CS 2033

Multimedia & Communications II

LECTURE 3 – HTML AND WEB FORMS

Websites from scratch!

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

Which browsers?

- 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

- The primary trinity of languages:
  - HTML5
  - CSS3
  - JavaScript

- Other languages:
  - PHP
  - ASP

HTML

- HTML: Hypertext Markup Language.
- Consists entirely of tags.
- Tons of elements, i.e. images, headers, tables, lists, paragraphs
- Standard language for making static websites.
- Files usually end in .html or .htm

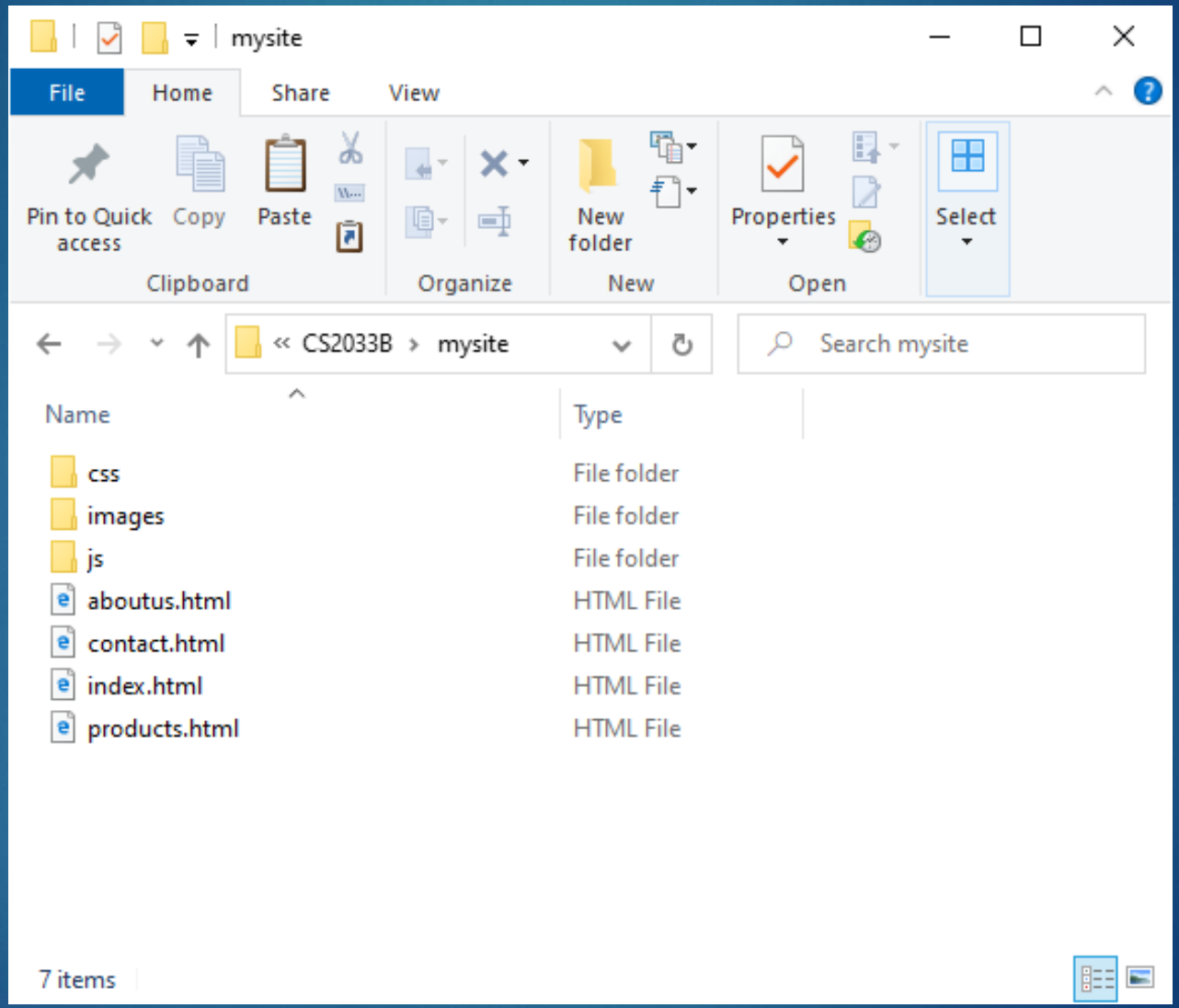
Files and folders

- 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

- 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.

