# In Class Website

Your client has provided you with the layout design. This is the layout you are to use for your in class website. Review the layout design then complete the steps below. The layout design can be found in the *In Class Website* section of this document.

Home Page

1. Load your html template file (MyTemplate.html) in Visual Code.
2. From the File menu, pick Save As. Save it to the COMP205\YourLastNameThenYourFirstNameInClassWebsite**V1**folder, calling it index.html. This will be the home page for your website.
3. Add relevant text for the title tag.
4. Add h1 and h2 headings.
5. Don’t worry about the hyperlinks for now.
6. Add multiple paragraphs. The content is to be 200 to 400 words of content related to your topic. This page is the user’s introduction to your website, company and/or products.
7. Include your footer text below the list.
8. Test your code before moving to the next point.
9. Somewhere below the headings and above the footer, add an ordered list with 4 (or more) points. Test it.
10. Add a nested unordered list inside the third point. Include two or more items in the list. The content of your lists should relate to your website topic.

# In Class Website

Goal: design and create the Tables page for your in class website. Remember to use the layout design provided by your client.

Copy your index page to a file called TablePage.html. Delete the paragraphs and lists. Change the title element. Using the Table Building Steps two pages back, code a table. The table should have a minimum of 5 rows and 4 columns **with relevant data inserted** in the cells. Include at least one colspan and one rowspan. The first row should be bolded headings.

# In Class Website

The goal now is to implement the hyperlinks in your website. Remember to use your client’s layout design.

The layout design shows links to four pages.

* Work on the Home page first. Create four links: Home, Table, Images, Form. When the link is clinked, the new page should open in the same tab. For now, the Images and Form page links do not work and the links will not be visually appealing.
* Once you have done the Home page, copy and paste the links to the Table page. The same four links should appear on every page of the site.
* Using your college email, add an email link to the footer of the Home page. Once you have tested it, copy the email link to the Table page.

Work through *Checklist for in class website (at end of Hyperlinks topic)*. It can be found in the in class website section of this playbook. Make sure your site contains all the items on the checklist before moving on. If something is missing, add it to the site.

# In Class Website

Remember your client’s design layout.

Compressing Images

1. Create a folder called imgs inside your in class website folder.
2. Our next step will be to add images to the website. There will be 25 – 50 words below each image talking about the specific image.
3. Earlier in the course, you downloaded images for your in class website. Move the images to the new imgs folder. There should be 8 – 12 images (no more, no less).
4. Find small logos for Facebook and Twitter and save them to the new images folder. These two hyperlinks will appear on every page.
5. Use a tool to compress and resize each image.
6. Run Windows Explorer and find the images. The compressed images should be much smaller. Delete the original large image files.

Adding Images to the Website

Add a new page to your in class website. Call it ImagesPage.html.

1. The first goal is to add a Facebook link and a Twitter link. To do this:
   1. Code the page so that the downloaded Facebook logo is a hyperlink that opens in a new tab. Tips:
      1. Write a line of code to create an external link to the facebook home page.
      2. Write a line of code to create an image. Set the height and width to 20.
      3. Combine the two lines in to one.
      4. If you get a border that you don’t like, set the border attribute of the img element to "0".
   2. Do the same for the Twitter link.
   3. Test fully.
2. When you are sure the Images page is done correctly, add the facebook and twitter link to the other pages. Test fully.
3. The next goal is to add images to your Images page. You will create a figure element for each image. Reference: <https://developer.mozilla.org/en-US/docs/Web/HTML/Element/figure>. To do this:
   1. Move the cursor below the hyperlinks.
   2. Open a figure tag.
   3. Place image element here. Set the height of the image to 250 and width to 300 (or similar values).
   4. Code a figcaption element. The caption text should be less than 8 words.
   5. Close the figure tag.
   6. Below the figure element, add 25 – 50 words about the image.
   7. Test fully.
   8. Repeat for the remaining seven images.
4. Next, try changing one of the filenames to be invalid and run it. This will test the alt attribute. Fix the filename to the correct spelling.

# In Class Website

Design a colour theme for your in class web site. References are provided below:

<https://www.w3schools.com/colors/colors_names.asp>

<http://www.w3schools.com/cssref/css_websafe_fonts.asp>

<https://www.webfx.com/web-design/color-picker/>

<https://color.adobe.com/explore/>. Type a colour in the textbox to see palettes in that colour theme.

Create a css folder inside your web site folder. Create a css file in the new folder. Add your name in a comment at the top: /\* your name here \*/. In your html code, link to the css file. You will have to place this link in all html files on your site.

Create the following css styles. Each style should be unique.

