COMP3322 MODERN TECHNOLOGIES ON WORLD WIDE WEB

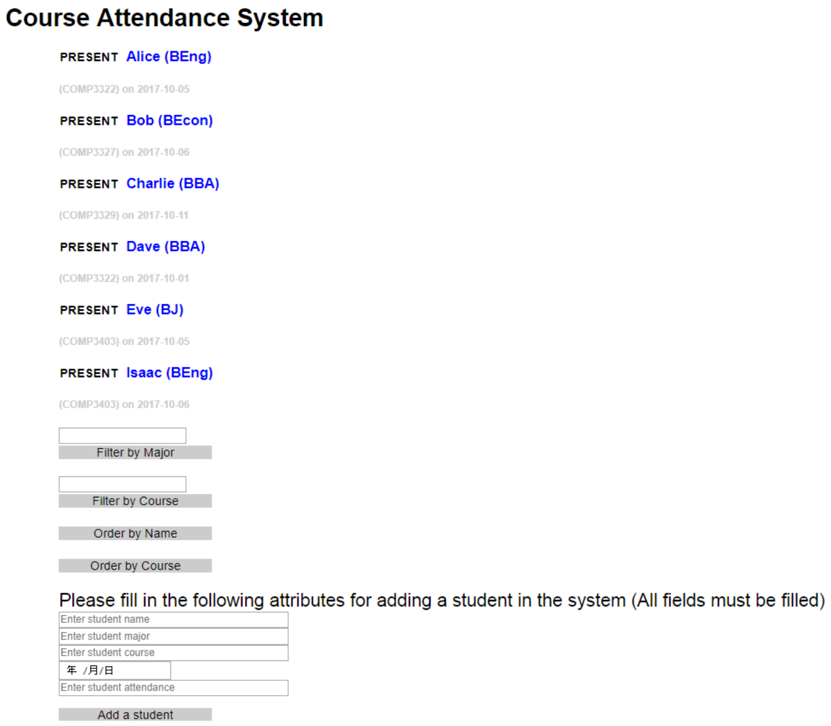
Workshop 3 jQuery

Overview

Note: Please download the template files from the Workshop 3 folder instead of reusing your own code in Workshop 2.

In this workshop, we will implement a web-based course attendance system, which is similar to the one in Workshop 2, but using **jQuery**, plus adding a few additional functionalities. Especially, except the functionalities as implemented in Workshop 2, we add two buttons "Order by Name (ascending)" and "Order by Course (ascending)" on the page. When each button is clicked, the attendance records displayed will be ordered alphabetically by student name or course code in ascending order respectively. You should make sure that the ordering can be successfully carried out no matter it is a full list or filtered list (filtered by Major/Filtered by Course). Some example screenshots are shown below.

The example screenshot of the page after initial loading is given in Fig. 1.



Course Attendance System

PRESENT Alice (BEng)

(COMP3322) on 2017-10-05

PRESENT Bob (BEcon)

(COMP3327) on 2017-10-06

PRESENT Charlie (BBA)

(COMP3329) on 2017-10-11

PRESENT Dave (BBA)

(COMP3322) on 2017-10-01

PRESENT Eve (BJ)

(COMP3403) on 2017-10-05

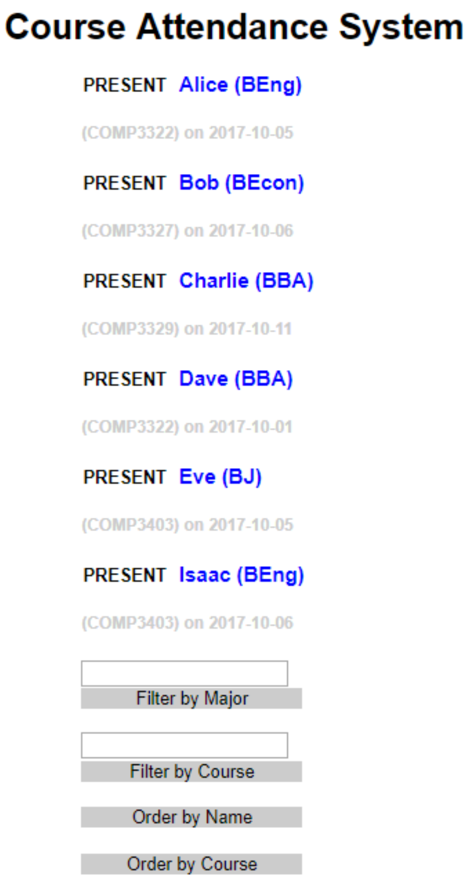
PRESENT Isaac (BEng)

(COMP3403) on 2017-10-06

Please fill in the following attributes for adding a student in the system (All fields must be filled)

Fig. 1

Example screenshots of the page after clicking the "Order by Name" button and after clicking the "Order by Course" button, are given in Fig. 2 and Fig. 3, respectively. Note that the displayed entries, no matter the full list or filtered list (as shown in Fig.3 and Fig.4), will be ordered accordingly when one of the two buttons are clicked.



