WIF2003 Web Programming

Tutorial 04: JavaScript and DOM

Question 1:

```html
<!DOCTYPE html>
<html>
<head>
<title>Apple Tours</title>
</head>
<body>
    <div id="main">
        <h1>Hello <span id="there">there</span></h1>
        <p>We provide the tours to these countries:</p>
        <ul>
            <li>France</li>
            <li>Spain</li>
            <li>Thailand</li>
        </ul>
        <a href="book.html" class="special">
            <button type="button">Book Tour</button>
        </a>
    </div>
</body>
</html>
```

Given the above HTML codes, write JavaScript codes using DOM querySelector() and querySelectorAll() methods to:

a)   *Select* all **<li>** and print out the list items in the browser's JavaScript console

b)   *Select* the <a> with **class** "**specia**l" and print a pop-up message "Book a tour" when it is moused over

c)   Define a new CSS class with the styles (i.e. blue font color and a dashed blue border). *Select* the <span> element with **id "there"** and toggle the new CSS class to manipulate the styles of the element.

NOTE: Refer to lecture notes for the sample codes to SELECT an element and save it to a variable, then MANIPULATE the style

Reference link – HTML DOM Document:
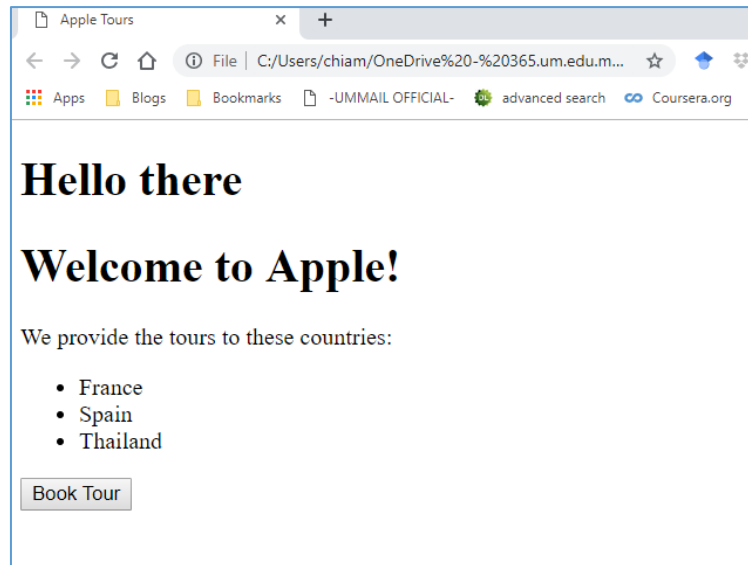**https://www.w3schools.com/jsref/dom_obj_document.asp**
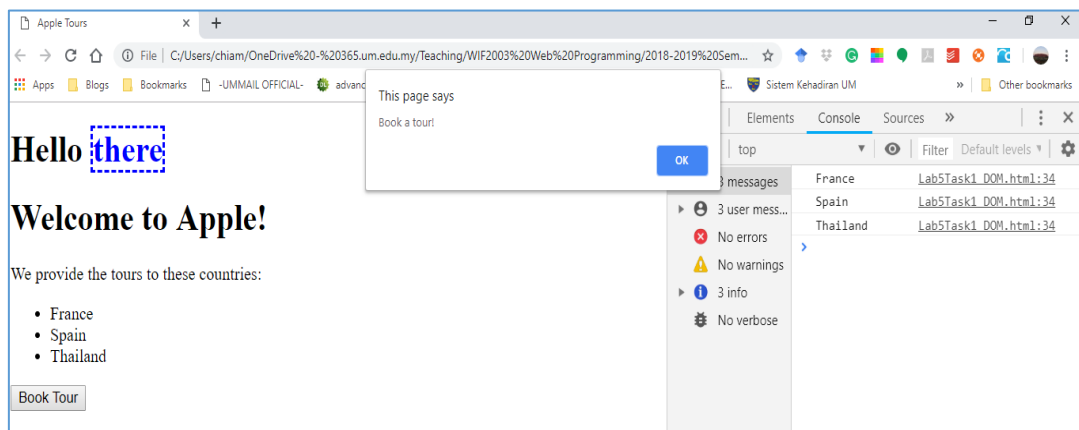
Figure 1: Screenshot before using DOM

Figure 2: Screenshot after using DOM

Question 2: Using DOM in JavaJam Coffee House website

In this task, you have to do the following steps:

1. Open the jobs.html file in your javajam5 project folder to modify the dropdown list codes *("Select job position:")*.

Remove/Comment the following codes from "jobs.html":

```
<select id="select-choice" class="form-control col-sm-4" >
      <option value="Barista">Barista</option>
      <option value="Manager">Manager</option>
      <option value="Trainee">Trainee</option>
</select>
```

2. Write JavaScript codes to replace the codes that have been removed/commented.

```
<label for="select-choice">Select job position:</label>
<script>
//Write codes for 2(a), 2(b), 2(c), 2(d)



</script>
```

a) Define an array of three `jobclass` objects (see Table 1). Each `jobclass` object should have `code`, `name`, and `scope` properties.