Files and folders



Filepaths

- Links and images must have the filepaths linked correctly.
- 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

- 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

- 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

- Most tags need to be closed, so that content is surrounded by the opening and closing tags.
  - i.e. This is a **great** paragraph.
- Some tags are self-closing.
  - i.e.

Attributes

- HTML tags often have attributes which provide extra information about the content or design of the element.
  -
  - 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

- 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

- 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

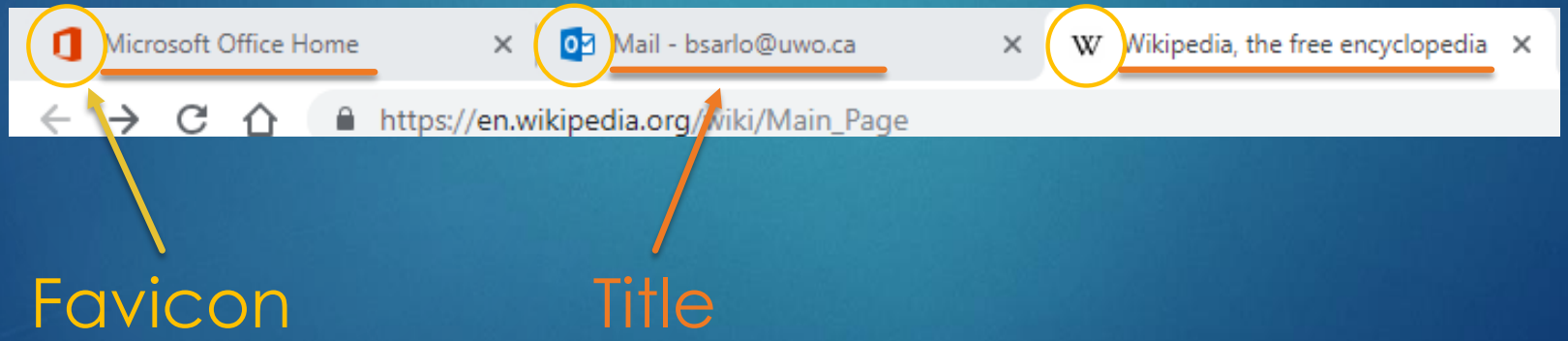

```
<!doctype html>
<html>
    <head></head>
    <body></body>
</html>
```

Title Property

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

Favicon

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



Favicon

- <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?

- 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!

- 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

- <h1>Largest header</h1>
- <h6>Smallest header</h6>
- Paragraph of text
- Link
-
- Bold text
- Italicized text

