

Hands-on Lab: Develop Product Feedback Survey Form Using Events



Estimated time needed: 30 minutes

What You Will Learn

In this lab, you will learn how to create a user feedback form for a luxury brand's products using HTML and JavaScript. You will discover how to design a structured form to collect diverse user information such as name, age, email, job, and product feedback. You will see how to use the JavaScript functions to manage form submission, capture input values, and dynamically display user-provided feedback on the webpage. You will also observe event handling mechanisms.

Learning objectives

After completing this lab, you will be able to:

- **Form handling and user input capture:** Explore the process of creating an interactive form using HTML elements, such as text inputs, select options, and text areas, to collect diverse user information, including name, age, email, job, product feedback, and more.
- **DOM manipulation with JavaScript:** Learn how JavaScript interacts with the Document Object Model (DOM) to retrieve input values from form elements and dynamically update the webpage content. This includes modifying the display style of an HTML element and populating its content based on user input.
- **Event handling and user interactions:** Explore event handling mechanisms in JavaScript, specifically the implementation of event listeners for button clicks and **Enter** key presses to trigger actions, such as submitting the form data and displaying feedback dynamically, providing a seamless user experience.
- **User interface interaction design:** Analyse the importance of presenting collected user information in a readable format on the webpage. Learners will gain insights into dynamically displaying user-provided feedback in a structured manner within the webpage, enhancing the overall user interface and experience.

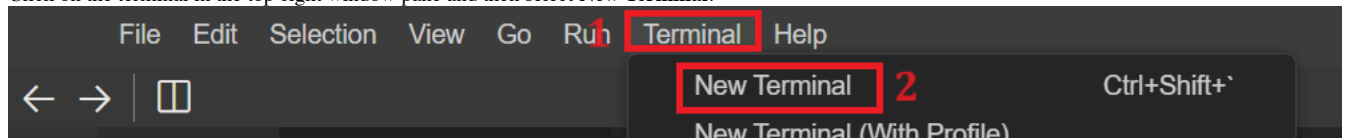
Prerequisites

- Basic Knowledge of HTML.
- Basic understanding of events and function in JavaScript.
- Web browser with a console (Chrome DevTools, Firefox Console, and so on).

Step 1: Setting up the environment

1. Firstly, you need to clone your main repository in the **Skills Network Environment** which you have created in the first lab and where you have pushed all of your previous labs files and folders. Follow the given steps:

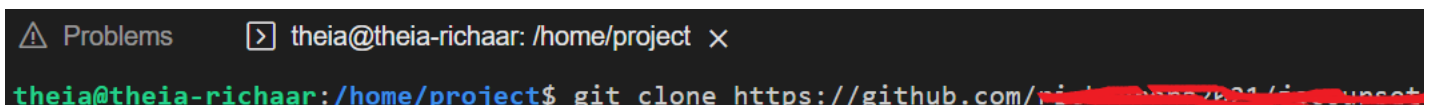
- Click on the terminal in the top-right window pane and then select **New Terminal**.



- Perform `git clone` command by writing given command in the terminal.

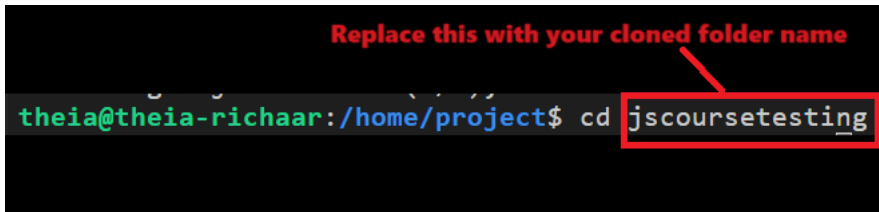
```
git clone <github-repository-url>
```

Note: Put your own GitHub repository link instead of `<github-repository-url>`.



- Above step will clone folder for your GitHub repository under project folder in explorer. You will also multiple folders inside cloned folder.
- Now you need to navigate inside the cloned folder. For this write given command in the terminal:

```
cd <repository-folder-name>
```



Note: Write your cloned folder name instead of <repository-folder-name>. Perform `git clone` if you have logged out of **Skills Network Environment** and you cannot see any files or folder after you logged in.

