MINIPROJECT-3[717821I147]

Task Scheduler:

Objective: Develop a web-based task management application where users can add, delete, modify, and Page 17 of 19 save tasks to local storage. The application will incorporate asynchronous programming, DOM operations, and JSON serialization for storing and retrieving tasks**.**

Code:

HTML:

```html
<!DOCTYPE html>
<html>
<head>
<title>Task Manager</title>
<link rel="stylesheet" href="style.css">
</head>
<body>
<h1>Task Manager</h1>
<div id="task-list">
<ul>
<li>
<span class="task-name">Complete project</span>
<span class="due-date">Due: 2023-09-15</span>
<span class="priority">High</span>
<span class="completed">Not Completed</span>
<button class="delete-task">Delete</button>
</li>
</ul>
</div>
<form id="add-task-form">
<input type="text" id="task-name" placeholder="Task Name">
<input type="text" id="due-date" placeholder="Due Date">
<input type="text" id="priority" placeholder="Priority">
<button type="submit">Add Task</button>
</form>
<script src="script.js"></script>
```

```
</body>
</html>
```

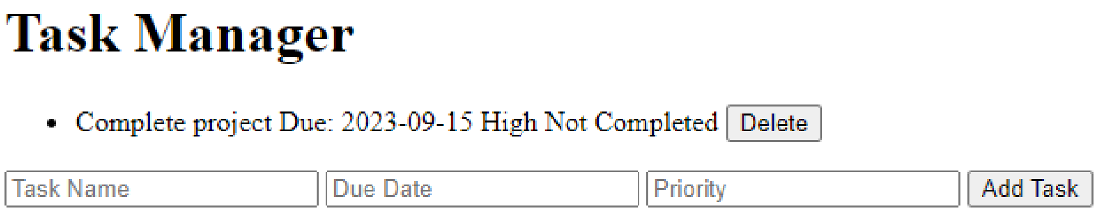
Css:

```
body {
    font-family: Arial, sans-serif;
    }

    h1 {
    text-align: center;
    }
    #task-list {
    margin: 20px;
    }
    ul {
    list-style: none;
    padding: 0;
    }
    li {
    display: flex;
    justify-content: space-between;
    align-items: center;
    border: 1px solid #ccc;
    padding: 10px;
    margin-bottom: 10px;
    }
    .task-name {
    font-weight: bold;
    }
    .completed {
    color: red;
    }
    .delete-task {
    background-color: #ccc;
    border: none;
    padding: 5px 10px;
    cursor: pointer;
    }
    .delete-task:hover {
    background-color: #aaa;

    }
```

Javascript:

```
const taskList = [];
function renderTasks() {
const taskListElement = document.querySelector('#task-list ul');
taskListElement.innerHTML = '';
for (let i = 0; i < taskList.length; i++) {
const task = taskList[i];
const taskElement = document.createElement('li');
taskElement.innerHTML = `
<span class="task-name">${task.taskName}</span>
<span class="due-date">Due: ${task.dueDate}</span>
<span class="priority">${task.priority}</span>
<span class="completed">${task.completed ? 'Completed' : 'Not
Completed'}</span>
<button class="delete-task">Delete</button>
`;
taskListElement.appendChild(taskElement);
const deleteButton = taskElement.querySelector('.delete-task');
deleteButton.addEventListener('click', () => {
taskList.splice(i, 1);
renderTasks();
});
}
}
const addTaskForm = document.querySelector('#add-task-form');
addTaskForm.addEventListener('submit', (event) => {
event.preventDefault();
const taskNameInput = document.querySelector('#task-name');
const dueDateInput = document.querySelector('#due-date');
const priorityInput = document.querySelector('#priority');
const task = {
taskName: taskNameInput.value,
dueDate: dueDateInput.value,
priority: priorityInput.value,
completed: false,

};
taskList.push(task);
taskNameInput.value = '';
dueDateInput.value = '';
priorityInput.value = '';
renderTasks();
});
renderTasks();
```

Output;



Task Manager

- Complete project Due: 2023-09-15 High Not Completed