

Scenario

In this exercise, you will create the Schedule page that displays a list of sessions.

First, you will use the HTML5 DOM to obtain a reference to the page's schedule list element. Then you will implement a function that creates list items (one list item for each session). Information about the sessions is stored in a file in JSON format. You will implement a function that reads this data and adds the details of each session to the list element. Finally, you will run the application and view the Schedule page to verify that it correctly displays the list of sessions.

The main tasks for this exercise are as follows:

1. Review the existing code for the Schedule page.
2. Write code to get the schedule list element on the Schedule page.
3. Implement the `createSessionElement` function that creates the list item for a session.
4. Implement the `displaySchedule` function that adds session items to the list for display.
5. Run the web application and view the Schedule page.

► Task 1: Review the existing code for the Schedule page

1. In the **Exercise 1** folder, review the content of the page **schedule.htm**. Notice that the **schedule** page section, which will be used to display the list of sessions, currently contains an empty list, also named **schedule**:

```
<section class="page-section schedule">
  <div class="container">
    <h1>Schedule</h1>
    <ul id="schedule"></ul>
  </div>
</section>
```

2. Also notice that the `schedule.htm` page references the JavaScript code in **schedule.js** script file:

```
<script src="schedule.js" type="text/javascript"></script>
```

3. Review the **schedule.js** script file. This file contains the details of each session held in JSON format. The data is held in an array named **schedule**, and each object in the array has three properties that specify the session id, the session title, and the tracks to which the session belongs (a session may be part of more than one track):



```
var schedule = [  
  {  
    "id": "session-1",  
    "title": "Registration",  
    "tracks": [1, 2]  
  },  
  {  
    "id": "session-2",  
    "title": "Moving the Web forward with HTML5",  
    "tracks": [1, 2]  
  },  
  {  
    "id": "session-3",  
    "title": "Diving in at the deep end with Canvas",  
    "tracks": [1]  
  },  
  {  
    "id": "session-4",  
    "title": "New Technologies in Enterprise",  
    "tracks": [2]  
  },  
  ...  
]
```

► Task 2: Write code to get the schedule list element on the Schedule page

1. In the **schedule.js** file, find the comment **TODO: Task 2**:
2. Write JavaScript code to get the **schedule** list element from the DOM and assign it to a variable named **list**. You will use this variable to display the details of each session in the list on the Schedule page.
3. Use the **jQuery** to find the list that has the **id** property set to **schedule**.

► Task 3: Implement the **createSessionElement** function that creates the list item for a session.

1. In the **schedule.js** file, find the comment **TODO: Task 3**. This comment is located in the **createSessionElement** function, which looks like this:

```
function createSessionElement(session) {  
  ...  
};
```

The purpose of this function is to create a list element containing the name of the session passed in as the parameter.

2. Add JavaScript code to return a **** element, set its text content to the session title.



► **Task 4: Implement the `displaySchedule` function that adds session items to the list for display.**

1. In the **`schedule.js`** file, find the **TODO: Task 4** comment. This comment is located in the `displaySchedule` function, which looks like this:

```
function displaySchedule() {  
    clearList();  
    ...  
};
```

The purpose of this function is to display the title of each session in the list on the Schedule page.

2. Add JavaScript code to iterate over the **`schedule`** array containing the JSON data by using a **for** loop. Create a **session** object for each item in the array and add the title of the session to the **list** element on the Schedule page.

- Use the **`createSessionElement`** function that you implemented in **Task 3** to create a list item for each session.
- Use the **list** variable that you created in **Task 2** to access the list element on the Schedule page.

► **Task 5: Run the web application and view the Schedule page.**

1. Run the application and view the `schedule.htm` page to verify that the list of sessions is displayed. The Schedule page should look like this:

