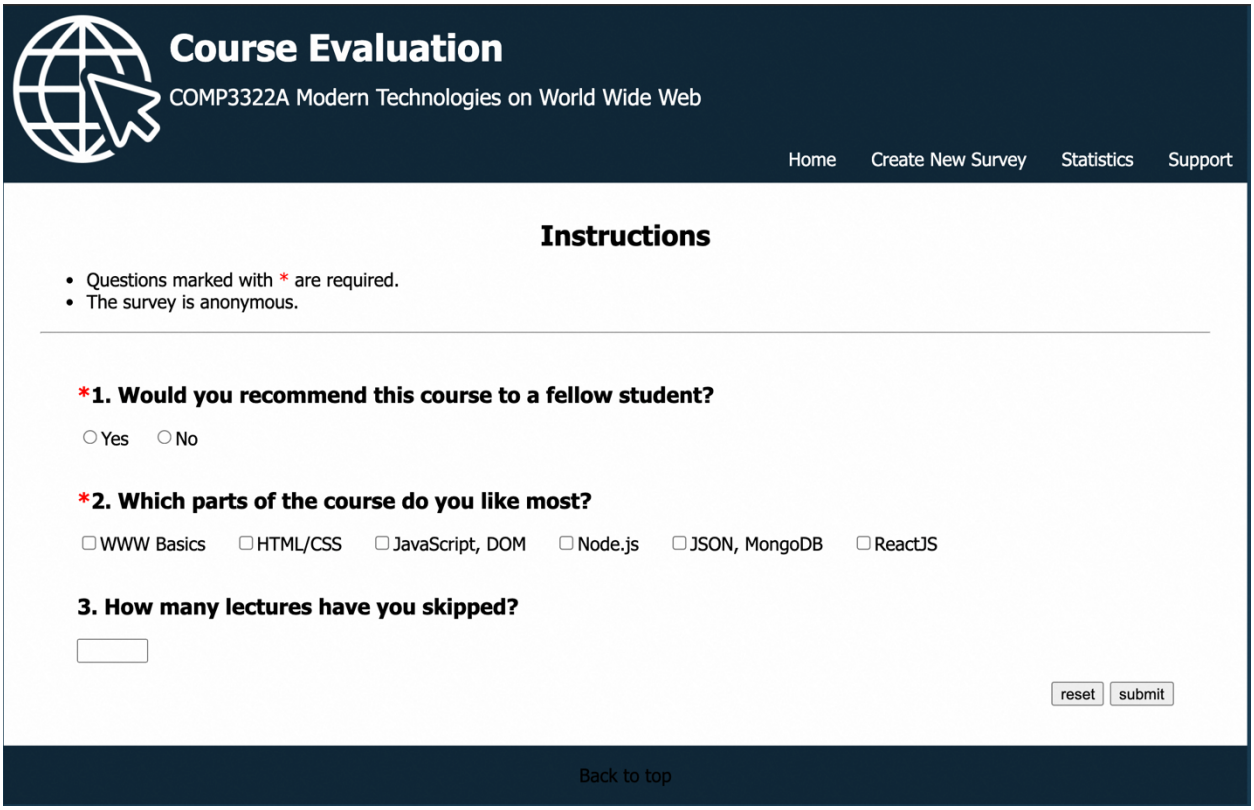
COMP3322A Modern Technologies on World Wide Web

Lab 1: HTML and CSS

Overview

In this lab exercise, we will create a web page in pure HTML and CSS that looks like the following:



Note: Due to compatibility of difference browsers, the same CSS code may lead to different display; we will check the assignment in the Google Chrome browser. Therefore, you are recommended to use the Chrome browser to develop/test your page as well.