1. Choose font name, font size, font colour and background colour for the body.
2. Ordered list numbering should display as A, B, C etc.
3. Define a style to set the unordered list bullet to a square.
4. Set width of table to 80% with a margin of auto. Set border collapse to collapse.
5. Write css code to define unique colours for the 3 hyperlink states. Set an opacity of 0.5 for hover.

# In Class Website

1. Today you will begin the final version of the in class website**. Before making any changes, it is important that you make a copy of the website and work with the copy**. Copy the files in your version 1 folder to your version 2 folder.
2. Be sure that you are now working with the version 2 folder.
3. At this point, the h1 and h2 elements are no longer needed. You may keep them or remove them but they should be on all pages or no pages.
4. Working with the Home page:
   1. Using a div element, create a class that encompasses the six hyperlinks. Name the class **navflexcontainer**. This name will make sense in our next topic. Code the class with a bottom margin of 2em.
   2. Using a div element, create a class that encompasses the content below the navigation and above the date and contact information. Name the class **maincontainer**. Style it with margins, padding and background colour. If desired, set a min-width of 20em (or em value of your choice).
   3. Using a div element, create a class to encompass the contact information and date information (coming later). Name the class **footercontainer**. Style it with font colour, background colour, margin, text align and padding.
5. Once the Home page is working fully, modify the Table page to include navflexcontainer, maincontainer and footercontainer.
6. Modify the Image page to include navflexcontainer and footercontainer. Do not add the maincontainer div to the Image page.
7. Choose one word (or a few words) on your Home page to highlight using a span element. It can be company name, topic name or any other. It should appear in italics with a noticeably defined font colour or background colour. The word needs to appear 4 or more times on the page. If it does not, add it.
8. Run the developer tool. Click on the various elements and examine the styles and the style precedence. Try resizing the webpage/code window to test the current responsiveness of your website. Try out the icon *Toggle device toolbar*.
9. Work through *Checklist for in class website (at end of Grouping Elements topic)*. It can be found in the in class website section of this playbook. Make sure your site contains all the items on the checklist. If something is missing, add it to the site.

# In Class Website

If you like, you can change the style for ol and ul in your **V2** copy. Do not change the V1 copy.

Navigation

In our last topic, you created a div class titled navflexcontainer. The goal now is to turn the navigation menu in to a flex layout. Navflexcontainer will have multiple flex items.

1. Start with the Home page.
2. Create a flex item, named **navflexitem**, for each of the six hyperlinks.
3. Create a flex layout for the navigation. Add padding, background colour and font settings to the navflexitem style. Do not add flex:1 to this style. Try adding a text shadow (optional):

text-shadow:0.2em 0.2em 0 rgba(0,0,0,0.2);

1. We want to create blank space between the Form link and the social media links. After the Form flex item, create a new flexitem div. The contents of the new div should be only a hard space. Type *&nbsp;* between the open tag and the close tag.
2. To create wide space between the Form link and the social media links, define the new extra space flex item to be three times wider than the other items.
3. To create a smaller space for the social media flex items, define each one to be 0.1 in relation to the other items.
4. Test fully to ensure your site is responsive. Resize the browser window slowly to examine how the page looks at various stages.

Once you are sure the above is working fully, modify the navigation on the other pages to mirror the Home page. Test fully.

# In Class Website

Verify your in class web site has a fully implemented flex layout that is responsive on all pages. Your goal now is to create a new page with input elements. Think of an appropriate form for your website topic. It can be a shopping basket, a contact us page or a form that matches your topic. Create a new webpage following your client’s design plan. Add the following elements inside the maincontainer div:

* Form element with styled border.
* Text input box.
* Text area.
* List box.
* Checkboxes (at least 2).
* Radio buttons (at least 2).
* Date element.
* Time element.
* Try out some of the input element attributes.
* Add a style of your choosing for the labels.
* Set a unique font colour for the data entered in the input boxes.
* Make sure the forms page works properly when resized.
* Optional: add additional styles.

**To earn the marks, your code must be tailored to your topic. It cannot be the same code or content as the Playbook.**

# In Class Website

Current Date

Add the current date to the footers. The date should appear in the format:

mmmmm dd, yyyy

For example, *December 10, 2023*. See provided file getMonthName.js.

