

```
1 <!DOCTYPE html>
2 <html lang="en">
3   <head>
4     <meta charset="UTF-8">
5     <meta name="viewport" content="width=device-width, initial-scale=1.0">
6     <title>To-Do List</title>
7   <style>
8     body{
9       font-family: Arial, Helvetica, sans-serif;
10      background-color: #ffffff;
11      display: flex;
12      justify-content: center;
13      align-items: center;
14      height:100vh;
15      margin:0;
16    }
17    .todo-container{
18      background-color: rgb(65, 65, 65);
19      border-radius: 8px;
20      padding:20px;
21      width: 500px;
22      box-shadow: 0 4px 10px rgba(255, 253, 253, 0.1);
23    }
24    .todo-container h2{
25      text-align: center;
26    }
```

```
2  <html lang="en">
3    <head>
7    <style>
24  }
25  .todo-container h2{
26    text-align: center;
27    color: #f4ff2b;
28    font-size: 24px;
29  }
30  .input-wrapper{
31    display: flex;
32    flex-direction: column;
33    margin-bottom: 20px;
34  }
35  .input-wrapper input,
36  .input-wrapper input[type="datetime-local"]{
37    width: 96%;
38    padding: 10px;
39    margin-bottom: 10px;
40    border: 1px solid #ddd;
41    border-radius: 4px;
42    font-size: 17px;
43  }
44  .input-wrapper button{
45    padding: 10px;
46    background-color: #4CAF50;
```

```
2 <html lang="en">
3 <head>
7 <style>
44 .input-wrapper button{
45     background-color: #f9f9f9;
47     border:none;
48     border-radius: 4px;
49     color: #white;
50     font-size: 17px;
51     cursor: pointer;
52     transition: background-color 0.3s;
53 }
54 .input-wrapper button:hover{
55     background-color: #45a049;
56 }
57 .task-list{
58     list-style-type: none;
59     padding:0;
60 }
61 .task-item{
62     background-color: #f9f9f9;
63     margin: 8px 0;
64     padding: 16px;
65     border-radius: 4px;
66     display:flex;
67     justify-content: space-between;
68     align-items: center;
```

```
2 <html lang="en">
3 <head>
7 <style>
61 .task-item{
68     align-items: center;
69     box-shadow: 0 2px 4px rgba(0,0,0,0.1);
70     position: relative;
71 }
72 .task-item.completed {
73     background-color: #d4eddb; /*Greenish background for completed tasks*/
74     text-decoration: line-through;
75 }
76 .task-item button{
77     background-color: #e74c3e;
78     border: none;
79     border-radius: 4px;
80     color: white;
81     font-size: 10px;
82     cursor: pointer;
83 }
84 .task-item button:hover {
85     background-color: #c0392b ;
86 }
87 .task-item .status {
88     font-weight: bold;
89     color: rgb(250, 163, 2); /* orange color for pending*/
```

```
2 <html lang="en">
3   <head>
7     <style>
87     .task-item .status {
89       color: rgb(250, 163, 2); /* orange color for pending*/
90     }
91     .task-item .status.completed{
92       color: #28a745;
93     }
94     .task-item .date-time {
95       font-size: 13px;
96       color: #888;
97     }
98     .task-item {
99       margin-bottom: 10px;
100    }
101    .task-text {
102      margin-right: 10px;
103    }
104    .date-time {
105      margin-left: 10px;
106      cursor: pointer;
107    }
108    .completed .task-text {
109      text-decoration: line-through;
110      color: gray;
```

```

2  <html lang="en">
3    <head>
7    <style>
111  }
112  </style>
113  </head>
114  <body>
115    <div class="todo-container">
116    <h2>To-Do List</h2>
117    <div class="input-wrapper">
118      <input type="text" id="taskInput" placeholder="Enter a Task...">
119      <input type="datetime-local" id="taskDateTime">
120      <button onclick="addTask()">Add Task</button>
121    </div>
122    <ul id="taskList" class="task-list"></ul>
123  </div>
124
125  <script>
126  function addTask() {
127    // Get task input and datetime
128    const taskInput = document.getElementById('taskInput');
129    const dateTimeInput = document.getElementById('taskDateTime');
130    const taskText = taskInput.value.trim();
131    const customDateTime = dateTimeInput.value || new Date().toLocaleString();
132
133    // Validate input

```

```
2 <html lang="en">
114 <body>
125 <script>
126 function addTask() {
133     // Validate input
134     if (taskText === "") {
135         alert("Please enter a task!");
136         return;
137     }
138
139     // Create task item
140     const li = document.createElement('li');
141     li.classList.add('task-item');
142
143     // Task text
144     const span = document.createElement('span');
145     span.textContent = taskText;
146     span.contentEditable = true;
147     span.classList.add('task-text');
148     li.appendChild(span);
149
150     // Date/time display
151     const dateTimeSpan = document.createElement('span');
152     dateTimeSpan.classList.add('date-time');
153     dateTimeSpan.textContent = `Set for: ${customDateTime}`;
154     dateTimeSpan.onclick = function () {
```

```

2  <html lang="en">
114  <body>
125  <script>
126  function addTask() {
153      dateTimeSpan.textContent = `Set for: ${customDateTime}`;
154      dateTimeSpan.onclick = function () {
155          const newDateTime = prompt("Edit the date and time", customDateTime);
156          if (newDateTime !== null && newDateTime.trim() !== "") {
157              dateTimeSpan.textContent = `Set for: ${newDateTime.trim()}`;
158          }
159      };
160      li.appendChild(dateTimeSpan);
161
162      // Edit button
163      const editButton = document.createElement('button');
164      editButton.textContent = "Save Edit";
165      editButton.onclick = function () {
166          const newText = span.textContent.trim();
167          if (newText === "") {
168              alert("Task description cannot be empty!");
169          } else {
170              span.textContent = newText;
171          }
172      };
173      li.appendChild(editButton);
174
175      // Delete button

```



```
2 <html lang="en">
114 <body>
125 <script>
126 function addTask() {
165     editButton.onclick = function () {
172     };
173     li.appendChild(editButton);
174
175     // Delete button
176     const deleteButton = document.createElement('button');
177     deleteButton.textContent = "Delete";
178     deleteButton.onclick = function () {
179         li.remove();
180         alert("Task has been deleted!");
181     };
182     li.appendChild(deleteButton);
183
184     // Complete button
185     const completeButton = document.createElement('button');
186     completeButton.textContent = "Mark as completed";
187     completeButton.onclick = function () {
188         li.classList.toggle('completed');
189         if (li.classList.contains('completed')) {
190             alert("Task is completed!");
191         } else {
192             alert("Task marked as pending.");
```

```
2 <html lang="en">
114 <body>
125 <script>
126 function addTask() {
187   completeButton.onclick = function () {
191     } else {
192       alert("Task marked as pending.");
193     }
194   };
195   li.appendChild(completeButton);
196
197   // Add to list
198   const taskList = document.getElementById('taskList');
199   taskList.appendChild(li);
200
201   // Clear inputs
202   taskInput.value = "";
203   dateTimeInput.value = "";
204 }
205 </script>
206
207 </body>
208
209 </html>
```

To-Do List

Enter a Task...

dd-mm-yyyy --:--

Add Task

Marksheet

Set for: 27/7/2025, 2:25:40 pm

Save Edit

Delete

Mark as completed

12th Result

Set for: 2025-07-28T14:26

Save Edit

Delete

Mark as completed