Lab Exercise

Part 1. Preparation

Step 1. Download the template code from Moodle

Download the code template "**lab1_materials.zip**" from HKU Moodle, and extract it to a folder. In this folder, you will find files structured like the following:

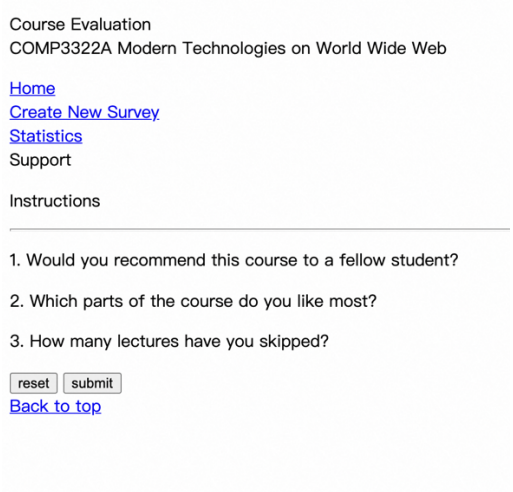
```
/
├─ assets
│  ├─ images
│  │  └─ logo.png
│  └─ stylesheets
│     └─ main.css
│     └─ basics.css
└─ index.html
```

"**index.html**" and "**main.css**" are the main files we will be working on. "**basics.css**" provides the styles for many basic elements that we will NOT modify, and is imported in "**main.css**". "**logo.png**" is the logo of our page.

Part 2. Build the HTML page

Step 2. Create section headings

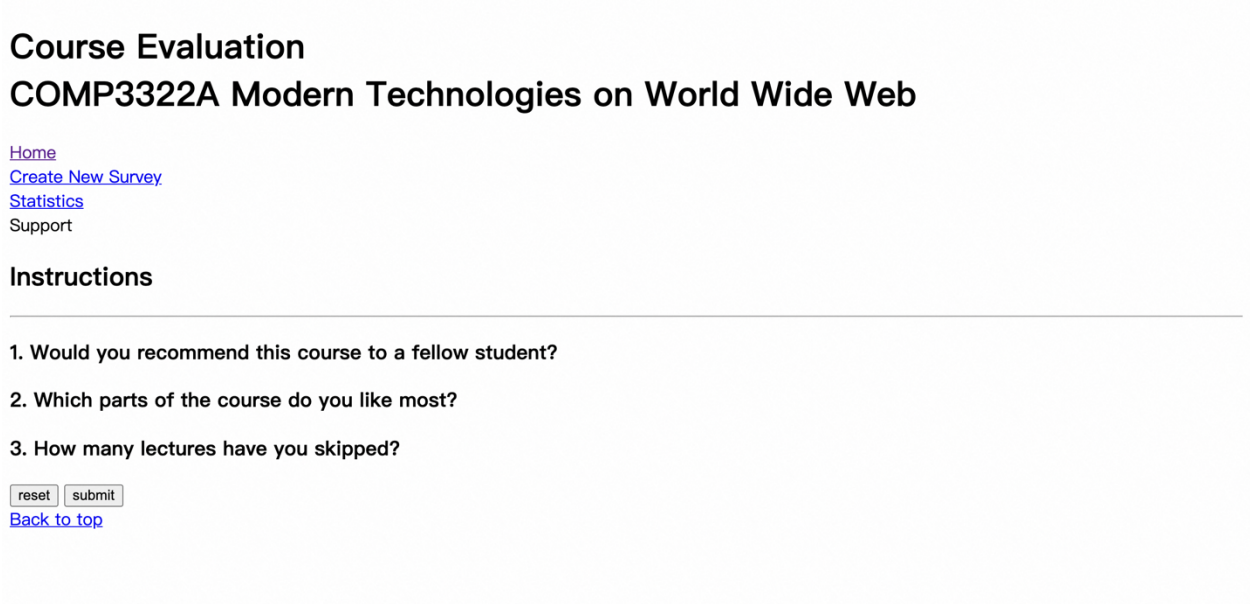
If we open "**index.html**" in the browser now, we'll see a page like this:



This page currently shows the main page heading, the nav bar, an empty instructions section, and three questions with no response fields. Currently all texts are of the same size. We will now set the section headings properly. Your tasks are:

- Change the main page heading ("Course Evaluation COMP3322A Modern Technologies on World Wide Web") to use **<h1>** instead of **<p>**.
- Similarly, change the instructions heading using **<h2>** and the three questions to use **<h3>**. (Don't modify the "class" attributes of the questions)

The page will look like this after this step:

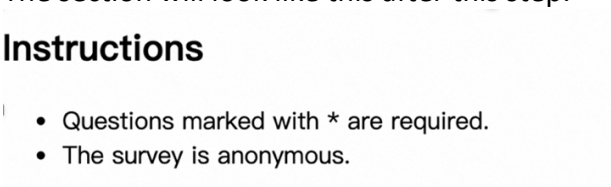


Step 3. Fill in the "Instructions" section using unordered lists

Now we will add contents to the "Instructions" section. Under the "Instructions" heading (within the ** ** container), we add an **unordered list** with the following two items.

1. Questions marked with * are required.
2. The survey is anonymous.

The section will look like this after this step:



Step 4. Add a link to the course moodle forum

Now we add a hyperlink to the course's moodle forum (https://moodle.hku.hk/mod/forum/view.php?id=2655658) for the "Support" entry in the **<nav>** container. When set correctly, the "Support" entry will become blue with an underline, and clicking it leads to the course's moodle forum.

Step 5. Create the question choices

Now we add the form elements under the three questions. Under the first question ("Would you recommend this course to a fellow student?"), add two **radio buttons** (https://www.w3schools.com/tags/att_input_type_radio.asp) with "Yes" and "No" **labels** (https://www.w3schools.com/tags/tag_label.asp), respectively. Set id, name, value attributes of the radio button elements. Set "for" attributes of the label elements.

Hint 1: the "name" attributes of radio buttons should be the same in a radio button group. If the buttons are grouped correctly, clicking one of the buttons automatically deselects others.
Hint 2: "for" attributes of the label elements should match the "id" attributes of the radio buttons. When set correctly, clicking on the label text should select the corresponding radio button.

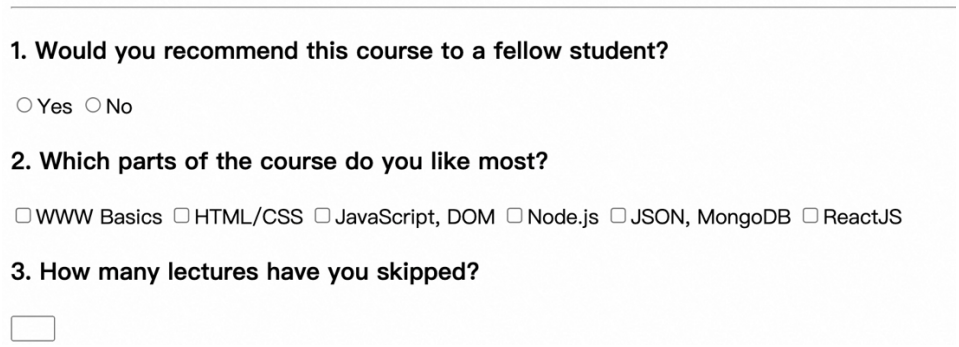
Under the second question ("Which parts of the course do you like most?"), create 6 **checkboxes** (https://www.w3schools.com/tags/att_input_type_checkbox.asp) with the following labels:

1. WWW Basics
2. HTML/CSS
3. JavaScript, DOM
4. Node.js
5. JSON, MongoDB
6. ReactJS

Set id, name, value attributes of the checkbox elements. Set "for" attributes of the label elements.
Hint: "for" attributes of the label elements should match the "id" attributes of the checkboxes. When set correctly, clicking on the label text should check the corresponding checkbox.

Under the third question ("How many lectures have you skipped?"), add a **number input** (https://www.w3schools.com/tags/att_input_type_number.asp). Set the "min" and "max" attributes of the number input to "0" and "3", respectively.

After this step, the page will look like:



Congratulations! By now we have finished all HTML elements of the page. We will then make the page look nicer by adding a stylesheet.

