UNIVERSITY MALAYSIA TERENGGANU
FACULTY OF COMPUTER SCIENCE & MATHEMATICS

FRONT-END PROGRAMMING
CSF 3103

LAB REPORT 5:
JQuery

Prepared by:
NUR ARIFAH BINTI MOHD HANAFIAH
S66428

Prepared for:
DR RABIEI BIN MAMAT

BACHELOR OF COMPUTER SCIENCE (MOBILE COMPUTING) WITH HONORS
SEMESTER II 2023/2024

Lab 5

Task 1

a. Get the code from https://frend.odaje.biz/lab5/task1.html, and evaluate and test the code.

    i. What is the id of button and input field
    For the id button is "button_change" and for the id input field is "input_color".

    ii. What happened when you clicked change button and why?
    When I click the button change is nothing happen and it is because of Javascript code inside script tag is empty or not in full code .

b. Your mission: when the button is clicked, all occurrences of the word Black to be changed to the text in the input field.

    a. Select the change button

```
var button = $('#button_change');
```

    b. Setting a click listener for button

```
$(button).click(function(){
    //your next code is here
});
```

    c. Retrieving and storing the input value

```
//select the input color
var input_color = $('#input_color');

//select the input value
var input_value = $(input_color).val();
```

    d. Select all appropriates black text by class selector

```
//select all appropriate black text recognise by class
var element = $('.door_color');
```
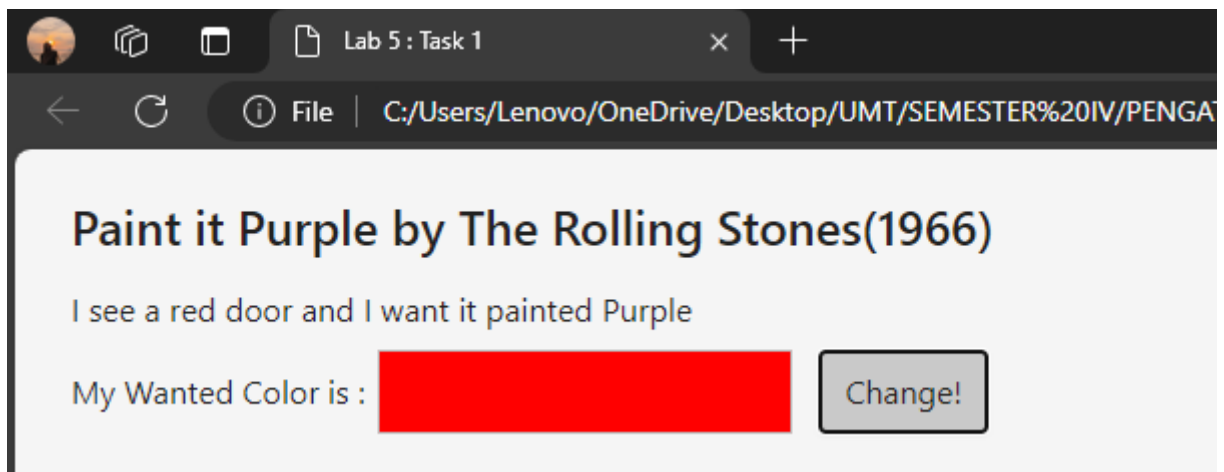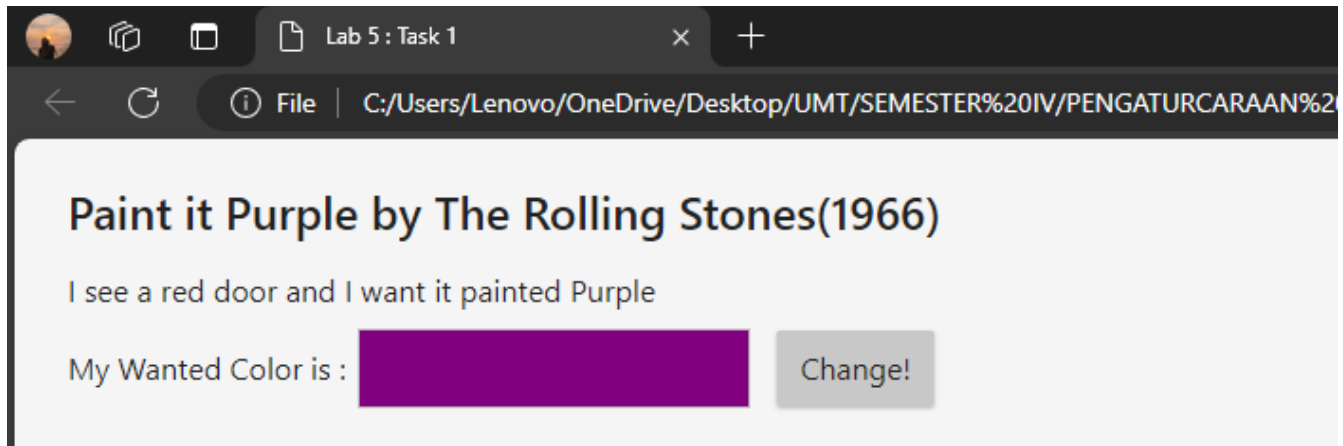
    e. Replace the text with the input value

```
//changed the selected text color to inpiut value color
$(element).html(input_value);
```