To have the date displayed automatically when the page loads add an id to the open body tag then code the javascript. If you would like a hint, start with the highlander5 date example. The event listener is a little different than our previous ones (click, change), in that brackets have to be added after the function name:

document.getElementById("loadingBody").addEventListener("load", showDate());

Jump To Top Button

Add a Jump To Top button at the bottom of the Images page. The javascript code to execute it is shown below:

window.scrollTo({top: 0, behavior: 'smooth'});

Change Table

The goal now is to give the user the ability to change the look of the table. To do this in javascript, you must have a border on your table. Verify that you have a table border defined in your css code. Add a button to the Table page. The button should read Change Table. Add an id to the open table tag. Write javascript code to style the table as follows:

* font colour green (keyword color).
* border width 0.3em (keyword borderWidth).
* border style: dotted (keyword borderStyle).
* border colour: yellow (keyword borderColor).

Note that you are only changing the outside border. Once you have it working, you can change the above values, making sure the change is noticeable. Try out various border styles: dashed, groove, ridge, hidden, solid, double.

See debugging tips on next page…

Debugging Tips

1. Both the event listener for the jump button and the event listener for the change table button need an if statement:

if (reference to html element)

create event listener

Without the if statement, the code fails. This is because the jump button is on the images page, but not other pages. When you load the tables page, the javascript will fail because it can’t find the jump button. If javascript fails, the remaining code is ignored. The same will happen with loading the images page, given the change table button is not on the images page.

1. If your js code is not working, check to see if you have a colour defined in a th or td style. If you do, comment it out and try the js code again. Setting a colour in the table style should not conflict with the js code.

**Page Layout for In Class Website Pages**

The main heading will appear on every page. Think of it as a banner at the top of every page on the site.

A second heading appears here. It will be slightly smaller than the main heading.

Your site will have the six hyperlinks shown below. You may add extra pages or links if you like. The headings can be above or below the hyperlinks, your choice.

Home Table Images Form <the space between the links will be created in a later topic > A white and blue thumb up

Description automatically generated A black square with a white bird in it

Description automatically generated

This part of the page varies for each page.

Write your footer here. Include text you want on the footer of every page. Include the current date. Include a link to your college email address. It should display in the following format:

Your first and last name **|** Your college email address | Current Date

# Instructions

Instructions for building the website can be found in each topic in this playbook. To perform a search, use the softcopy of this document (on Blackboard).

# Checklist for In Class Website (at end of Lists topic)

# index.html

1. Load your html file in IE. You should have:
   1. The tab of the page should have a relevant title that is unique to this page.
   2. An h1 and an h2 heading.
   3. Steps described in the Headings and Lists section of this document under heading *In Class Website.*
   4. A footer that will be common to every page.
2. Load your html file in Visual Code. You should have the following elements:

* doctype
* your name in a comment
* html lang
* head
* meta
* title
* body
* h1
* h2
* p (for paragraph)
* footer

# Checklist for In Class Website (at end of Hyperlinks topic)

Elements for Every html Page

* doctype
* your name in a comment
* html lang
* head
* meta
* title
* body
* h1
* h2
* four hyperlinks
* footer with email link

Additional Elements

* Home page
  + - Steps described in the Headings and Lists section of this document under heading *In Class Website.*
* Table page
  + - A 5x4 (or larger) table with data relating to your website. Includes table headings. Includes at least one rowspan and colspan.

# Checklist for In Class Website (at end of Grouping Elements topic)

Elements for Every html Page

* doctype
* your name in a comment
* html lang
* head
* meta
* title
* link rel
* body
* six hyperlinks
* footer with email link

Additional Elements

* Home page
  + - Steps described in the Headings and Lists section of this document under heading *In Class Website.*
* Table page
  + - A 5x4 (or larger) table with data relating to your website. Includes table headings. Includes at least one rowspan and colspan. Table is not full width of the page and has unique border width and colour.
* Colour scheme as described in CSS Styles section of this document.
* Steps described in the Grouping Elements section of this document under heading *In Class Website Exercise*

# Final Checklist for In Class Website

Elements for Every html Page

* doctype
* your name in a comment
* html lang
* head
* meta
* title
* link rel
* body
* six hyperlinks
* footer with current date and contact us email link

Additional Elements

* Home page
  + - Steps described in the Headings and Lists section of this document under heading *In Class Website.*
* Table page
  + - A 5x4 (or larger) table with data relating to your website. Includes table headings. Includes at least one rowspan and colspan. Table is not full width of the page and has unique border width and colour.
* Colour scheme as described in CSS Styles section of this document.
* Steps described in the Grouping Elements section of this document under heading *In Class Website*
* Steps described in the Layouts section of this document under heading *In Class Website*
* Steps described in the Forms section of this document under heading *In Class Website*
* Steps described in the Javascript section of this document under heading *In Class Website*
* Web site is responsive. All pages work properly when resized and zoomed.