ASSIGNMENT 1:
BOOTSTRAP WEB PAGE
WITH A WEB FORM

RELEASED: **4 FEB 2021**
DUE: **26 FEB 2021 23:59**

SYNOPSIS

You are going to build a web page to introduce yourself, with a basic comment system. This page should be well-decorated and responsive. There are also some special features.

BASIC CONTENTS (10%)

(Done in Lab 1) The contents should be about yourself. You are not required to put in completely factual contents, but at least your name and department should be correct.

These HTML elements ***must be*** included in your HTML5 file:

♦   Heading

♦   Paragraph

♦   List

♦   Table

♦   Image / Picture

♦   Flex / Grid

These HTML5 semantic elements ***must be*** used appropriately:

♦   `<header>`

♦   `<nav>`

♦   `<section>`

♦   `<footer>`

STYLING (10%)

(Done in Lab 1) Using CSS and Bootstrap, at least these ***must be*** applied to your page:

♦   Header background image

♦   Horizontal link items in navbar

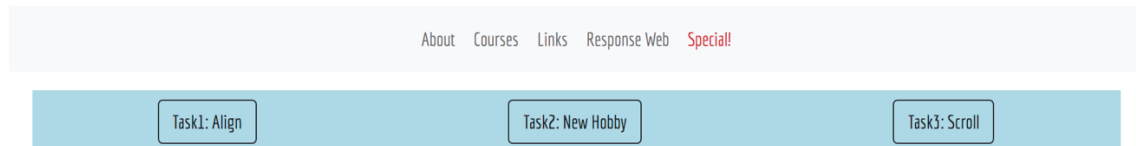♦   Sections with background color

♦   Web font and icons

There should be at least one *visible* example using Bootstrap column layout.

DYNAMIC CONTENTS (30%)

Implement these items using JavaScript in one *external JS file*.

0. ***"Special" link in navbar***

    Include an extra "Special" link to hide or show an extra bar of buttons for 3 Tasks, with their features described below. The extra bar should be hidden by default. There are no other style requirements for this bar. Here is *an example*:
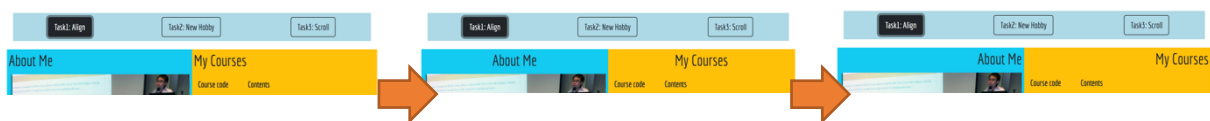

1. ***Task 1: Align***

    When this button is clicked, all the section headings should change the alignment once. Further clicks will change the alignment in this sequence:

    Left >> Center >> Right >> Left >> …

    Illustrated like this,

    Note: the classes in Bootstrap is **.text-start**, **.text-center**, and **.text-end**

2. ***Task 2: New Hobby***

    When this button is clicked, a dialog box (using **prompt()**) asks the user to enter a new hobby. If the user entered something, a new item should be added to the group of "hobbies" in your page. If nothing is entered, no new item should be created. "Hobby" here is only an example. You can use this button to create items for any list/table/flex/… groups in your page. Adjust the text appropriately.

3. ***Task 3: Scroll***

    When this button is clicked, a Bootstrap progress bar (*https://getbootstrap.com/docs/5.0/components/progress*) is shown or hidden below the button bar. If it is shown, when the user scrolls the page, the progress bar adjusts according to the scroll position, i.e. 0% at the top and 100% at the bottom. The progress bar should be applied a **.sticky-top** class, so that it remains visible on the screen when the page is scrolled.

You may find these helpful: `window.scrollY`, `document.body.clientHeight`, `window.innerHeight`

Here is an example:

You may choose to use Bootstrap classes or CSS styles for all styling tasks here.