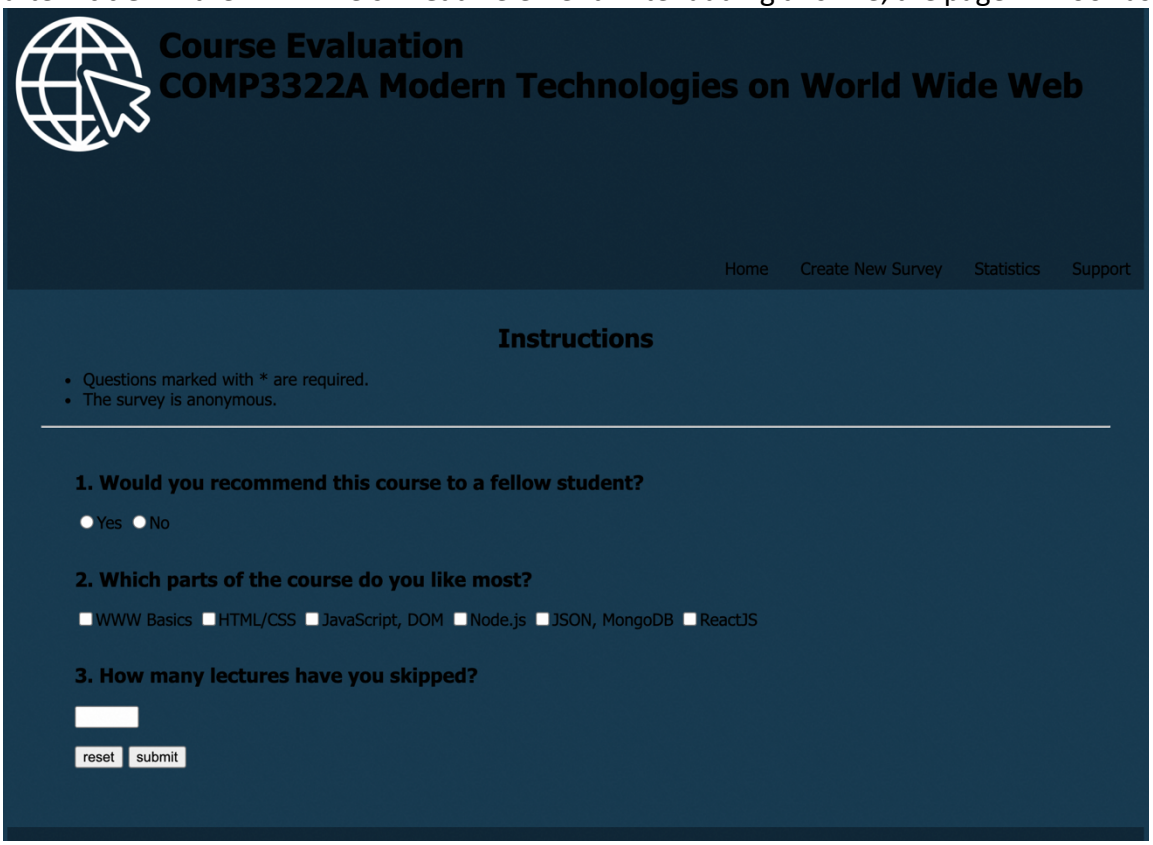
Part 3. Add CSS styles

Step 6. Link the HTML page with the stylesheet

Now we will link our HTML page with the provided CSS template file. This is done by adding

```
<link rel="stylesheet" href="assets/stylesheets/main.css">
```

after **<title>** in the HTML file's **<head>** element. After adding this line, the page will look as follows:



Step 7. Modify text and background color

Opening the "**main.css**" file, we can see an import rule. The import rule imports the content of "**basics.css**" file into "**main.css**". Please DO NOT MODIFY the import rule nor the "**basics.css**" file. Instead, we will build our new styling rules in "**main.css**".