   Sample code to define an array:
```
var person = [{firstName:"John", lastName:"Doe", age:46},
              {firstName:"Aminah", lastName:"Ali", age:40},
             ];
```

   Table 1: Three `jobclass` objects with `code`, `name` and `scope` properties

code	name	scope
ba	Barista	Preparing and serving hot and cold drinks such as coffee, tea, artisan and speciality beverages.
ma	Manager	Managing day-to-day operations of the cafe.
tr	Trainee	Supporting daily operations of the cafe.

b) Given the provided array of **jobclass**, using `document.write` and a loop to generate a drop-down list that display the **jobclass** `name`. The `value` attribute for each item in the list should be the **jobclass** `code`. The sample browser screenshot should look like **Figure 3**.
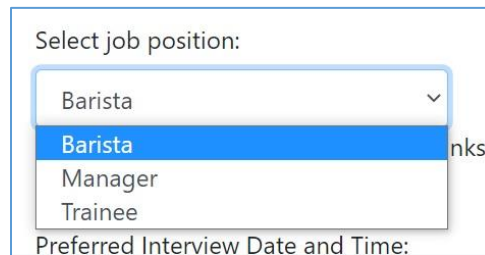
Figure 3: Screenshot for the drop-down list that display the list of job positions

c) Add a new paragraph <p> HTML element with an id named `"jobScope"` after the closing tag of </script> to display the scope of each job position (see Figure 4).

```
<script>


</script>
<p id="jobScope"> Preparing and serving hot and cold drinks such
as coffee, tea, artisan and speciality beverages.</p>
```
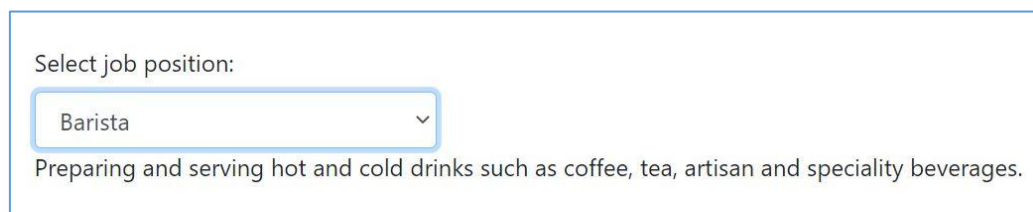

Figure 4: Screenshot of the paragraph that display job scope description

d) Write JavaScript code using DOM to add an **event listener** to handle the **change event** of the drop-down list. This event handler should **display the job scope** of the currently **selected job position** within the paragraph `<p id="jobScope"></p>`. The sample browser screenshots should look like Figure 5 and Figure 6.

Hint: Your event handler may use the `value` property of the drop-down list to which show the 'code' of the `jobclass` in the array in order to retrieve the right job scope ('scope') in the array to be displayed in the `textContent` or `innerText` property of the `<p id="jobScope">`.
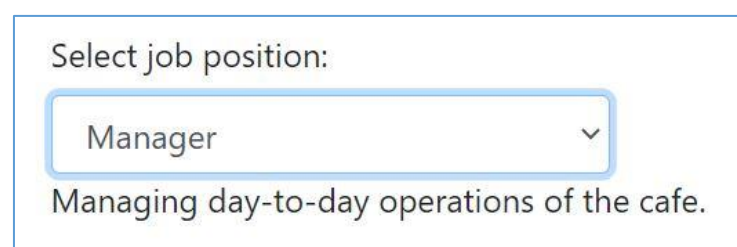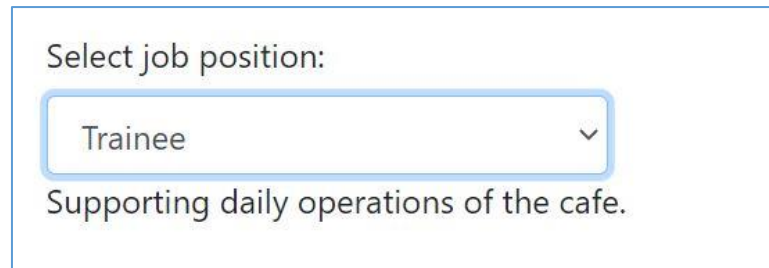

*Figure 4: Screenshot when "**Manager**" job position is selected, the browser should display the description "**Managing day-to-day operations of the cafe.**"*

Figure 5: *Screenshot when "**Trainee**" job position is selected, the browser should display the description "**Supporting daily operations of the cafe.**"*