

Employee Task Management

You're tasked with building a simple employee task management system. Your goal is to create components to display employee tasks and allow toggling their status between "done" and "not done".

Requirements:

TaskList Component:

Create a TaskList component that displays all employee tasks.

Each task should include a name and a status (either "done" or "not done").

Use state to manage the task list.

ToggleStatus Functionality:

Implement a function in the TaskList component that toggles the status of a task between "done" and "not done" when a button is clicked.

Use the useState hook to update the task status.

Task Breakdown:

TaskList Component:

Render a list of employee tasks with their names and statuses.

Utilize state to manage the task list.

ToggleStatus Functionality:

Implement a function that toggles the status of a task between "done" and "not done" when the corresponding button is clicked.

Use the useState hook to update the task status.

Sample Tasks:

```
const [tasks, setTasks] = useState([
  { name: "Task 1", status: "done" },
  { name: "Task 2", status: "not done" },
  { name: "Task 3", status: "done" }
]);
```

Solution –

App.js

```
import React from "react";  
import TaskList from "../TaskList";
```

```
function App() {  
  return (  
    <div className="App">  
      <TaskList />  
    </div>  
  );  
}
```

```
export default App;
```

```
task-management > src > JS App.js > ...  
1  import React from "react";  
2  import TaskList from "../TaskList";  
3  
4  function App() {  
5    return (  
6      <div className="App">  
7        <TaskList />  
8      </div>  
9    );  
10 }  
11  
12 export default App;  
13 |
```

Index.css

```
body {  
  font-family: 'Arial', sans-serif;  
  background-color: #f9f9f9;  
  margin: 0;  
  padding: 0;  
}  
  
.container {  
  padding: 20px;  
  max-width: 600px;  
  margin: 50px auto;  
  background: white;  
  border-radius: 8px;  
  box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);  
}  
  
.title {  
  text-align: center;  
  font-size: 24px;  
  margin-bottom: 20px;  
  color: #333;  
}  
  
.task-list {  
  list-style: none;  
  padding: 0;  
}  
  
.task-item {  
  display: flex;
```

```
justify-content: space-between;
align-items: center;
padding: 10px 15px;
margin: 10px 0;
border: 1px solid #ddd;
border-radius: 4px;
background: #fff;
transition: background 0.3s;
}
.task-item:hover {
  background: #f5f5f5;
}
.task-name {
  font-weight: bold;
}
.task-name.done {
  text-decoration: line-through;
  color: green;
}
.task-status {
  margin-right: auto;
  margin-left: 10px;
  font-size: 0.9rem;
  color: #666;
}
.toggle-button {
  background-color: #007bff;
```

```
color: white;
border: none;
padding: 6px 12px;
border-radius: 4px;
cursor: pointer;
font-size: 0.9rem;
transition: background-color 0.3s;
}

.toggle-button:hover {
background-color: #0056b3;
}
```

```
task-management > src > # index.css > .container
1  body {
2    font-family: 'Arial', sans-serif;
3    background-color: #f9f9f9;
4    margin: 0;
5    padding: 0;
6  }
7
8  .container {
9    padding: 20px;
10   max-width: 600px;
11   margin: 50px auto;
12   background: white;
13   border-radius: 8px;
14   box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);
15 }
16
17 .title {
18   text-align: center;
19   font-size: 24px;
20   margin-bottom: 20px;
21   color: #333;
22 }
23
24 .task-list {
25   list-style: none;
26   padding: 0;
27 }
28
```

task-management > src > # index.css > .container

```
29 .task-item {
30   display: flex;
31   justify-content: space-between;
32   align-items: center;
33   padding: 10px 15px;
34   margin: 10px 0;
35   border: 1px solid #ddd;
36   border-radius: 4px;
37   background: #fff;
38   transition: background 0.3s;
39 }
40
41 .task-item:hover {
42   background: #f5f5f5;
43 }
44
45 .task-name {
46   font-weight: bold;
47 }
48
49 .task-name.done {
50   text-decoration: line-through;
51   color: green;
52 }
53
54 .task-status {
55   margin-right: auto;
```

task-management > src > # index.css > .container

```
53
54 .task-status {
55   margin-right: auto;
56   margin-left: 10px;
57   font-size: 0.9rem;
58   color: #666;
59 }
60
61 .toggle-button {
62   background-color: #007bff;
63   color: white;
64   border: none;
65   padding: 6px 12px;
66   border-radius: 4px;
67   cursor: pointer;
68   font-size: 0.9rem;
69   transition: background-color 0.3s;
70 }
71
72 .toggle-button:hover {
73   background-color: #0056b3;
74 }
75
```

TaskList.js

```
import React, { useState } from "react";

const TaskList = () => {

  const [tasks, setTasks] = useState([

    { name: "Task 1", status: "done" },

    { name: "Task 2", status: "not done" },

    { name: "Task 3", status: "done" },

  ]);


  // Function to toggle task status
  const toggleStatus = (index) => {

    setTasks((prevTasks) =>

      prevTasks.map((task, i) =>

        i === index

          ? { ...task, status: task.status === "done" ? "not done" : "done" }

          : task

        )

      );

  };


  return (

    <div className="container">

      <h1 className="title">Employee Task Management</h1>

      <ul className="task-list">

        {tasks.map((task, index) => (

          <li key={index} className="task-item">

            <span className={`task-name ${task.status === "done" ? "done" :

              ""}`}>
```

```

        {task.name}
      </span>

      <span className="task-status">({task.status})</span>

      <button className="toggle-button" onClick={() =>
toggleStatus(index)}>

        Toggle Status

      </button>

    </li>

  )))}

</ul>

</div>

);

};

```

```
export default TaskList;
```

```

task-management > src > JS TaskList.js > TaskList
1  import React, { useState } from "react";
2
3  const TaskList = () => {
4    const [tasks, setTasks] = useState([
5      { name: "Task 1", status: "done" },
6      { name: "Task 2", status: "not done" },
7      { name: "Task 3", status: "done" },
8    ]);
9
10   // Function to toggle task status
11   const toggleStatus = (index) => {
12     setTasks((prevTasks) =>
13       prevTasks.map((task, i) =>
14         i === index
15           ? { ...task, status: task.status === "done" ? "not done" : "done" }
16           : task
17       )
18     );
19   };
20
21   return (
22     <div className="container">
23       <h1 className="title">Employee Task Management</h1>
24       <ul className="task-list">
25         {tasks.map((task, index) => (
26           <li key={index} className="task-item">
27             <span className={`task-name ${task.status === "done" ? "done" : ""}`}>
28               {task.name}

```



```

21     return (
22       <div className="container">
23         <h1 className="title">Employee Task Management</h1>
24         <ul className="task-list">
25           {tasks.map((task, index) => (
26             <li key={index} className="task-item">
27               <span className={`task-name ${task.status === "done" ? "done" : ""}`}>
28                 {task.name}
29               </span>
30               <span className="task-status">{task.status}</span>
31               <button className="toggle-button" onClick={() => toggleStatus(index)}>
32                 Toggle Status
33               </button>
34             </li>
35           ))}
36         </ul>
37       </div>
38     );
39   };
40
41   export default TaskList;
42

```

OUTPUT –

