Intern - JavaScript Exercises

Task 1: Implement a Countdown Timer

Instructions:

- Create an HTML page with:
    - A date and time picker is input for the user to select the target date and time.
    - A button to start the countdown.
    - A display area to show the remaining time until the target date and time.

Requirements:

1. **Functionality:**
    - The user should be able to:
        1. Select a future date and time.
        2. Start the countdown by clicking the "Start Timer" button.
        3. View the remaining time until the selected date and time in a readable format (e.g., days, hours, minutes, seconds).
    - The countdown should be running and updated every second.
    - Use JavaScript to handle the countdown logic.
2. **Data Persistence:**
    - Store the selected date and time in a cookie when the user starts the countdown.
    - On page load, retrieve the date and time from the cookie and initialize the countdown based on this value.
    - Ensure that the countdown continues correctly even if the page is refreshed.
3. **Completion Message:**
    - When the countdown reaches zero, display a message saying "Time’s up!"

Task 2: Build a Simple Form with Validation

Instructions:

- Create an HTML form with the following fields:
  - Name (text)
  - Email (email)
  - Age (number)
  - Gender (dropdown)
  - Submit button

Requirements:

1. **Form Functionality:**
   - Use JavaScript to validate the form:
     1. All fields are required.
     2. The email must be in a valid email format.
     3. Age must be a number greater than 18.
   - Display appropriate error messages below the fields if validation fails.
   - Prevent the submission of the form until all validation rules are met.
2. **Data Persistence:**
   - When the form is successfully submitted, store the form data in `localStorage`.

Task 3: Create a Simple To-Do List Application

Instructions:

- Create a basic HTML page with:
    - An input field for entering tasks.
    - A date picker input for setting the due date.
    - A button to add tasks to the list.
    - A list to display tasks with options to mark them as complete or delete them.

Requirements:

1. **Functionality:**
    - The user should be able to:
        1. Enter a task and set a due date.
        2. Add the task to the list.
        3. Mark tasks as complete by clicking a "Complete" button next to each task.
        4. Remove tasks from the list by clicking a "Delete" button.
        5. Show confirmation dialogue when we click on Delete and Complete.
    - Use JavaScript to handle the functionality for adding, marking, and deleting tasks.
2. **Data Persistence:**
    - Store the tasks in `localStorage` so that the tasks persist even when the page is refreshed.
    - On page load, retrieve the tasks from `localStorage` and display them in the list.
3. **Visual Indicators:**
    - Completed tasks should visually change (e.g., strikethrough text).
    - Tasks that have passed their due date and not completed should be displayed in red.
4. **Input Validation:**
    - Prevent adding tasks if the input field or due date is empty. Show a message or indication if either is missing.

Tasks	Due date	Action	
~~ABC~~	~~19/09/2024~~	Delete	
ABC 1	19/09/2024	Completed	Delete

ABC 3

Add new Task:

Title:

_____

Due Date:

_____

ADD