Common body elements

- creates an unordered list.
- creates an ordered list.
- adds a list item to a <ul> or

```
<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

- <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

- 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?

- 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?

- 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!
- I'm so excited about my first div!

More on layouts

- In CS 1033, 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

- 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.

Forms

- What is a form?

Canadian Heritage | Patrimoine canadien

LIABILITY WAIVER FORM

Must be returned within 30 days of receipt

Please complete this form, print it and have it signed and dated, and then submit with your application.

Officer to complete

Group:

Activity:

Fiscal year:

File Number:

Amount approved:

1. Activity will take place between

Canada Revenue Agency | Agence du revenu du Canada

2019 Personal Tax Credits Return

Protected B when completed

TD1

Read page 2 before filling out this form. Your employer or payer will use this form to determine the amount of your tax deductions.

Fill out this form based on the best estimate of your circumstances.

Last name	First name and initial(s)	Date of birth (YYYY/MM/DD)	Employee number
Address	Postal code	**For non-residents only** – Country of permanent residence	Social insurance number

1. Basic personal amount – Every resident of Canada can claim this amount. If you will have more than one employer or payer at the same time in 2019, see "More than one employer or payer at the same time" on page 2. If you are a non-resident, see "Non-residents" on page 2. **12,069**

2. Canada caregiver amount for infirm children under age 18 – Either parent (but not both), may claim $2,230 for each infirm child born in 2002 or later, that resides with both parents throughout the year. If the child does not reside with both parents throughout the year, the parent who is entitled to claim the "Amount for an eligible dependant" on line 8 may also claim the Canada caregiver amount for that same child who is under age 18.

3. Age amount – If you will be 65 or older on December 31, 2019, and your net income for the year from all sources will be $37,790 or less, enter $7,494. If your net income for the year will be between $37,790 and $87,750 and you want to calculate a partial claim, get Form TD1-WS, Worksheet for the 2019 Personal Tax Credits Return, and fill in the appropriate section.

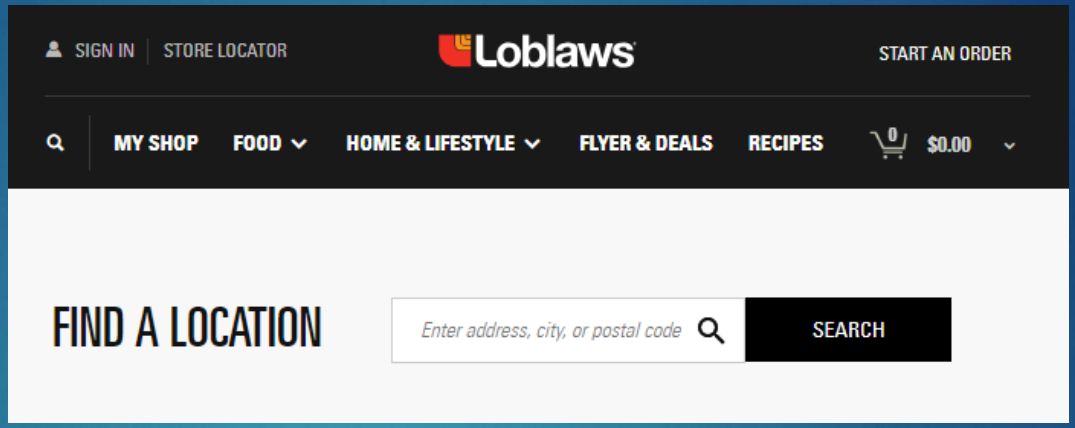
Forms

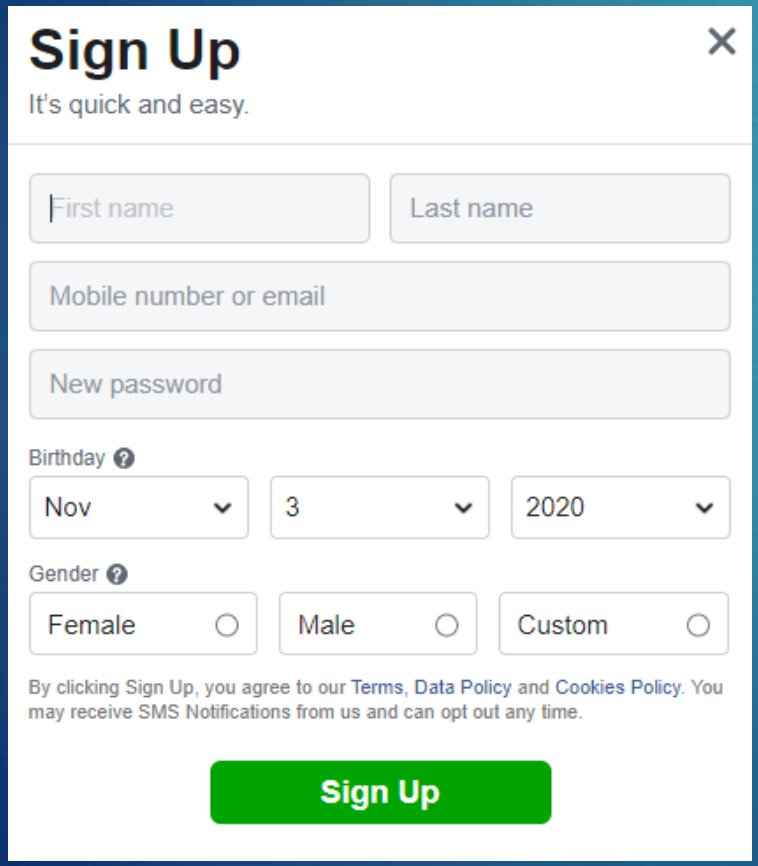
- Notice that the forms are asking people to fill in some kind of info.
- Forms can be used in websites too.
- Made with <form> tag.
- What kinds of user interaction have we seen so far?
  - Scrolling with mouse/finger
  - Clicking links with mouse/finger

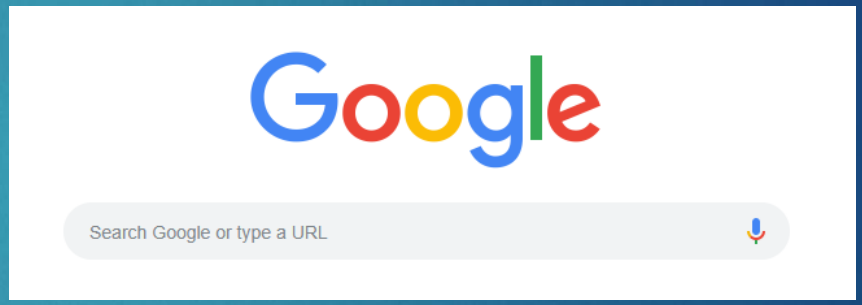
Forms

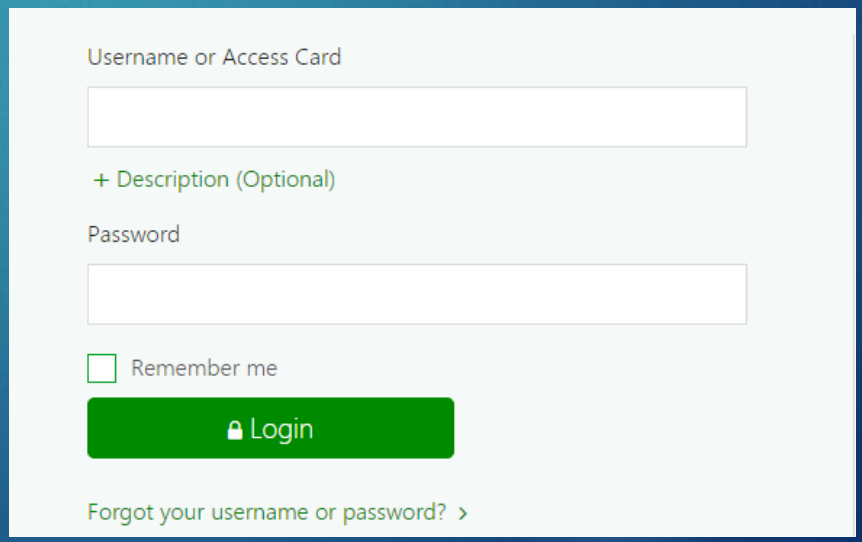
- Web forms provide a way to receive more kinds of user input.
- Examples:
  - Typing their name in a textbox.
  - Selecting their year of birth from a dropdown menu.
  - Checking boxes to indicate which foods they like.

Forms


SIGN IN
STORE LOCATOR
Loblaws
START AN ORDER
MY SHOP
FOOD
HOME & LIFESTYLE
FLYER & DEALS
RECIPES
$0.00