    f. Test and evaluate your code. Comment on the new behaviour of your code

    g. Basically, all those id and value in the code can be access directly without using any variables. **Rewrite the code by directly using the jQuery selector**

    h. Rewrite the code: can refer to task 2 if in doubt
        i. Instead of input field, change it to dropdown which display available color understood by HTML (use select tag)
        ii. Retrieve the value from the select tag

iii.   **Change the color of all appropriate text to selected color**

The Output :

Task 2

a. Get the code from https://frend.odaje.biz/lab5/task2.html, and evaluate and test the code
   i. What id's that exist in the code above?
   ii. Open the file inside browser, what have you see in the browser?
   iii. Click any button, what happened? Why

b. Your mission:
   i. Click on Toggle Colors button: Toggles the color between white and cardinal
   ii. Click on Toggle Rounded Borders: Toggles the appearance of a border on the boxes
   iii. Click add a new box: adds one new box to the container

c. Implements the following to accomplished your mission
   i. Add outlined class to each <div>. Refer to <div> for box2 as an example. Save and open the file in browser
        i. What changes have you see?
   ii. Add a click function to button_toggle_color, and implementing change color process

```
//add button_toggle_color click function
$('#button_toggle_colors').click(function(){
  //get box1
  var box1 = $('#box1');
  //check and changed background color of box1
  if($(box1).css('background-color')==colorTeal){
    //remove the color
    $(box1).css('background-color','');
  }else{
    //put back colorTeal as background
    $(box1).css('background-color',colorTeal);
  }
});
```

   iii. Create an event listener for button_toggle_roundedges
   iv. Inside the function add the following code

```
$(".box").each(function(){
  if ($(this).hasClass('round-edge')){
    $(this).removeClass('round-edge');
  }else{
    $(this).addClass('round-edge');
  }
});
```

   v. Based on code in iv, can you modify a code in ii, so it change all box background and returned all to vanilla as well!

vi. Create an event listener for button_add_box.

vii. Inside the function put the following code

    i. Create new element using javascript

```javascript
var new_box = document.createElement('div');
```

