CS2204 Fundamentals of Internet Application Development
Course Work No. 3 (CW3)
Semester A 2023-2024

13% Marks
Due Date: 10 Dec 2023, 23:59

Learning Outcomes:

- Able to write JavaScript and execute them on Web pages
- Able to enhance visual interest, access data, and manipulate elements of Web pages by applying JavaScript to appropriate HMTL and CSS components.

1. Overview

Web pages prepared in CW1 and CW2 will be used. Enhance their interactivities and make all specified features work. JavaScript will be added, and you could also change the HTML/CSS in CW2, if necessary, to facilitate the functioning of JavaScript. Detail requirements are listed in the following sections.

To ensure timely feedback to the student's questions, each of the following TAs will be responsible for different groups of students according to **the last digit of your student ID**. If you have any questions, you can contact the corresponding TA directly.

- Student ID's **last digit 0 ~ 1:** ZENG Yuhan (yhzeng3-c@my.cityu.edu.hk)
- Student ID's **last digit 2 ~ 3:** HONG Chenhao (HONG.Chenhao@my.cityu.edu.hk)
- Student ID's **last digit 4 ~ 5:** Muratov Ali (amuratov4-c@my.cityu.edu.hk)
- Student ID's **last digit 6 ~ 7:** MUKASHEV Alikhan (amukashev2-c@my.cityu.edu.hk)
- Student ID's **last digit 8 ~ 9:** SANIYAZOV Dias (dsaniyazo2-c@my.cityu.edu.hk)

Guideline for late submissions:

- Late within **3 hours - 10% marks deduction**
- Late within **3 ~ 12 hours - 50% marks deduction**
- After **12 hours - no** submission will be accepted

2. External JS Library

An external JavaScript file - https://personal.cs.cityu.edu.hk/~cs2204/2023/cw3cs2204.js must be used in the Visit Page. The function *reserve()* in the library will be used to simulate server action. You are not allowed to use your copy in the final submission.

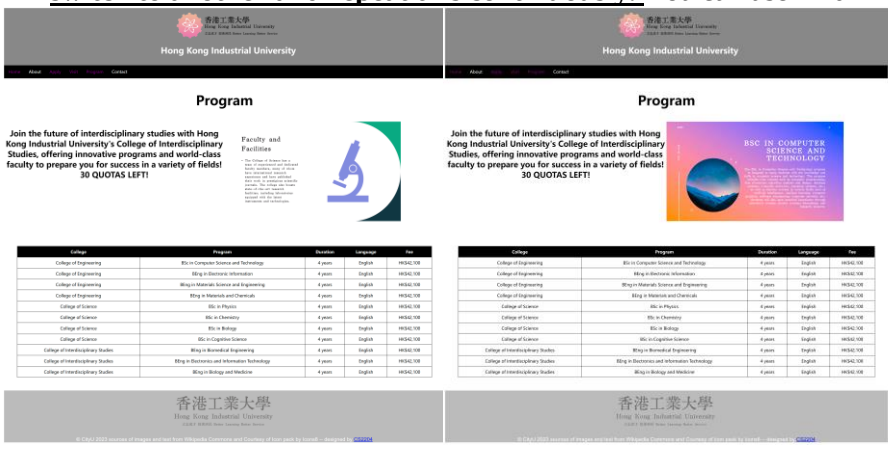
```
function reserve(date, time, no) {
    /*
    *data: the date of reservation
    *time: the time of reservation
    *no: the number of people, type: int
    */
    if (arguments.length != 3) {
        alert("incorrect arguments detected"); // this is not a good way to handle error, this alert should not be activated by your code
        return;
    }
    //an random int as the valid quota
    let valid_quota = Math.floor(Math.random() * 2.0 * no);
    if (no > valid_quota) return false // the quota is full
    else return true; // the quota is not full
}

```

You should note that the *no* parameter should be an integer; **if something wrong happens, you should check whether you pass a non-integer or invalid value to this function.**
Use the HTML and CSS files provided in Canvas to finish the following tasks in this CW.

3. Program Page

- Declare an array to hold three promotional text messages, e.g., "Join Hong Kong Industrial University's College of Science for world-class education and research opportunities in science and technology! 20 QUOTAS LEFT!", "Join the future of engineering with Hong Kong Industrial University's College of Engineering, offering innovative programs and world-class faculty to prepare you for success in the field! 40 QUOTAS LEFT!" and "Join the future of interdisciplinary studies with Hong Kong Industrial University's College of Interdisciplinary Studies, offering innovative programs and world-class faculty to prepare you for success in various fields! 30 QUOTAS LEFT!". On page load, **pick a message at random** and display it in the promotion information block. See the JavaScript Cookbook at the end of the document.