Course Attendance System

PRESENT Alice (BEng)

(COMP3322) on 2017-10-05

PRESENT Bob (BEcon)

(COMP3327) on 2017-10-06

PRESENT Charlie (BBA)

(COMP3329) on 2017-10-11

PRESENT Dave (BBA)

(COMP3322) on 2017-10-01

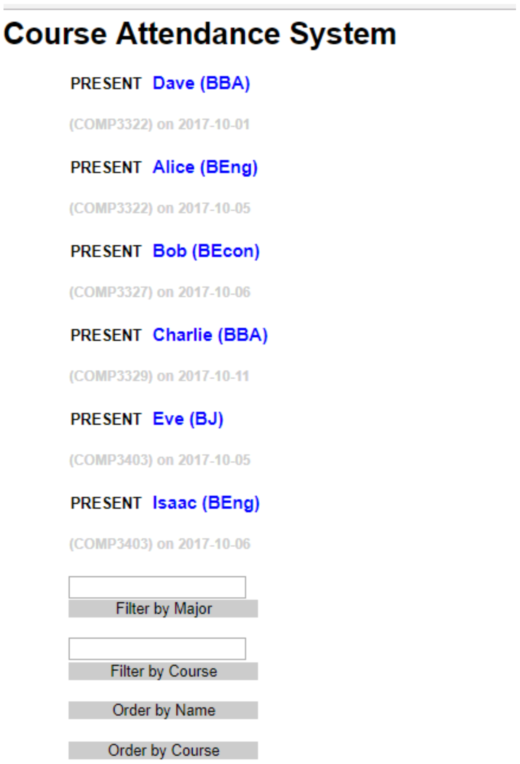
PRESENT Eve (BJ)

(COMP3403) on 2017-10-05

PRESENT Isaac (BEng)

(COMP3403) on 2017-10-06

Fig. 2 Order by Name (ascending, full list)



Course Attendance System

PRESENT Dave (BBA)

(COMP3322) on 2017-10-01

PRESENT Alice (BEng)

(COMP3322) on 2017-10-05

PRESENT Bob (BEcon)

(COMP3327) on 2017-10-06

PRESENT Charlie (BBA)

(COMP3329) on 2017-10-11

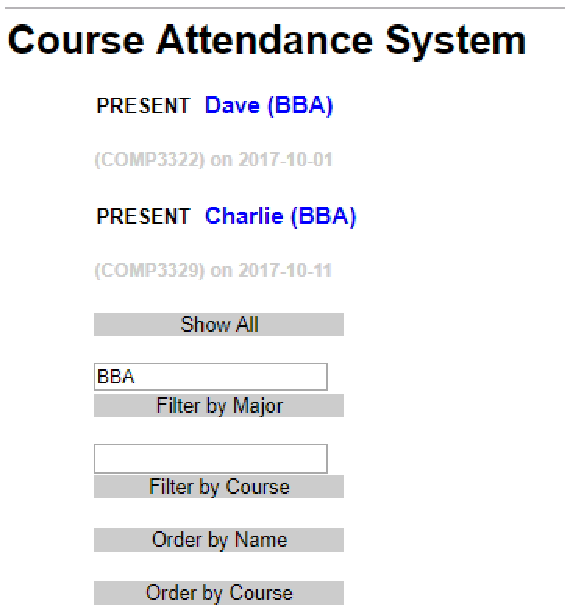
PRESENT Eve (BJ)

(COMP3403) on 2017-10-05

PRESENT Isaac (BEng)

(COMP3403) on 2017-10-06

Fig. 3 Order by Course (ascending, full list)



Course Attendance System

PRESENT Dave (BBA)

(COMP3322) on 2017-10-01

PRESENT Charlie (BBA)

(COMP3329) on 2017-10-11

BBA

Fig. 4 Order by Course (ascending, filtered list, filter by BBA)

Task 1 : Reimplement Workshop 2 functionalities using jQuery

Step1: We will use the same database attendanceList which you created and used in Workshop 2. Make sure there are at least 6 records in the database table.

Step2: Download **Workshop3_2020.zip** and you will find all 6 files we need in this workshop.

Step3: In **index_WS3.html**, load the jQuery library by adding the following line (jquery-3.3.1.js is included in the zip file):

```
<script src=" jquery-3.3.1.js"></script>
```

In addition, add code in **index_WS3.html** to link to **script_WS3.js**, in which you will implement all the jQuery code.