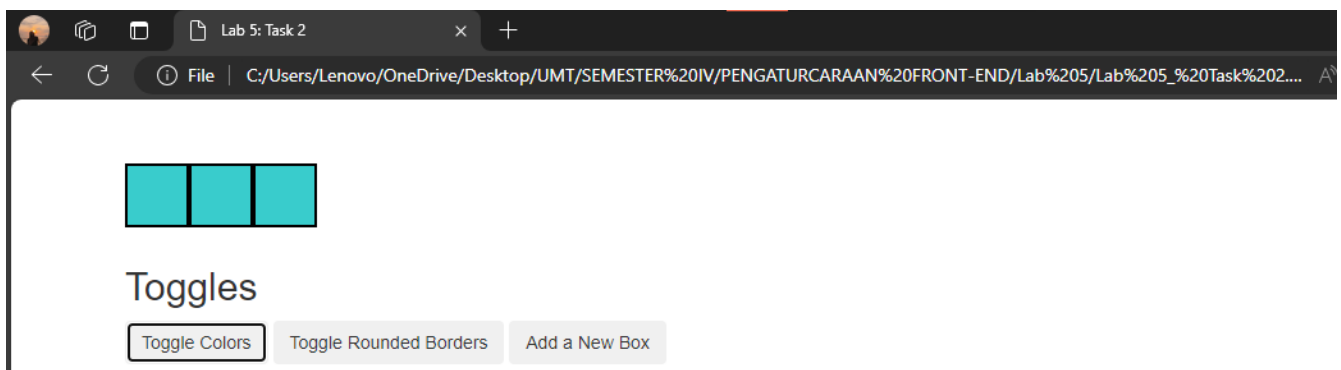   ii. Get current number of element with class .box

```javascript
var existingBoxes = $(".box").length;
```

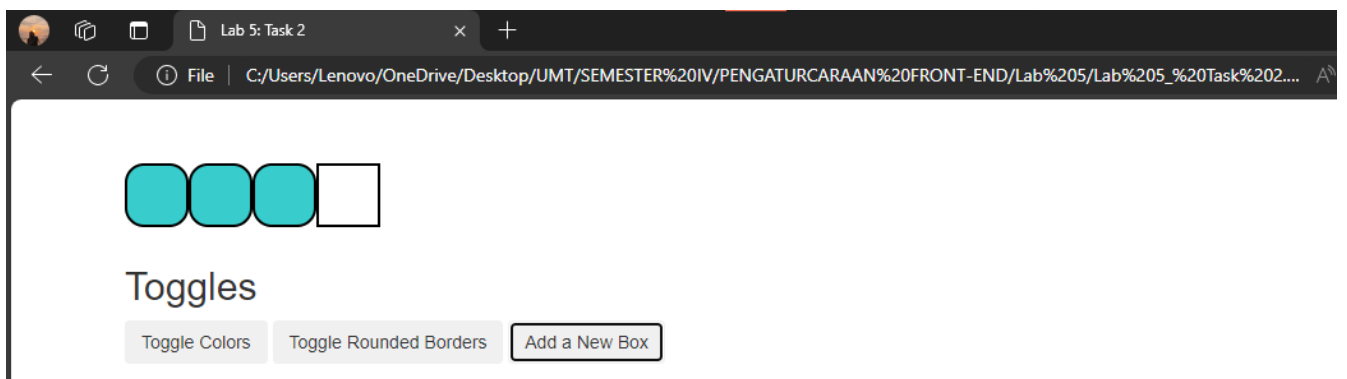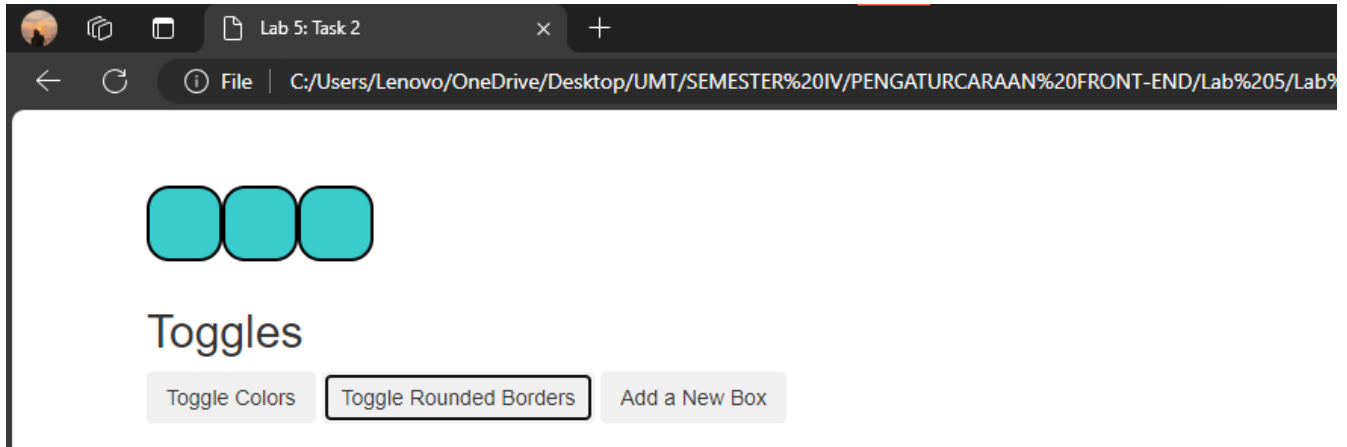   iii. With the current number of box create new id for new box

```javascript
var nextBoxId = existingBoxes+1;
$(new_box).attr("id","box"+nextBoxId);
```