  Keep every message for 3 seconds and switch to the next message. Display three messages in a loop. See the JavaScript Cookbook at the end of the document.
- Use one of the two videos below as a default in your HTML. **At the end of showing one video, switch to another and repeat this continuously.** You can use *innerHTML*.

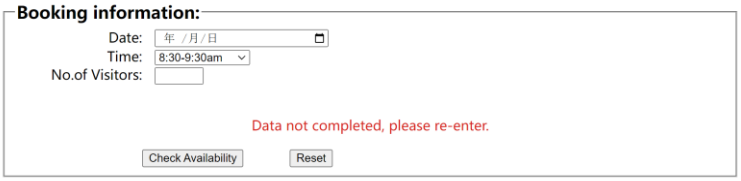


4. Visit Page

- When submitting the Reservation form, check that all fields are not blank (i.e., empty string or all spaces). Display an error message “Data not completed; please re-enter” above the *middle of the box* when an error is found.
- The color of the warning message should be red.
- Do not use *alert()*; see the JavaScript cookbook.

- When **<u>you recheck the availability</u>**, the **<u>previous error messages must be cleared</u>**, and all error messages or alerts are newly displayed.

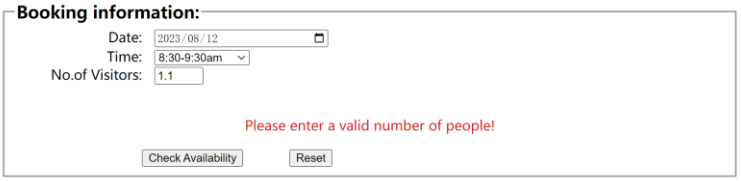
Hong Kong Industrial University is having an open day on campus! Please make a reservation to visit our campus!



- Even if all fields are not blank, **<u>you should also check whether the visitor number is valid, i.e., not smaller than one and is integer.</u>** If it is not valid, you should warn the user, "Please enter a valid number of people!". Do not use *alert()*.

Hong Kong Industrial University is having an open day on campus! Please make a reservation to visit our campus!

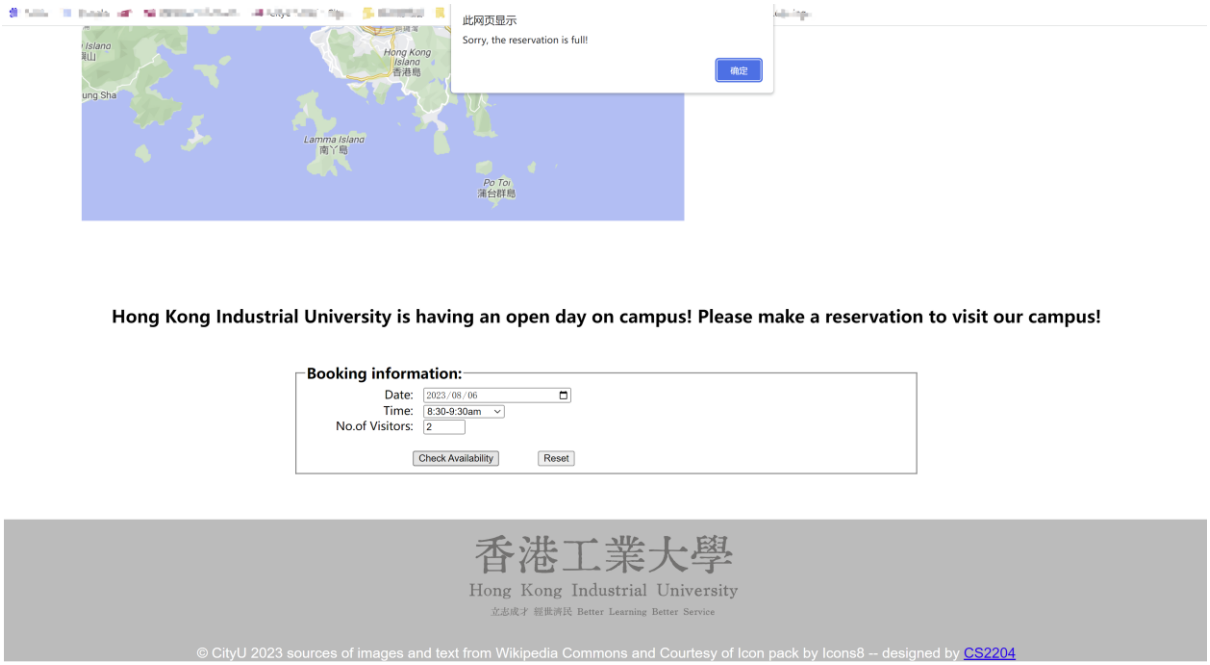
Hong Kong Industrial University is having an open day on campus! Please make a reservation to visit our campus!