Step4: Change the database connection setting in **queryEntries_WS3.php** and **updateState_WS3.php** to the correct setting for connecting to your MySQL account at sophia.cs.hku.hk.

```
  define("DB_HOST", "sophia.cs.hku.hk");
  define("USERNAME", "tmchan");
  define("PASSWORD", "abcd1234");
  define("DB_NAME", "tmchan");
```

Step5: In **script_WS3.js**, we have provided the following code:

```
$(document).ready(function () {
    showAll();
    $("#button_all").click(showAll);
    $("#fM").click(function () {
    });
    $("#fC").click(function () {
    });
    $("#add").click(function () {
    });
    $("#orderByName").click(function () { //order by name
    });
    $("#orderByCourse").click(function () { //order by course
    });
});
function showAll() {
}
function changeState(elem) {
    var itemID = $(elem).parent().attr("id");
    if ($(elem).html() === 'PRESENT') {
        newvalue = 'ABSENT';
    } else {
        newvalue = 'PRESENT';
    }
}
```

(1) Implement the function **showAll()** such that all entries stored in attendanceList table in the MySQL database would be displayed by using jQuery function $.get (to issue AJAX request for **queryEntries_WS3.php**) . It is also expected that after showing all the entries, the "Show All" button would <u>not be seen</u> from the web page (using jQuery method).

(2) Complete the callback function in $("#fM").click(function () { }); and $("#fC").click(function () { }); such that when the user clicks the Filter by Major/Filter by Course button, the displayed entries would be filtered according to the text entered in the textbox above Filter by Major/Filter by Course button, by using jQuery function $.get (to issue AJAX request for **queryEntries_WS3.php**). It is also expected that after the filtered list is displayed, the "Show All" button would be <u>displayed</u> on the web page (Fig. 4).

(3) Implement the function **changeState()** using jQuery method load() to issue an HTTP POST AJAX request for **updateState_WS3.php** such that when the user clicks the attendance status (i.e. the PRESENT/ABSENT text before the student's name) of a student, it will toggle (i.e. from PRESENT to ABSENT/ABSENT to PRESENT).

```
$(elem).load("updateState_WS3.php",{id: itemID, newValue: newvalue});
```

(4) Implement the callback function in $("#add").click(function () { }); such that the data entered in the addListForm in index_WS3.html would be added to the attendnaceList table in the MySQL database and all the students' attendance record stored in the attendanceList table would be displayed after the insertion of a new student (Use jQuery function $.get to issue AJAX request for **queryEntries_WS3.php**). The AJAX request would only be initiated if the following conditions are satisfied:

1. All the required information is filled (including student name, student major, student course, student course date and student attendance)
2. The entered student's attendance is either "PRESENT" or "ABSENT" (case-sensitive)

Otherwise an alert box displaying the message "Check your input Attendance record (PRESENT/ABSENT) and whether all fields are filled." will be shown.

Step 5: Clear the data entered by the user in the following situations:

1. After successful insertion of a new student's record, the data entered by the user in the addListForm should be cleared.
2. After successful filtering (filter by major/filter by course), the filtered value entered by the user in the textbox above the button "Filter by Major"/"Filter by Course" should be cleared.

Task 2: Order the displayed entries

In **script_WS3.js**, we also have two event handlers for "click" on the two additional buttons for ordering the entries shown on the page: $("#orderByName").click(function () { }); and $("#orderByCourse").click(function () { });.

Step 1. Implement the event handling function in $("#orderByName").click(function () { }); as follows:

```
$entrydivs=$("#entries").children();
$entrydivs.sort(function(a,b){
  var an = $($(a).find('h3')[0]).text();
  var bn = $($(b).find('h3')[0]).text();
  if(an > bn) {
    return 1;
  }
  if(an < bn) {
    return -1;
  }
  return 0;
});
$entrydivs.detach().appendTo($("#entries"));
```

Understand the above code for sorting the set of child <div> elements in the division with id “entries”, by referring to the following references:

- JavaScript sort() method https://www.w3schools.com/jsref/jsref_sort.asp
- jQuery .detach() method https://api.jquery.com/detach/
- jQuery .appendTo() method http://api.jquery.com/appendto/

Step 2. Implement the event handling function in $("#orderByCourse").click(function () { }); by learning from the code in Step 1.

Submission

Upload all your files: index_WS3.html, script_WS3.js, style.css, queryEntries_WS3.php, updateState_WS3.php, and jquery-3.3.1.js to a preset directory under the course account (c3322b). We have created a folder for each student under the path for Workshop3: /course/home/c3322b/public_html/Workshop3/ For example, a student with the CS account username - tmchan, would have a directory: /course/home/c3322b/public_html/Workshop3/tmchan/

Please finish this workshop exercise before 23:59 April 8, 2020 (Wednesday). We will go to your individual webpage at http://i7.cs.hku.hk/~c3322b/Workshop3/[your_CSID]/index_WS3.html to check your page. If the page is correctly implemented, you will get 2 points for this workshop exercise.