    iv. Add two classes (box and outlined) to new_box

```javascript
$(new_box).addClass("box");
$(new_box).addClass("outlined");
```

    v. Finally, attached new box into box container

```javascript
$("#boxes").append(new_box);
```

viii. Test and evaluate your code

ix. Modify the code above so that when button clicked, it only show one element which selected randomly .

The Output :

Task 3

a. Get the code https://frend.odaje.biz/lab5/task3.html, and evaluate and test the
   code:
   i.   Determine the id's of each element
   ii. Recognise any input element that associated with the form
   iii. Your Task are:
        1. Add DatePicker
        2. Use selectable interaction
        3. Use Sortable widget
        4. Use the tooltip

b. Inserting the datepicker into a form
   i.   In the document ready function add this code

```
$('#date').datepicker();
```

        i. Can you set the date to limit the selectable date range to only
           include dates from the previous two weeks to the next two weeks.

c.   Using selectable for enrolment type
   i.   Comment radiobutton input
   ii. Add the following code

```
<ol id="selectable">
    <li class="ui-widget-content" title="taking 6.813">Undergraduate</li>
    <li class="ui-widget-content" title="taking 6.831">Graduate</li>
</ol>
```

   iii. Add some css in style section

```css
#selectable .ui-selecting {
  background: #FECA40; }

#selectable .ui-selected {
  background: #F39814; color: white; }

#selectable li:before { content: "\2713 "; color: transparent; }

#selectable li.ui-selecting:before { content: "\2713 "; color: black; }

#selectable li.ui-selected:before { content: "\2713 "; color: white; }

#selectable { list-style-type: none; margin: 0; padding: 0;
  width: 150px; cursor: pointer; }

#selectable li {
  margin: 3px; padding: 0.4em;
  font-size: 1.0em; height: 16px; }
```

    iv.   And control the behaviour using jQuery selector

```javascript
$("#selectable").selectable();
```

    v.   How can we retrieve the value of the selected item?

