

The image shows two examples of forms. On the left is a 'LIABILITY WAIVER FORM' from the Canadian Red Cross, which includes fields for Group, Activity, Fiscal year, Age Number, and Amount approved. On the right is a '2019 Personal Tax Credits Return' form, which includes a table for tax credits and a section for the taxpayer's signature.

Forms

33

- ▶ Notice that the forms are asking people to fill in some kind of info.
- ▶ Forms can be used in websites too.
- ▶ Made with <form> tag.
- ▶ What kinds of user interaction have we seen so far?
 - ▶ Scrolling with mouse/finger
 - ▶ Clicking links with mouse/finger

Forms

34

- ▶ Web forms provide a way to receive more kinds of user input.
- ▶ Examples:
 - ▶ Typing their name in a textbox.
 - ▶ Selecting their year of birth from a dropdown menu.
 - ▶ Checking boxes to indicate which foods they like.

Forms

35

The image shows two examples of web forms. On the left is a 'Find a Location' form from Loblaw's, which includes a search bar and a 'SEARCH' button. On the right is a 'Create a New Account' form, which includes fields for First name, Last name, Mobile number or email, New password, Birthday, Gender, and a 'Sign Up' button.

Types of user input

36

- ▶ Text input:
 - ▶ Text
 - ▶ Textarea
 - ▶ Password
- ▶ Differentiate text, textarea, and password inputs.

The image shows three examples of text input fields. The first is a text box with the value 'London, ON'. The second is a textarea with the text 'I am a second year student at western in Computer Science.'. The third is a password field with masked characters 'XXXXXXXXXX'.

Types of user input

37

▶ Selection input:

▶ Radio button

☒ Male ☐ Female ☐ Other

▶ Checkbox

☒ Pizza ☐ Wings ☒ Garlic Bread

▶ Select list

CS

BMGS

IS

FMS

Other

- ▶ Differentiate radio buttons, checkboxes, and select list inputs.

Types of user input

38

▶ Buttons

▶ Submit button

▶ Reset button

▶ Generic button

- ▶ Differentiate submit, reset, and other generic button inputs.

Types of user input

39

▶ Other input:

▶ Hidden

▶ Date chooser

▶ Colour chooser

▶ And several others!

Types of user input

40

▶ Most input types are added like:

```
<input type="type" />
```

▶ i.e. `<input type="text" />`

▶ Some have their own tags:

```
<textarea></textarea>
```

```
<select>
```

```
<option>1</option>
```

```
<option>2</option>
```

```
<option>3</option>
```

```
</select>
```

Input attributes

41

▶ Name

- ▶ ID for the input which will help when we get to the JavaScript portion.

▶ Value

- ▶ Text displayed on/in the input.

▶ Placeholder

- ▶ Temporary, descriptive text.

Input attributes

42

▶ Tabindex

- ▶ Gives an order to the elements for users to "tab" through them.

First Name:	University:
Last Name:	Program:
Enter	

OR

First Name:	University:
Last Name:	Program:
Enter	

Labels

43

- ▶ Labels are often placed beside or above each of the input fields to indicate the purpose of that field.
- ▶ When you click a label, the associated field becomes **focused**.
- ▶ Focus means the field is activated so that the user can begin typing it in, or for other input types, it may mean selecting a radio button or checkbox.

Labels

44

- ▶ Link a label to an input field using the **for** attribute and referencing the field by its ID attribute.

```
<label for="email" id="Email">Email</label>
```

- ▶ Clicking a label for a radio button selects that button.

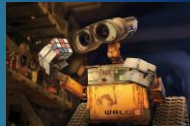
```
<label for="canada" id="Canada">Canada</label>
<input type="radio" name="country" id="canada" />
<label for="usa" id="USA">USA</label>
<input type="radio" name="country" id="usa" />
```

☒ Canada ☐ USA

Captcha

45

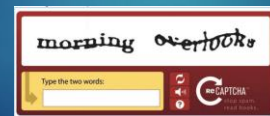
- ▶ Prevents bots from automatically filling out your web forms!
- ▶ Include something that only a human can answer.
- ▶ Some sites have a simple math question or ask you to type a word (i.e. the current month)



Captcha

46

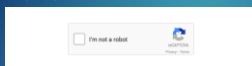
- ▶ Those can be sufficient for low-key websites but bot programmers may catch on to those questions.
- ▶ Bigger sites often have more complex captchas like this:



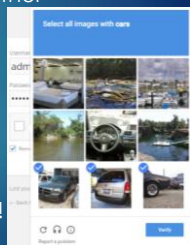
Captcha

47

- ▶ Google's reCaptcha is another popular method.



- ▶ Sometimes it will ask you to select images that contain a certain item.
- ▶ Very hard for bots to get in!