FIND A LOCATION
Enter address, city, or postal code
SEARCH

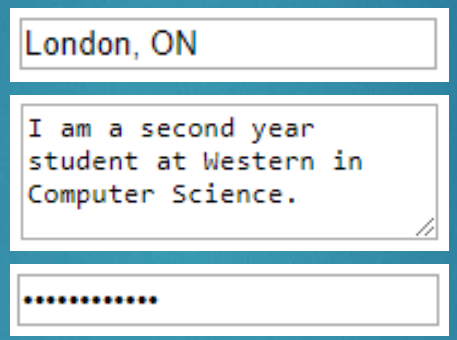

Sign Up
It's quick and easy.
First name
Last name
Mobile number or email
New password
Birthday
Nov
3
2020
Gender
Female
Male
Custom
By clicking Sign Up, you agree to our Terms, Data Policy and Cookies Policy. You may receive SMS Notifications from us and can opt out any time.
Sign Up


Google
Search Google or type a URL


Username or Access Card
+ Description (Optional)
Password
Remember me
Login
Forgot your username or password? ›

Types of user input

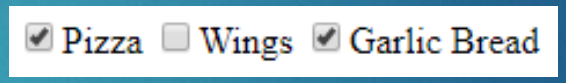
- Text input:
  - Text
  - Textarea
  - Password

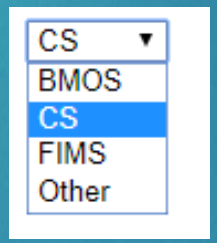


- Differentiate text, textarea, and password inputs.

Types of user input

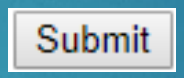
- Selection input:
  - Radio button
  - Checkbox
  - Select list



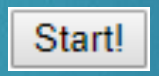


- Differentiate radio buttons, checkboxes, and select list inputs.

Types of user input

- Buttons
  - Submit button 

  - Reset button 

  - Generic button 

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

Types of user input

- Other input:
  - Hidden
  - Date chooser
  - Colour chooser
  - And several others!

Types of user input

- 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>

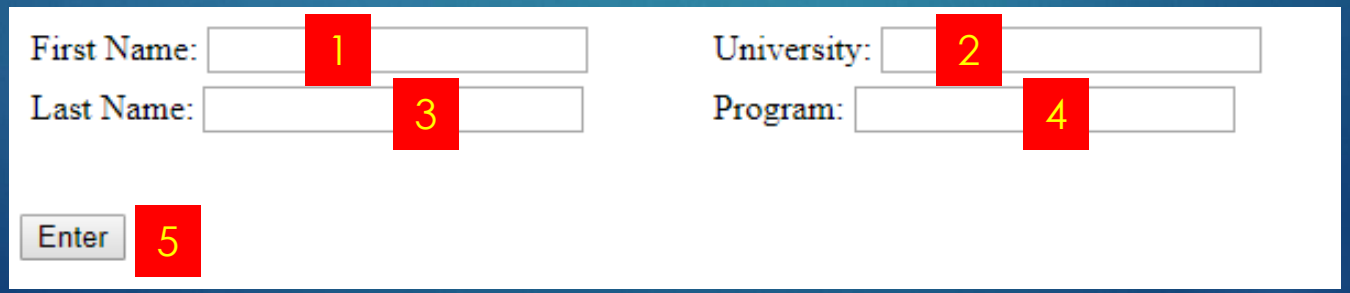
Input attributes

- 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

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

OR



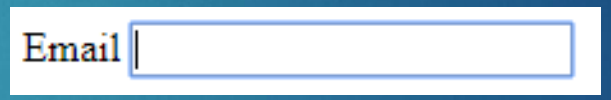
Labels

- 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

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