- Now select **cloned Folder Name**, right click on it and click on **New Folder**. Enter folder name as **productSurvey**. It will create the folder for you. Then select **productSurvey** folder, right click and select **New File**. Enter file named as **product_survey.html** and click OK. It will create your HTML file.
- Now, select the **productSurvey** folder again, right click and select **New File**. Enter file named as **product_survey.js** and click OK. It will create your JavaScript file.
- Create a basic template structure of HTML file by adding the content provided below.

```
<!DOCTYPE html>
<html>
<head>
  <title>Luxury Brand Feedback</title>
</head>
<body>
  <h1>Welcome to Luxury Brand Products Feedback</h1>
  <h2>Provide Your Feedback</h2>
  <label for="name">Name:</label><br>
  <input type="text" id="name" placeholder="Your Name" required><br>
  <label for="age">Age:</label><br>
  <input type="number" id="age" placeholder="Your Age" required><br>
  <label for="email">Email:</label><br>
  <input type="email" id="email" placeholder="Your Email" required><br>
  <label for="job">Job:</label><br>
  <input type="text" id="job" placeholder="Your Job" required><br>
  <label for="designation">Designation:</label><br>
  <input type="text" id="designation" placeholder="Your Designation" required><br>
  <label for="productType">Product Type:</label><br>
  <select id="productType" required>
    <option value="perfume">Perfume</option>
    <option value="cream">Cream</option>
    <option value="oils">Oils</option>
  </select><br>
  <label for="feedbackText">Feedback:</label><br>
  <textarea id="feedbackText" rows="5" cols="30" placeholder="Enter your feedback here" required></textarea><br>
  <button id="submitBtn">Submit Feedback</button>
<hr>
<div id="userInfo" style="display: none;">
  <h2>User Information:</h2>
  <p>Name: <span id="userName"></span></p>
  <p>Age: <span id="userAge"></span></p>
  <p>Email: <span id="userEmail"></span></p>
  <p>Job: <span id="userJob"></span></p>
  <p>Designation: <span id="userDesignation"></span></p>
  <p>Feedback for <span id="userProductChoice"></span>: <span id="userFeedback"></span></p>
</div>
<script src="./product_survey.js"></script>
</body>
</html>
```

- **User feedback form:** The HTML structure comprises of input fields for the user's name, age, email, job, designation, a dropdown to select a product type (perfume, cream, oils), and a text area for feedback.
- **Submission button:** The "Submit Feedback" button triggers the processing of user-entered details using JavaScript when clicked.
- **Dynamic display:** Upon submission, JavaScript processes the entered information and dynamically displays the user's provided details, including name, age, email, job, designation, and feedback for the chosen product type.
- **Hidden section display:** Initially hidden, a section beneath the form becomes visible after submission, showing the user's information and feedback and providing a streamlined view of the entered details.
- To include js file in **product_survey.html** script tag is used in HTML file above the </body> tag to include the script file.

Note: When you have pasted the code, save your file.

Step 2: Defining variables and function

1. Create a function named **submitFeedback** inside **product_survey.js** to take the feedback from the user survey form created in **product_survey.html** file.

```
function submitFeedback() {  
}
```

2. Variables like username, age, email, job, designation, productType, and feedback are declared and assigned values from the respective HTML input elements using `getElementById`. For example:

```
const username = document.getElementById('name').value;  
const age = document.getElementById('age').value;  
const email = document.getElementById('email').value;  
const job = document.getElementById('job').value;  
const designation = document.getElementById('designation').value;  
const productType = document.getElementById('productType').value;  
const feedback = document.getElementById('feedbackText').value;
```

The above code retrieves the values using HTML input elements with the specified IDs and stores them in variables.

3. Next, declare a variable named **submitButton** and initialize it with `document.getElementById('submitBtn')`; This line of code fetches the HTML element with the ID 'submitBtn' using its unique identifier ('getElementById') and assigns it to the variable 'submitButton' for further interaction or manipulation in JavaScript.

```
const submitButton=document.getElementById('submitBtn');
```

4. Now, assigns the function **submitFeedback** to execute when the **onclick** event occurs on the HTML element referenced by the variable **submitButton**. Include given code outside the function in **product_survey.js** file.

```
submitButton.onclick = submitFeedback;
```

5. Include the given code in the **submitFeedback** function after the declaration of the feedback variable to display thank you message when the user submits the form.

```
alert('Thank you for your valuable feedback')
```

Step 3: Display user feedback on webpage

1. As the **submitFeedback** function will fetch the user feedback information in the second point of step 2, the details should also be displayed on the web page to inform the user about the information entered.
2. To achieve this, include the following code inside **submitFeedback** function to put information.

```
document.getElementById('userName').innerHTML = username;
document.getElementById('userAge').innerHTML = age;
document.getElementById('userEmail').innerHTML = email;
document.getElementById('userJob').innerHTML = job;
document.getElementById('userDesignation').innerHTML = designation;
document.getElementById('userProductChoice').innerHTML = productType;
document.getElementById('userFeedback').innerHTML = feedback;
```

3. For example, the line of code `document.getElementById('userName').innerHTML = username;`, finds the HTML element with the ID **userName** and replaces its inner HTML content with the value stored in the variable **username**. It dynamically updates the displayed content within that specific HTML element.
4. In the HTML code at line 29, the entire div element, which will be used to display information, is styled with the 'display: none' property to be visible only after the user clicks on the submit button. To make this part visible after the button click, include the following code inside the **submitFeedback** function.

```
document.getElementById('userInfo').style.display = 'block';
```

5. Above line of code uses JavaScript to access an HTML element identified by the ID 'userInfo'. It then modifies the CSS style property 'display' of that element, setting it to 'block', which alters its visibility on the webpage to be displayed as a block-level element.