Form validation

48

- ▶ Some fields (inputs) are required while some may be optional.
- ▶ It should be clear to the user which are required. Asterisks (*) are often used to indicate required fields.
- ▶ Only make inputs required if you really require that info! People usually don't like entering tons of information unless it's necessary.

Form validation

49

New User Registration * Required Fields
Users under the age of 16 are not permitted to use this website.

Step 1: Choose your login information.
This information is used to log you in.
Username (4-20 characters) *
Password (4-20 characters) *
Re-enter Password *

Step 2: PUBLIC Information.
This information is visible to everyone.
First Name *
Last Name *
State or Province *
Country *
If you do not want your location displayed, please leave it blank out of consideration to users who use this feature to find and trade with gamers in their area.

Step 3: PRIVATE Information.
This information will subject to our [Privacy Policy](#).
Your Email *
Date of Birth: December 06 2016 *

Form validation

50

- ▶ Now the users can see which fields are required, but how do we enforce it?
- ▶ The **required** attribute.
- ▶ Look what happens when you push Enter without typing in a university.

First Name: Bryan
Last Name:
University (*)
Program:
Enter
Please fill out this field.

Form validation

51

- ▶ What else might need to be validated?
- ▶ What happens if the user's input doesn't match what is expected?
 - ▶ i.e. They need to enter a 4-digit PIN but enter 5 numbers. Or 3 numbers.
 - ▶ i.e. They need to enter their name but enter a number.

Form validation

52

- ▶ Text inputs can be given a "maxlength" attribute to limit how many characters can be entered.
- ▶ There's no "minlength" (yet). ☹
- ▶ There's also no way in HTML to restrict which types of characters are allowed. ☹
- ▶ Other languages can do this!

Form validation

53

- ▶ Which language to use for this?
 - ▶ Good question! It depends on:
 - ▶ What you need to do with the information.
 - ▶ System (computer or server) specs.
 - ▶ Whether you need the validation or processing done in real-time or on submission.

Server vs client processing

54

- ▶ Form validation and processing, and other functions can be done on the **server side** or the **client side**.
- ▶ Server side work is done on the host machine, while client side work is done on the user's device.
- ▶ PHP is a common server side language. JavaScript is used for client side coding.

Server vs client processing 55

- ▶ How do you know which work should be done on which side?
- ▶ Server side is used for:
 - ▶ Databases / external files.
 - ▶ Processing orders (e-commerce).
 - ▶ Functions involving security.
 - ▶ Anything that the host needs to know about or track.

Server vs client processing 56

- ▶ Client side is used for:
 - ▶ Interactivity / animations.
 - ▶ Non-secure form processing.
 - ▶ Receiving user input in real-time.
 - ▶ Loading or changing elements that appear on a user's device.
- ▶ Can you differentiate the two?

Server vs client processing 57

- ▶ Websites usually have both types of functionality for different features.
- ▶ What if we want to do server side work but to invoke (start) it based on user interaction? Can we mix client side interactivity and user input with server side work?
- ▶ Yes!

Server vs client processing 58

- ▶ AJAX is a technology that combines JavaScript with PHP or other server side languages.
- ▶ We can run server side code (i.e. find keywords in a database) triggered by user input (i.e. typing on a keyboard).

Tri
Trinidad and Tobago
Triangle H
Trigonometric functions
Trinity
Triangle
Trisha Yearwood
Tristan da Cunha
Trilobite
Trinity (nuclear test)
Trinitrophenol
Tri

Form validation 59

- ▶ How should we validate / process our web forms?
- ▶ Do you need to store the form data in a file or database? Are there private data like PINs or passwords?
 - ▶ Server side
- ▶ Do you need it to run in real time?
 - ▶ Client side

Form validation 60

- ▶ It also may depend on what system you are using.
 - ▶ JavaScript will run on your laptop right now.
 - ▶ PHP won't. ☹ It only runs on servers. You can install virtual server software for free if you want to run PHP locally.
- ▶ We won't do PHP in this course.

Form validation

61

- ▶ Do you think this Facebook signup form is validated with PHP or JavaScript?
- ▶ Hint: I did not click "Sign Up" but only clicked into each of the text boxes.

The image shows a screenshot of the Facebook 'Create a New Account' form. The form is titled 'Create a New Account' and has a sub-header 'It's free and always will be.' Below this, there are four input fields: 'First name', 'Last name', 'Mobile number or email', and 'New password'. Each of these four fields has a small red circle with a white exclamation mark inside, indicating a validation error. Below these fields is a 'Birthday' section with a date picker set to 'Dec 7 1993' and a question 'Why do I need to provide my birthday?'. There are radio buttons for 'Female' and 'Male'. At the bottom, there is a small line of text: 'By clicking Sign Up you agree to our Terms, Data Policy and Privacy Policy. You may receive SMS notifications from us and can opt out any time.' and a green 'Sign Up' button.

Closing remarks

62

- ▶ That's all for today.
- ▶ Have a great week!