- If there is no error in the form data, call the function *reserve* in the library as *reserve(date, time, no-of-visitors)*. Display "Your reservation is successful!" or "Sorry, the reservation is full!" to get a function return of *true* or *false*, respectively. Cancel the form submission in all situations after the message is displayed. You do not need to clear form contents on the page.

Hong Kong Industrial University is having an open day on campus! Please make a reservation to visit our campus!

香港工業大學
Hong Kong Industrial University
立志成才 經世濟民 Better Learning Better Service

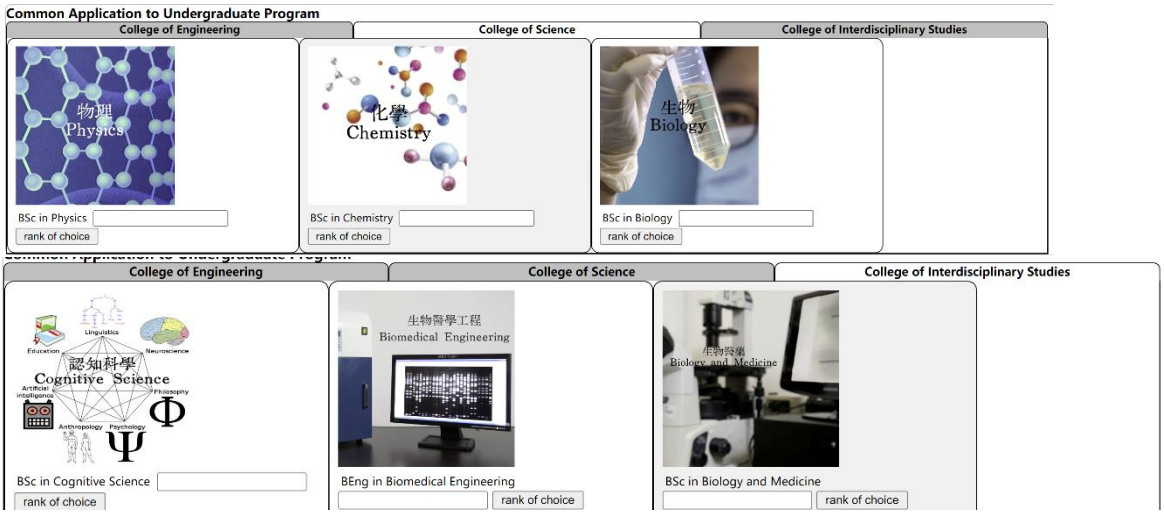


5. Apply Page

Take away the CSS rules implementing hover and show/hide the effect of CW2 for the headings and chosen majors.

- **After page loading, show the *College of Engineering* heading and major application forms and set the background color of the College of Engineering heading to white and the other two headings with another color of your choice**

- Redo show/hide effect in CW2 by using the ***on-click* event** for the three headings, i.e**., clicking any header will show the corresponding major application forms and hide all other major application forms. The background color of the clicked header will change to white, and the other two will change to your selected color.**



With JavaScript or otherwise (see JavaScript Cookbook), set up **event handlers** for all "rank of choice" buttons. You should assume the number of buttons/chosen majors is only known after the page loads. Write codes for the event handler to update the result table when clicking a button. You may modify the HTML and CSS to achieve all these things.