```
<label for="emailField">Email</label>
<input type="text" name="email" id="emailField" />
```



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

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

Captcha

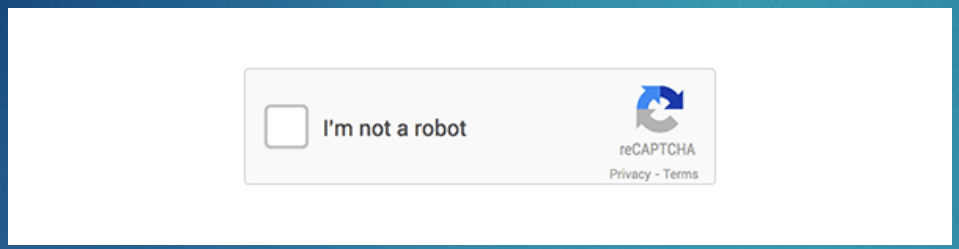
- Prevents bots from automatically filling out your web forms!
- Include something that only a human can answer.
- Some sites have a simple math question or ask you to type a word (i.e. the current month)

Captcha

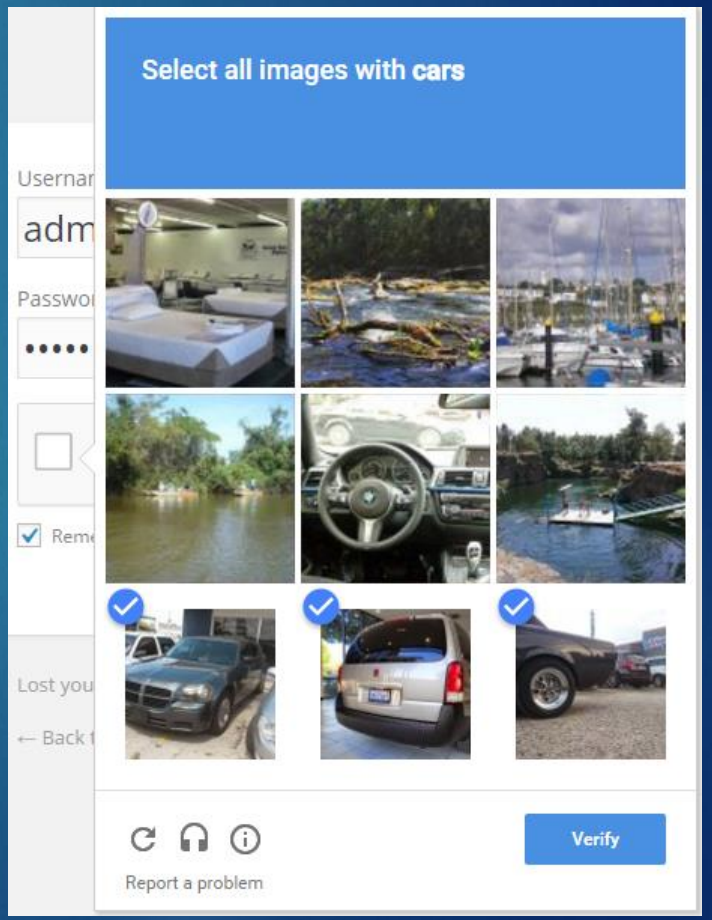
- Those can be sufficient for low-key websites but bot programmers may catch on to those questions.
- Bigger sites often have more complex captchas like this:

Captcha

- 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

- 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

New User Registration * Required Fields

Users under the age of 16 are not permitted to use this website.

Step 1. Choose your login information.

Username (4-20 characters)	*
Password (4-20 characters)	*
Re-enter Password	*

Step 2. PUBLIC Information.
This information visible to everyone

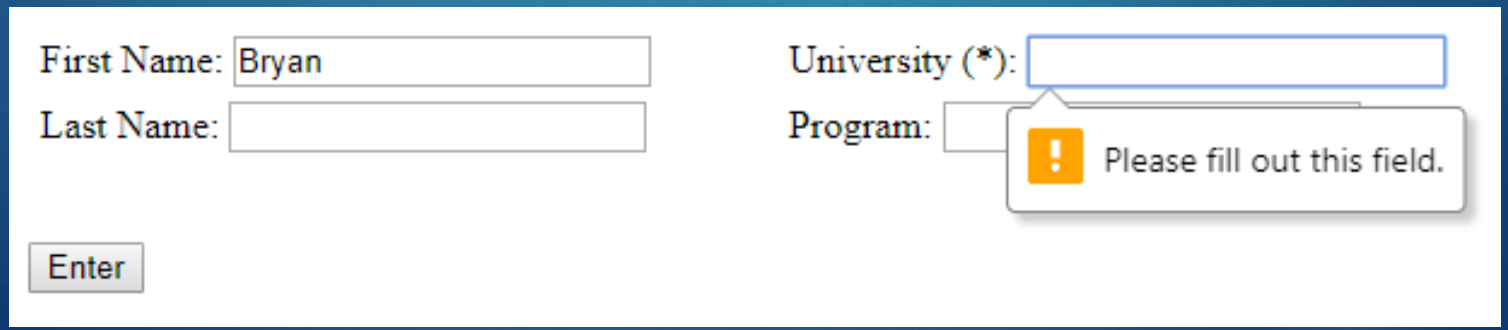
First Name	*
Last Name	*
State or Province	or