Step 4: Implement key press to submit feedback

1. Include the following code at the end of the **product_survey.js** file to ensure that if the user presses the 'Enter' key, the feedback form should submit, similar to clicking the submit button:


```
document.addEventListener('keydown', function(event) {
  if (event.key === 'Enter') {
    submitFeedback();
  }
});
```

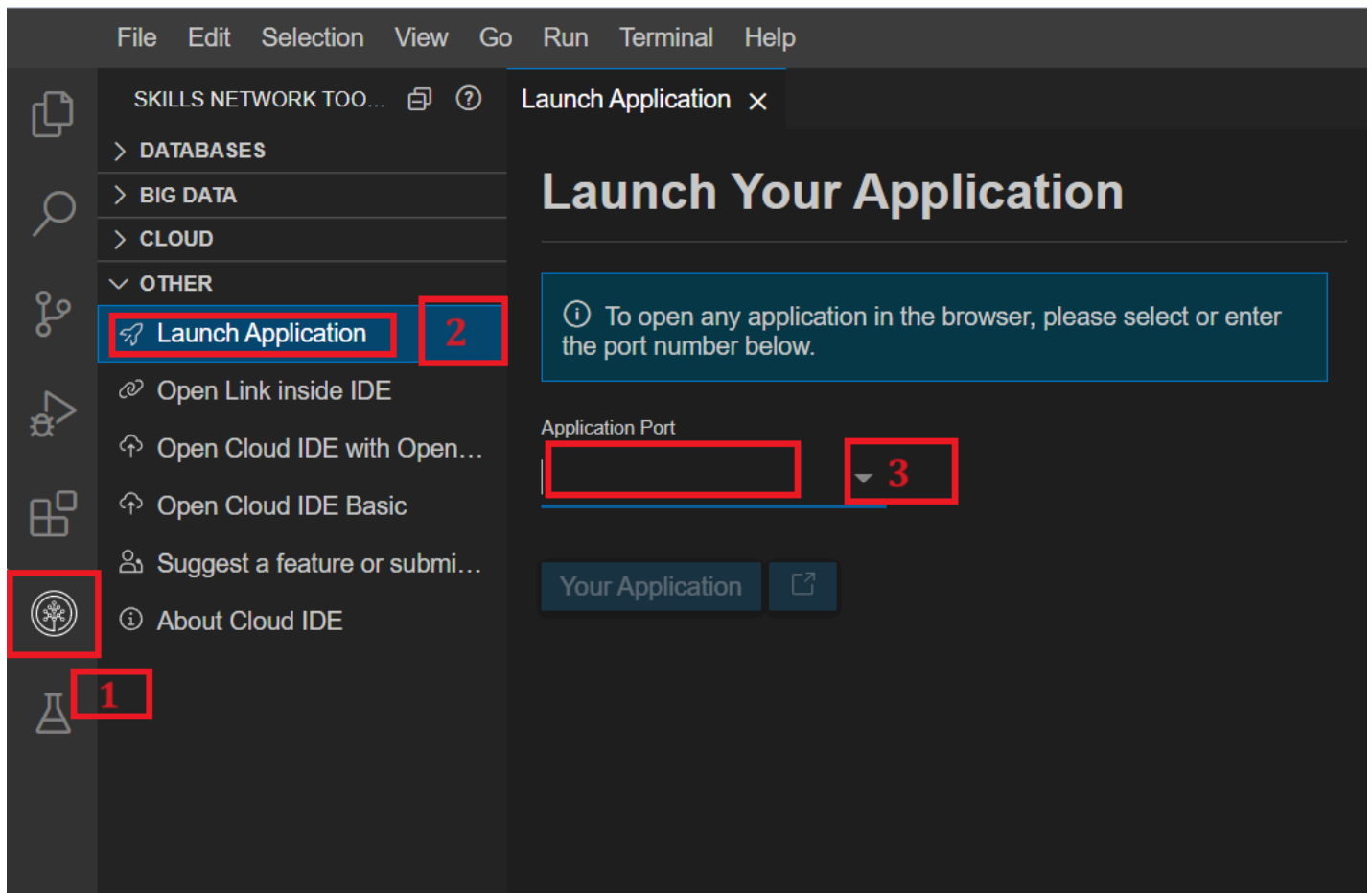
2. This code snippet adds an event listener to the entire document, listening for any keydown event. When the event occurs, it checks if the pressed key is 'Enter'. If it is, it calls the function `submitFeedback()`, executing the feedback submission functionality.