At first, the "Your chosen majors" table is empty. Only the ranks show. At the beginning, in the "Your chosen majors" table, there should be a rank from 1-10 in the rank column:

Your chosen majors:

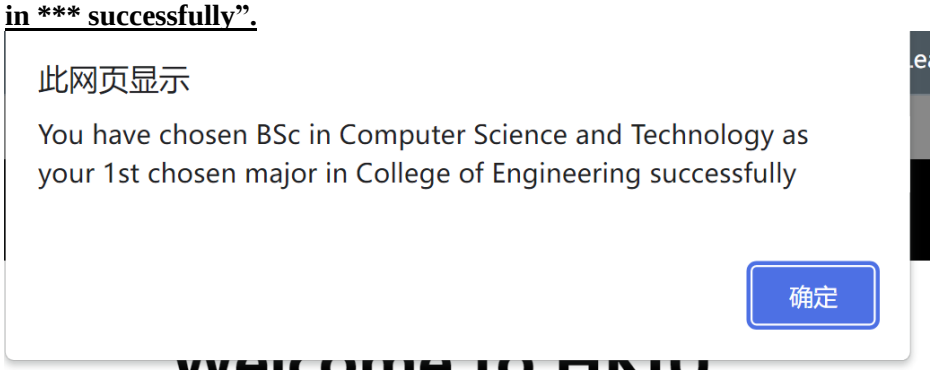
Last change time: 2023/8/27 01:00:21

College	Major	Rank
		1
		2
		3
		4
		5
		6
		7
		8
		9
		10

Total Number of Majors Applied: 0

submit clear

After you click the "rank of choice" button, a new major will be added to the table according to the rank. **Note that you need special treatment for some ordinal words, such as first, second, and third. The alert message should be "You have chosen ****** as your *** chosen major in *** successfully".**



Your chosen majors:

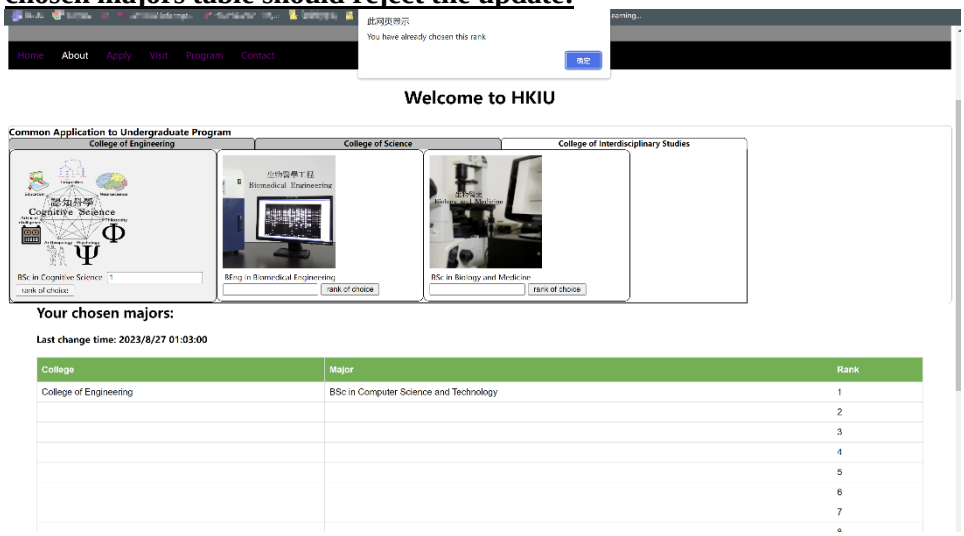
Last change time: 2023/8/27 01:03:00

College	Major	Rank
College of Engineering	BSc in Computer Science and Technology	1
		2
		3
		4
		5
		6
		7
		8
		9
		10

Total Number of Majors Applied: 1

submit clear

<u>If a duplicate in rank happens, it will show, "You have already chosen this rank". Your chosen majors table should reject the update:</u>