d. Using sortable for the color input
    i.   Comment all input element that related to color selections
    ii.   Replace with the following code

```html
<ul id="sortable">
  <li class="ui-state-default">#ff0000</li>
  <li class="ui-state-default">#0000ff</li>
  <li class="ui-state-default">#ff00b3</li>
  <li class="ui-state-default">#ded447</li>
  <li class="ui-state-default">#125e18</li>
</ul>
```

    iii.   Adding a control to sort the list and disable the selection

```javascript
$("#sortable").sortable();
$("#sortable").disableSelection();
```

    iv.   Check your output, now you can drag any color to a new position
    v.   Add the following style to your code

```css
#sortable {
    list-style-type: none;
    margin: 0;
    padding: 0;
    width: 150px;
}
#sortable li {
    margin: 0 3px 3px 3px;
    padding: 0.4em;
    padding-left: 1.5em;
    font-size: 1.0em;
    height: 12px;
}
```

      vi.   What do you think this CSS did? You can reverse the step to see the changes after and before the use of CSS

      vii.   Add the following span with class before color code in the <li> in (ii) above

```html
<span class="ui-icon ui-icon-arrowthick-2-n-s"></span>
```

e. Adding a Tooltip

      i.   Add the following code inside document ready function

```javascript
$(document).tooltip({
  items: '[title], [class]',
  position: { my: "left center", at: "right center" },
  content: function() {
    var element = $(this);
    if (element.is('[title]')){
      return element.attr('title');
    }
    if (element.is('[class]')){
      console.log('class', element);
      if (element.hasClass('ui-state-default')) {
        var text = element.text();
        return '<div class="color_swatch" style="background:' +
              text + '"></div>'
      }
    }
  }
});
```

      ii.   Add the following css

```css
.color_swatch{ height:50px; width: 50px; }
```

The Output :

Task 4
   a. Get the code from https://frend.odaje.biz/lab5/task3.html, and evaluate and test
      the code.

   b. Getting the context of the canvas

        a. Add the following code:
```
var canvas = document.getElementById("myCanvas");
var ctxt = canvas.getContext('2d');
```
             i. What is the meaning of context in this case?
   c. Drawing a triangle to canvas
        a. Setup the fillStyle, strokeStyle and linewidth.
```
ctxt.fillStyle    = '#f00';
ctxt.strokeStyle = '#000';
ctxt.lineWidth    = 3;
```
        b. Describe the purpose of each method in (a)
        c. Put these codes into an appropriate function:
```
ctxt.beginPath();
ctxt.moveTo(60, 240);
ctxt.lineTo(240, 240);
ctxt.lineTo(150, 60);
ctxt.closePath();
ctxt.fill();
ctxt.stroke();
```
        d. Test you code and evaluate the output
        e. Rearrange the position of coordinate and test your code.
        f. What is the use of moveTo methods above?