Step 5: Check the output

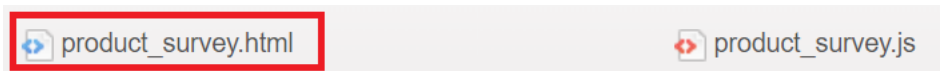
1. To view your HTML page, right-click on the **product_survey.html** file after selecting this file, then select "Open with Live Server".
 - The server should start on port 5500, indicated by a notification on the bottom right side.



2. Click on the Skills Network button on the left (refer to number 1), it will open the "Skills Network Toolbox". Click on Launch Application (refer to number 2). From there, you enter the port no. as 5500 at number 3 and click on this button .



3. It will open your default browser where you will see **cloned-folder-name** folder name. Click on that **cloned-folder-name** folder name. After clicking you will see multiple folders name, among those folders name click on **productSurvey** folder. You will see files related to this folder where again you will click on **product_survey.html** file as shown below.



4. It will open the HTML page and you will see the feedback form as below.

Welcome to Luxury Brand Products Feedback

Provide Your Feedback

Name:

Age:

Email:

Job:

Designation:

Product Type:

Feedback:

5. Fill the feedback form and then click on **Submit Feedback** button.

6. As soon as you submit the feedback form, you will see an alert pop up box displaying the thank you message.

Thank you for your valuable feedback

7. After clicking the OK button in the alert popup box, the details of the feedback form will be displayed below the feedback form that the user filled before submitting. The screenshot below provides a demonstration of how the details will be displayed.

User Information:

Name: Peter

Age: 23

Email: peter@gmail.com

Job: Quality Assurance

Designation: Manager

Feedback for perfume: It has mild fragrance.

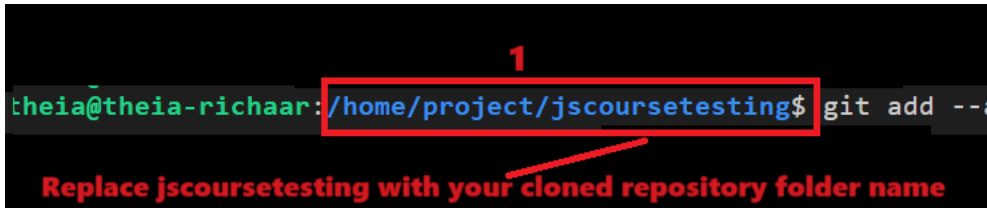
Note: After pasting the code, save your file. You can use any output method for this. If you edit your code, simply refresh your browser running through port number 5500. This way, there is no need to launch the application again and again.

Step 6: Perform Git commands

1. Perform `git add` to add the latest files and folder in the git environment.

```
git add --a
```

- Make sure the terminal has the path as follows:



2. Then perform `git commit` in the terminal. While performing `git commit`, terminal can show message to set up your `git config --global` for `user.name` and `user.email`. if yes, then you need to perform `git config` command as well for `user.name` and `user.email` as given.

```
git config --global user.email "you@example.com"
```

```
git config --global user.name "Your Name"
```

Then, perform commit command as given:

```
git commit -m "message"
```

3. Then perform `git push` just by writing given command in terminal.

```
git push origin
```

- After push command username and password will be asked to enter. You need to enter username for your GitHub account and password that you have created in first lab. After entering password and username, all of your latest folders and files will be pushed in your GitHub repository.

Practice task

1. In this task you need to create one more text area to ask users to share their experience with the product as well and show the result along with other details after clicking on **Submit Feedback** button.
2. To complete this task, you can use `<textarea>` tag in html file and give `id='userExperience'` as attribute for this tag.

3. Then access the details entered by user in javaScript file by accessing the input of this textarea using `document.getElementById('userExperience')` and store its value in one variable.
4. Then show the output after button is clicked with other user's entered details.

Summary

1. **Feedback form elements:** The HTML structure includes input fields for name, age, email, job, designation, a dropdown for product type selection, and a textarea for feedback submission. This comprehensive design ensures the collection of user-provided information.
2. **Dynamic user information display:** JavaScript function "submitFeedback()" processes user input upon button click or Enter key press, displaying entered details dynamically below the form, including name, age, email, job, designation, selected product type, and the provided feedback.
3. **Visibility toggle:** Initially hidden, a section below the form remains concealed until feedback submission. Upon submission, JavaScript alters its display to "block," showcasing the user's information and feedback for the selected product type.
4. **Event listeners implementation:** Event listeners are employed to trigger the feedback submission function, enabling user interaction via both the "Submit Feedback" button click and the Enter key press for a seamless experience.

© IBM Corporation. All rights reserved.