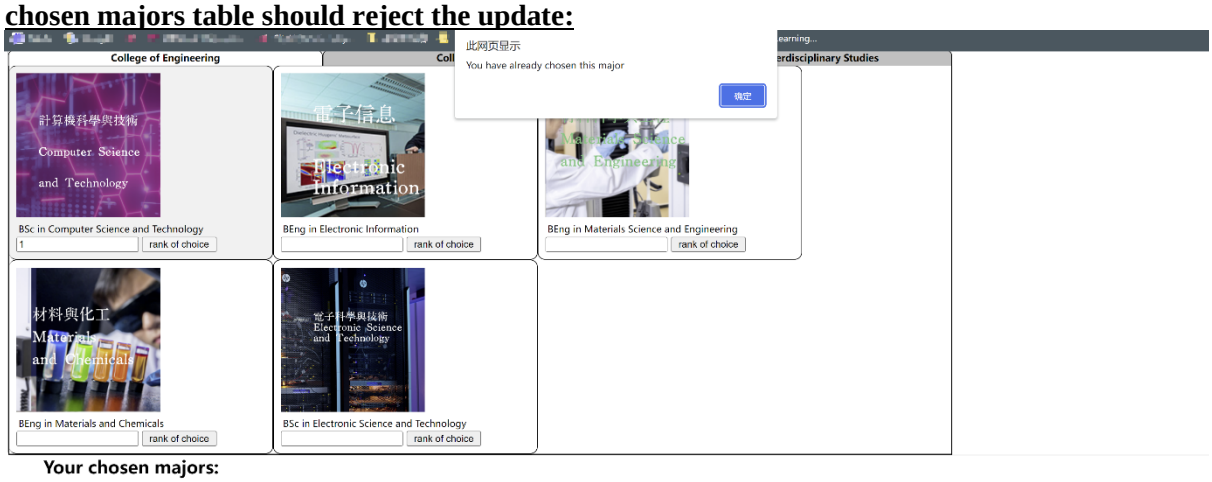


Your chosen majors:

Last change time: 2023/8/27 01:03:00

College	Major	Rank
College of Engineering	BSc in Computer Science and Technology	1
		2
		3
		4
		5
		6
		7

<u>If a duplicate in major happens, it will show, "You have already chosen this major". Your chosen majors table should reject the update:</u>



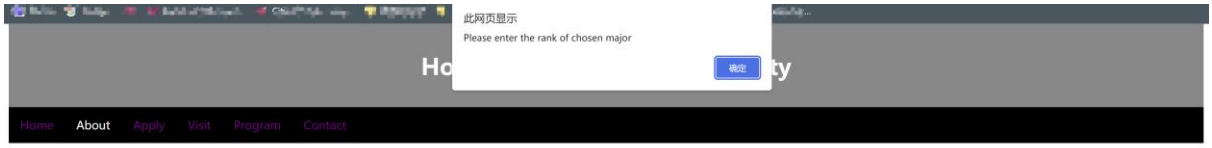
Your chosen majors:

Last change time: 2023/8/27 01:03:00

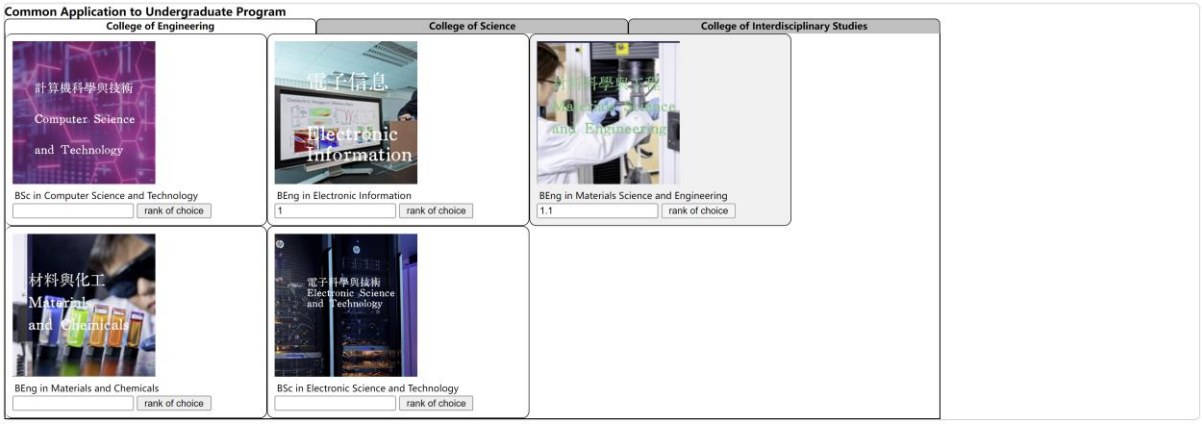
College	Major	Rank
College of Engineering	BSc in Computer Science and Technology	1
		2
		3
		4
		5

<u>Let's be consistent: if we add a new major with duplicate majors and duplicate ranks, we only need to alert "You have already chosen this major," as shown in the screenshot above.</u>

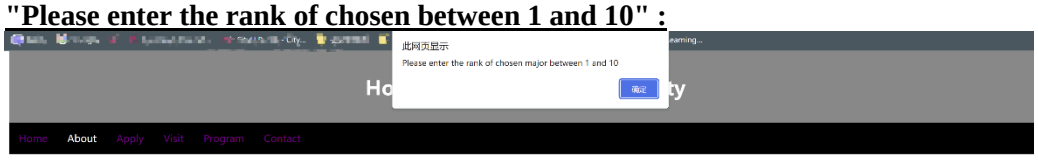
Also, **if given an empty rank or the type of rank is not an integer, it will alert "Please enter the rank of chosen major":**