Country	If you do not want your location displayed, please leave it blank out of consideration to users who use this feature to find and trade with gamers in their area.

Step 3. PRIVATE Information.
This information will subject to our Privacy Policy

Your Email	*
Date of Birth	December 06 2018 *

Form validation

- 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.



Form validation

- 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

- 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

- 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

- 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

- 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

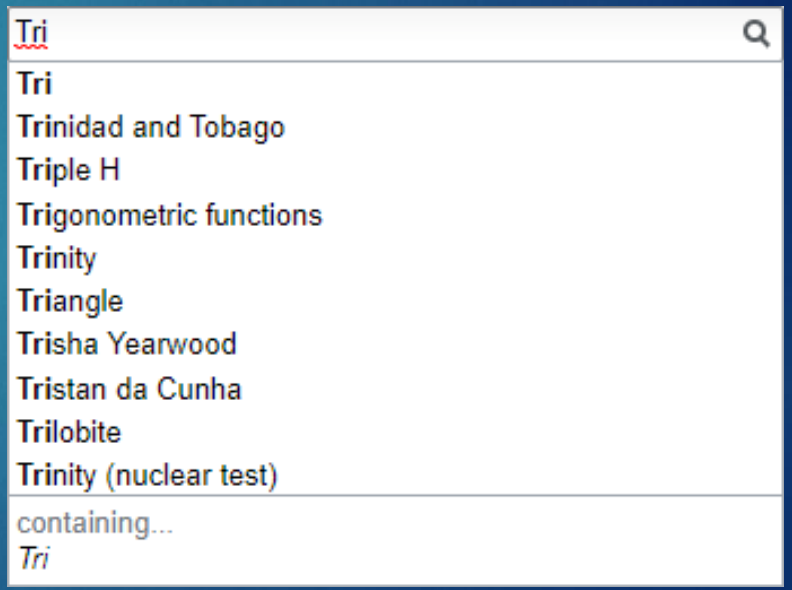
- Client side is used for:
  - Interactivity / animations.
  - Non-secure form processing.
  - Receiving user input in real-time.
  - Loading or changing elements that appear on a user's device.
- Can you differentiate the two?

Server vs client processing

- 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

- 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).



Form validation

- 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

- 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

- 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.