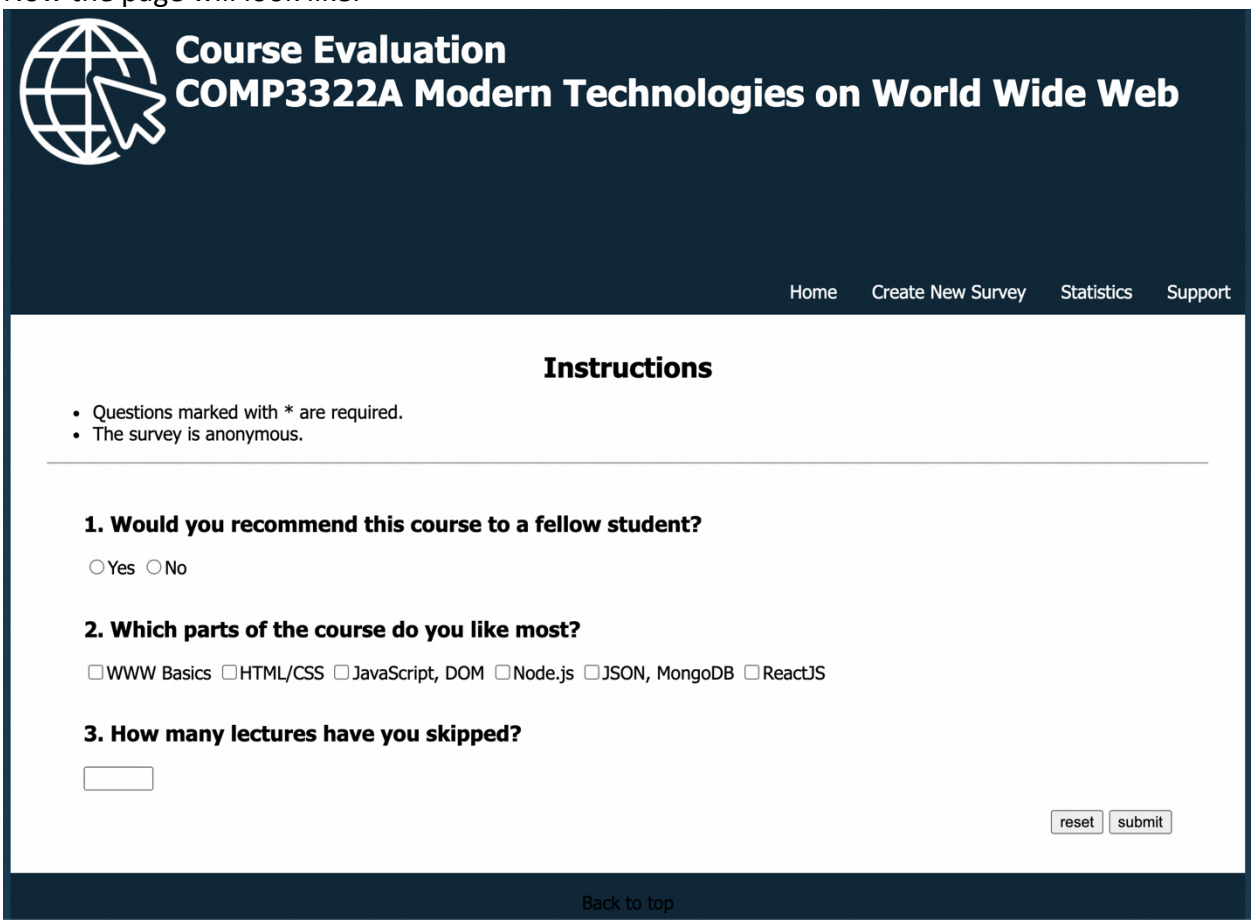
The first thing to do is to change the text and background color so the contents are easier to read. We do this in two steps:

1. We can see an empty set of styling rules for elements in **<header>** section and the **<nav>** section. Add a rule there to change the **text color** in the **<header>** and **<nav>** sections to **white (#FFFFFF)**. (https://www.w3schools.com/css/css_text.asp)
2. There is also an empty set of styling rules for the class "contents". If we look at the HTML file, we will see this class is specified on the **<section>** container that holds most of the contents we added in Part 2. Add a rule for this selector to change the **background color** to **white (#FFFFFF)**. (https://www.w3schools.com/cssref/pr_background-color.asp)

Step 8. Make the action buttons float to right

Now we want the action buttons ("Reset" and "Submit") appear at the right side of the page. Similar to Step 7, we can see an empty set of styling rules for the selector "#actions". Add rules in there to make the corresponding ** float to the right** (https://www.w3schools.com/cssref/pr_class_float.asp)

Now the page will look like:



Step 9. Reduce the font size of the course name

In this step, we reduce the font size for the course name ("COMP3322A Modern Technologies on World Wide Web") **without changing the size of the main title** ("Course Evaluation"). To do this, wrap the course name in a **** element with a "class" attribute, and then add CSS rules as follows for the class:

```
{
    font-size: 0.6em;
    font-weight: normal;
}
```

Now the page header should look like:

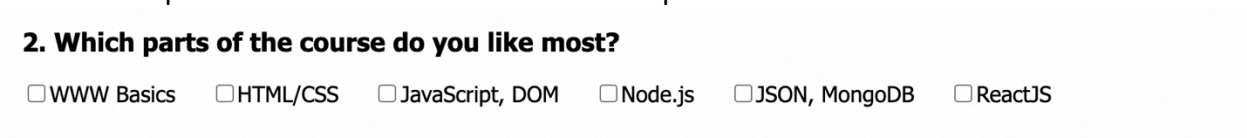
Step 10. Add spaces between checkboxes

The choices underneath the second question are dense. In this step, we will insert spaces after each checkbox. Create a selector that chooses **label elements that follow input elements whose "type" attributes are "checkbox"** and apply the following style:

```
{
    margin-right: 20px;
}
```

Hint: use attribute selectors (https://www.w3schools.com/css/css_attribute_selectors.asp) to select the checkboxes and use adjacent sibling selectors (https://www.w3schools.com/css/css_combinators.asp) to select the label elements following them.

The second question will look like this after this step:



Step 11. Add hover effects to the navigation links

We use the "**pseudo-class**" selector to change the background color of navigation links when mouse is **hovering** over them (refer to page 15 of lecture notes 5_cssbasics_COMP3322A_f2022.pdf and https://www.w3schools.com/csSref/sel_hover.asp). The background-color should be specified using:

```
{
    background-color: #163a50;
}
```

After this step, the navigation bar should look like this when the mouse is over the "Statistics" entry (mouse is not shown in the screenshot):

Note that we are actually implementing a transition effect of the background color when the mouse is over a navigation link. See the styling rule "transition: background-color 200ms linear;" on the navigation bar that we have provided in **basics.css**, and refer to https://www.w3schools.com/css/css3_transitions.asp to understand how the transition works.

Step 12. Add red asterisks for required questions

In this step, we add a red asterisk (*) to hightlight required questions (class="required"). Write a CSS rule that selects the **::before pseudo-element** (refer to https://developer.mozilla.org/en-US/docs/Web/CSS/::before) of required questions and set it to a red asterisk. Replace the * in "Questions marked with * are required" under instructions in the HTML file (added in Step 3) by an empty span (****), to ensure it looks the same as those * before the required questions.

Now you have finished the entire lab! The resulting web page should look like the screenshot on the first page of this document.

Submission:

Please finish this lab exercise before 23:59 Thursday Sep. 22, 2022. Please compress the entire project folder (i.e., the folder containing your **index.html** and the entire **assets** folder with **main.css** modified) into a .zip file and submit it on Moodle.

(1) Login Moodle.
(2) Find "Labs" section and click "Lab 1".
(3) Click "Add submission", browse your .zip file and save it. Done.
(4) You will receive an automatic confirmation email, if the submission was successful.
(5) You can "Edit submission" to your already submitted file, but ONLY before the deadline.