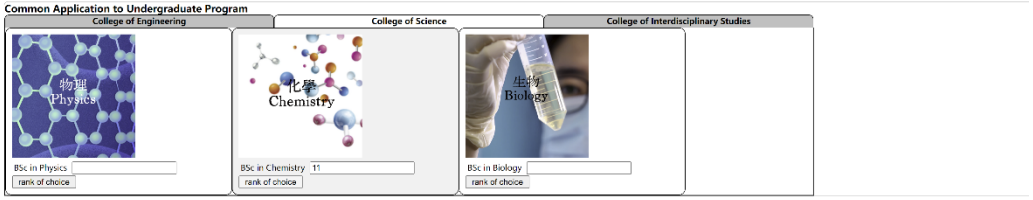
Welcome to HKIU

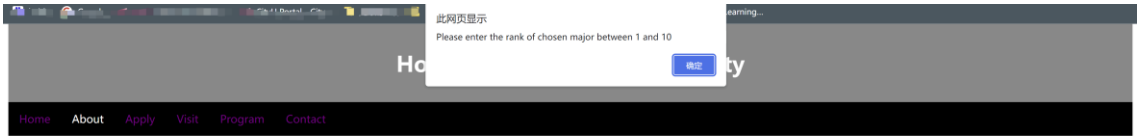


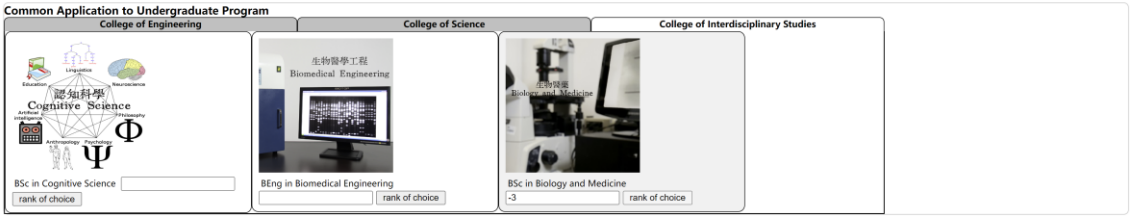
If the rank of chosen is an invalid integer, which is too big or too small, you should show "Please enter the rank of chosen between 1 and 10" :



Welcome to HKIU







In general, the pseudo-code for checking the rank and major should be:

```
let rank_input: string = getRank();//get rank from user input, at first the rank may be a string
//check if rank is a int
if (isIntegerInput(rank_input) === false) {
    alert("Please enter the rank of chosen major");
    //return;
}
let rank: number = parseInt(rank_input); //convert rank to int
//check if rank a int between 1 and 10
if (rank < 1 || rank > 10) {
    alert("Please enter the rank of chosen major between 1 and 10");
    //return;
}
for (let i: number = 1; i <= 10; i++) {//go through all the majors
    if (checkDuplicateMajor() === true) { //check if the major is already chosen
        alert("You have already chosen this major");
        //return;
    }
    if (checkDuplicateRank() === true) {
        alert("You have already chosen this rank");
        //return;
    }
}
```

Hint: We can implement a JavaScript function. We first get the information about the chosen major and check the rank: check if the rank is an integer input and, if so, convert the rank to an integer from string, using the parseInt() function to do the conversion. If the rank is empty, null, undefined, or not an integer input, a warning alerts the user to reenter a rank. If the rank is less than one or greater than 10, a warning prompts the user to enter a rank between 1 and 10. Then, we check for duplicates of the major and rank of choice. Major: iterates through the numbers from 1 to 10 in a loop, checking to see if the major is the same as the one already in the list, and if there is, an alert pops up that the user has selected that major. Then, check if the user has selected the rank already. Then, determine the college affiliation, display a success message, and update the table.

- Write a function *updateTable()* to calculate the Total Number of Majors Applied and update the Last change time. The table should always be updated if you add a chosen major successfully.