   d. a new rotate triangle function
        a. Set the new position of the new triangle at the middle of previous triangle
```
ctxt.translate(canvas.width / 2, canvas.height / 2);
ctxt.rotate(Math.PI / 4);
ctxt.beginPath();
```
        b. Start drawing a new triangle using a new coordinate
```
ctxt.moveTo(-90, 90);
ctxt.lineTo(90, 90);
ctxt.lineTo(0, -90);

ctxt.closePath();
ctxt.fill();
ctxt.stroke();
```
        c. Test and evaluate your result.
        d. You can press the rotate many times, and can have different triangle at
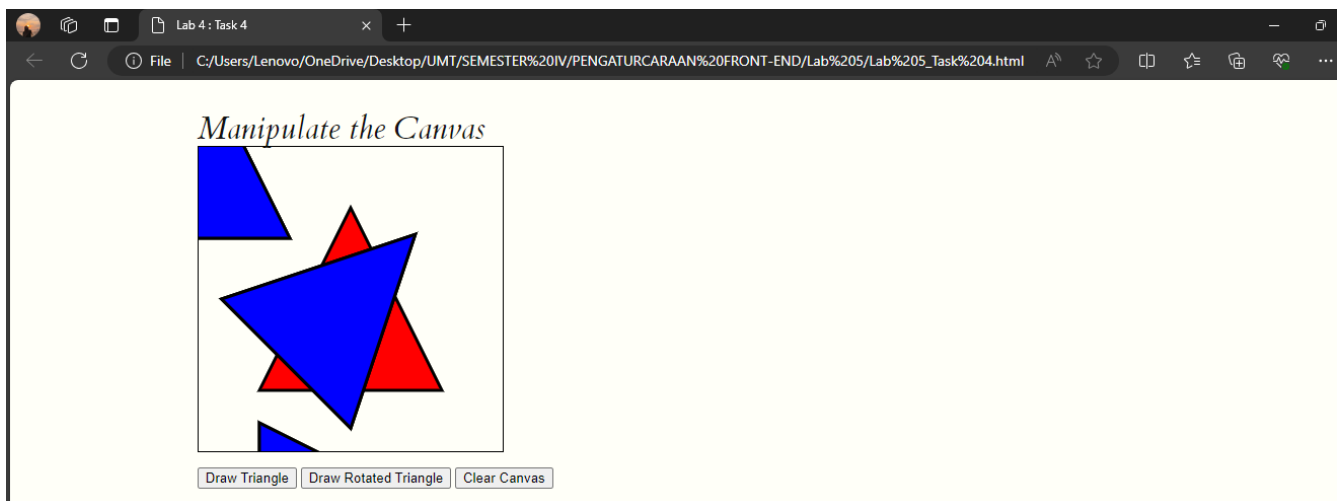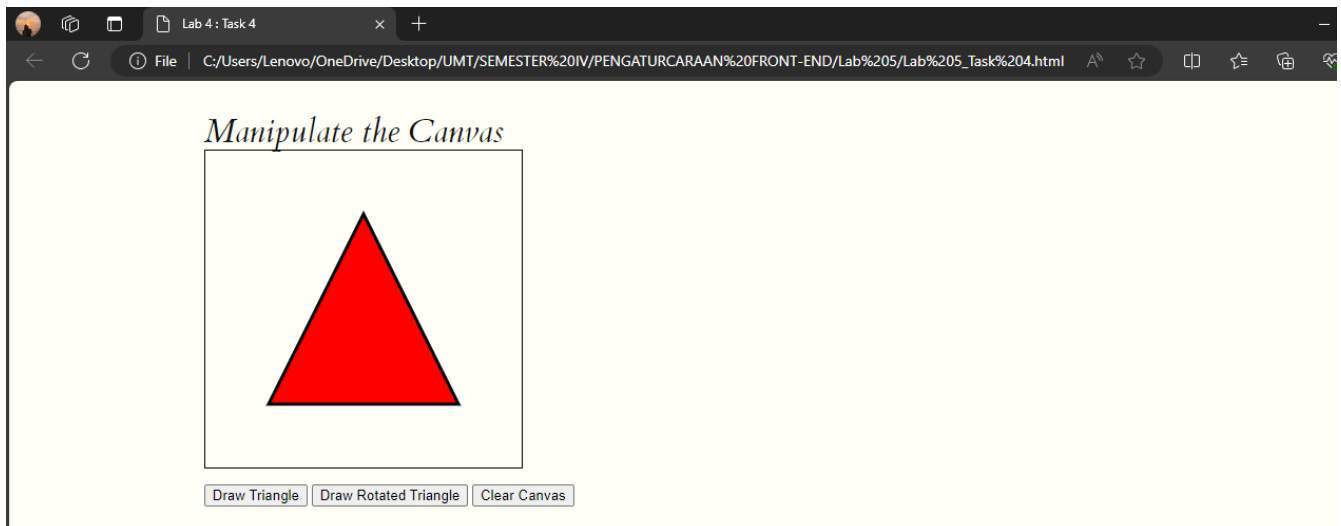   different location, why it's happened?

   e. Clear the canvas
        a. Put this code to clear the canvas

```
ctxt.clearRect(0, 0, canvas.width, canvas.height);
```

b. Can you write more better clear canvas , to clear all canvas

The Output :

Notes: When I click the button clear canvas the triangle will be clear.