RESPONSIVE WEB FEATURES (10%)

(Done in Lab 1) You need to incorporate **at least 3 design elements** which would change according to screen size. Visually different layouts must be applied for a mobile phone screen (*width < 500px*) and a large screen (*width >= 500px*). Describe what you have done in a separate section in the page with no fewer than 50 words.

ACCEPTING COMMENTS (10%)

(Done in Lab 3) You could use materials from this file:

[*http://www.cse.cuhk.edu.hk/~chuckjee/2720lab3/index.html*](http://www.cse.cuhk.edu.hk/~chuckjee/2720lab3/index.html)

Add the web form into this page to collect the following information from the user:

♦ Email address

♦ Comment

♦ Color: *Red/Green/Yellow/Blue*

You are recommended to use **input** boxes for the shorter text, a **textarea** for the comment paragraph, and a group of **radio** buttons for the color. You may also decide your own implementation achieving similar user experience.

There should be an **Add comment** button which adds the comment to the bottom of all comments. After the new comment appears, the form should become blank again.

It is okay for your page to come with default comments included, and a comment data file.

FORM VALIDATION (10%)

When the user choose **Add comment**, there could be issues, such as:

♦ Incorrect email format (not x@y.z)   ♦ Empty comment/email

A checking should be done at the beginning when processing the form.

Check `.validity.valid` of the **#new-email** and **#new-comment** boxes . If it is not valid, use `.classList.add("is-invalid")` to the box and **return** (end the function prematurely). Else, use `.classList.remove("is-invalid")` to clear the error highlight.

LOADING AND SAVING COMMENTS (10%)

(*Done in Lab 3*) There should be a **Save file** button to save the comments to a file. This can be done using JS Fetch, with **Web Server for Chrome**. Your comment file can be in text or json.

After saving, when the page is reloaded, all the comments except the original one should disappear. But you should be able to see the contents in the generated file in the web server directory, which it can be loaded using Fetch with a **Load file** button.

AUTO SAVE AND LOAD (10%)

To avoid the user from forgetting to save the comments, you will further simplify the program flow by:

● Loading the comments from the save file when page is loaded, and

● Saving the comments whenever a new comment is added

After this is done, you can remove the **Load** and **Save** buttons. The requirements here override that of the previous part on *Loading and Saving Comments*.

BONUS FEATURE

Besides the information supplied by the user, it is usual for comment systems to keep extra information. Gather these data (*without asking the user*) and show them for each comment:

♦ Browser information (e.g. WebKit)

♦ User's date and time

♦ User's IP address

♦ User's location (e.g. Hong Kong)

♦ Any one more item about the client/server

Only for this part, you are allowed to consult external web services.

> This task is NOT a requirement of the assignment. But if you successfully do it, it contributes to a few bonus points for your course grade:
>
> ♦ *All 5 items = **0.5 points***
>
> ♦ *3 items or more = **0.25 points***

FEATURES AND FRAMEWORKS

You are welcome to implement extra styles and features at your own ability. For Bootstrap, please use the current CDN versions instead of a local copy:

♦ Bootstrap (5.0 beta or later)

`https://cdn.jsdelivr.net/npm/bootstrap@5.0.0-beta1/dist/css/bootstrap.min.css`

Do NOT use any extra plugins or frameworks other than this.

SUBMISSION

We will only visit your web page submission using ***Google Chrome*** (almost-latest versions). Please utilize the ***Developer Tools*** for debugging of your code.

Plagiarism is heavily penalized. Do not attempt sharing code other than those given in labs. Please read this article carefully: *http://www.cuhk.edu.hk/policy/academichonesty*

Include your full name and student ID in ***all code files*** using comments. Zip all your files and submit it on the course site on Blackboard. ***Only these files should be included:***

♦ 1 `.html` file

♦ Optional `.css` file

♦ 1 `.js` file

♦ Image files / comment data file

REFERENCE POINTS

Bootstrap: *https://getbootstrap.com/docs/5.0*