Your chosen majors:

Last change time: 2023/8/27 01:51:42

College	Major	Rank
College of Engineering	BEng in Electronic Information	1
College of Interdisciplinary Studies	BSc in Cognitive Science	2
		3
		4
		5
		6
		7
		8
		9
		10

Total Number of Majors Applied: 2

submit clear

- Set up and write the *onclick* event handler for the *submit* link.
- This link is used to submit the user's chosen major. It first checks for errors; if there are any, it displays the appropriate error message.
- **<u>If you submit an empty table, you should display the following: "You have not chosen any chosen." Note that you should not use *alert():*</u>**

Your chosen majors:

Last change time: 2023/8/27 01:52:13

College	Major	Rank
		1
		2
		3
		4
		5
		6
		7
		8
		9
		10

Total Number of Majors Applied: 0

submit clear

You have not chosen any major

- Check if there **<u>exists a gap</u>** in your table when you submit:
- This is a gap:

Your chosen majors:

Last change time: 2023/8/27 01:53:12

College	Major	Rank
College of Engineering	BSc in Computer Science and Technology	1
		2
College of Engineering	BEng in Electronic Information	3
College of Engineering	BEng in Materials Science and Engineering	4
College of Engineering	BEng in Materials and Chemicals	5
College of Engineering	BSc in Electronic Science and Technology	6
		7
		8
		9
		10

Total Number of Majors Applied: 5

submit clear

You have not chosen your 2nd chosen major, you can not leave any gap between your majors

- This is a gap:

Your chosen majors:

Last change time: 2023/8/27 01:54:53

College	Major	Rank
		1
College of Engineering	BSc in Computer Science and Technology	2
College of Engineering	BEng in Electronic Information	3
College of Engineering	BEng in Materials Science and Engineering	4
College of Engineering	BEng in Materials and Chemicals	5
College of Engineering	BSc in Electronic Science and Technology	6
College of Science	BSc in Physics	7
College of Science	BSc in Chemistry	8
		9
College of Science	BSc in Biology	10

Total Number of Majors Applied: 8

submit clear

You have not chosen your 1st chosen major, and 9th chosen major, you can not leave any gap between your majors

- This is a gap:

Last change time: 2023/8/27 01:56:16

College	Major	Rank
		1
College of Engineering	BEng in Electronic Information	2
College of Engineering	BSc in Computer Science and Technology	3
College of Interdisciplinary Studies	BSc in Biology and Medicine	4
College of Interdisciplinary Studies	BEng in Biomedical Engineering	5
		6
		7
		8
		9
		10

Total Number of Majors Applied: 4

submit clear

You have not chosen your 1st chosen major, you can not leave any gap between your majors

- This is not a gap:

Your chosen majors:

Last change time: 2023/8/27 01:56:54

College	Major	Rank
College of Science	BSc in Physics	1
College of Science	BSc in Chemistry	2
College of Science	BSc in Biology	3
		4
		5
		6
		7
		8
		9
		10

Total Number of Majors Applied: 3

submit clear

You have successfully submitted your application at time 2023/8/27 01:56:57

- **<u>If you submit the table with gaps, you should display it this way: "You have not chosen your " + gap chosen majors+ ", you can not leave any gap between your chosen majors". Note that you should not use *alert():*</u>**
- **<u>If the form has no problems, the successful submission message and the current time are displayed. You should display “You have successfully submitted your application at time …”.</u>**

Your chosen majors:

Last change time: 2023/8/27 01:58:29

College	Major	Rank
College of Science	BSc in Physics	1
College of Science	BSc in Chemistry	2
College of Science	BSc in Biology	3
College of Engineering	BSc in Computer Science and Technology	4
College of Engineering	BEng in Electronic Information	5
College of Engineering	BEng in Materials Science and Engineering	6
College of Engineering	BEng in Materials and Chemicals	7
College of Engineering	BSc in Electronic Science and Technology	8
		9
		10

Total Number of Majors Applied: 8

submit clear

You have successfully submitted your application at time 2023/8/27 01:56:57

- Set up and write the *onclick* event handler for the *clear* link.
- **<u>This link is used to empty the chosen major. It empties the content for each rank. It then updates the table with the last change time and the Total Number of Majors Applied (i.e., 0).</u>**

Your chosen majors:

Last change time: 2023/8/27 01:58:29

College	Major	Rank
College of Science	BSc in Physics	1
College of Science	BSc in Chemistry	2
College of Science	BSc in Biology	3
College of Engineering	BSc in Computer Science and Technology	4
College of Engineering	BEng in Electronic Information	5
College of Engineering	BEng in Materials Science and Engineering	6
College of Engineering	BEng in Materials and Chemicals	7
College of Engineering	BSc in Electronic Science and Technology	8
		9
		10

Total Number of Majors Applied: 8

submit clear Click here

You have successfully submitted your application at time 2023/8/27 01:56:57

Your chosen majors:

Last change time: 2023/8/27 01:59:51

College	Major	Rank
		1
		2
		3
		4
		5
		6
		7
		8
		9
		10

Total Number of Majors Applied: 0

submit clear

6. Assessment

You will be assessed by how much and how well you can apply what has been learned from the course; some considerations are:

- The requirements are met.
- Two ways of event handler setup (by JavaScript and in HTML. See the JavaScript Cookbook; must be used at least once. ***State in the Design page where the two techniques are used (i.e., identify the JavaScript line no. at which the event handler is set up with what kind of method)***
- **Use of other JavaScript libraries or frameworks, such as jQuery, Angular or React.js, etc., is NOT allowed;**
- Sensible use of external, embedded, or inline scripts.
- Show a clear separation of structure, style, and script.
- Use appropriate HTML tags, classes, ids, or CSS to help your JavaScript.
- Arrange your website directories properly, e.g., HMTL, style sheets, images, scripts, etc.
- CSS effect in CW2 will not be rechecked, but the error should not be seen in HTML validation.
- In June 2015, ECMAScript 6 was officially adopted as an international standard. Its goal is to make the JavaScript language usable for writing complex, large-scale applications and to become an enterprise-level development language. Currently, browser support for ES6 can be viewed at kangax.github.io/compat-table/es6/. To encourage students to engage with the latest advances in JavaScript, this assignment will give bonuses for using the following ES6 features. ***Indicate on the Design page where ES6 features are used and which features are used. (i.e., indicate the JavaScript line number and explain the use of the feature to convince the Teaching Assistant that you are not just copying sample code and using it for the sake of using it).***
  (1) *let* and *const*
  (2) *set* and *map*
  (3) *class*

7. Submission

- Submit a zip file of your website with appropriate folders set up so that it can be used directly by unzipping

8. Useful techniques to complete the coursework(JavaScript Cookbook)

Multiple ways to write JavaScript for a problem may exist, with no single "correct" answer. It is essential to first think of a way to solve the problem before writing your codes.

i) Some common techniques include but are not limited to using variables, Array/JSON to store information or DOM objects, selecting DOM objects, using loops, etc.

ii) Get hold of elements by CSS selectors:

```
x=document.querySelector('div p')
//'div p' is a string containing any valid CSS selector
//x points to the first <p> no matter how many p are selected

x=document.querySelectorAll('div p')
//x becomes an array even if one or no p is selected
```

iii) Loop to scan through selected tags and change properties:

```
x=document.querySelectorAll('div p');
for (let i=0; i<x.length; i++) {
  x[i].style.color="red";
}
//Set the color of selected p by a loop
```

iv) Change elements' properties, especially styles or innerHTML; use document.write( ) method at the position where you want to create HTML in <body>

v) Show or hide elements dynamically

vi) Set up event handler

vii) Random number generation: use methods Math.random() and Math.floor(), you can refer to https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Math/random and https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Math/floor .

viii) Running JS functions continuously:

setInterval('myfunc()', 3000) // run myfunc() every 3 seconds

you can refer to https://developer.mozilla.org/en-US/docs/Web/API/setInterval.

ix) The *let* declaration declares re-assignable, block-scoped local variables, initializing each to a value optionally, while *const* declares a block-scoped, read-only named constant. You can refer to https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Grammar_and_types.

x) For the set, you can refer to https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Set , and for the map, you can refer to https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Map .

xi) ES6 provides a closer approximation to traditional languages by introducing the concept of class. Classes can be defined using the class keyword. ES6's class can be thought of as just syntactic sugar. The new way of writing classes makes it clearer to write object prototypes in a way that is more like the syntax of object-oriented programming. You can refer to https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Classes.

```
class Person {
    constructor(name, age) {
        this.name = name;
        this.age = age;
    }
    sayHello() {
        console.log("Hello");
    }
}
class Student extends Person {
    constructor(name, age, course) {
        super(name, age);
        this.course = course;
    }
    sayHello() {
        super.sayHello();
        console.log("I am a student and I am studying " + this.course);
    }
}
```

xii) We strongly recommend you read MDN to better understand JavaScript: https://developer.mozilla.org/en-US/docs/Web/JavaScript.

~ End ~