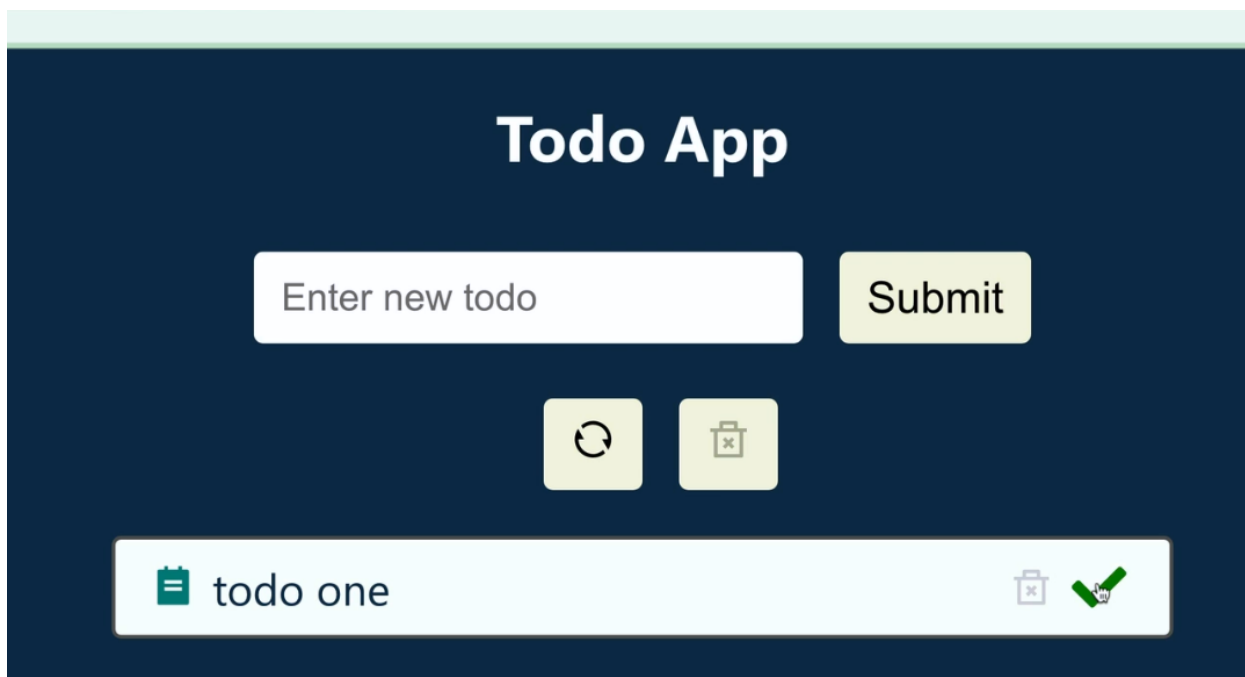


Проект Todo App v2



Изменение структуры данных на массив объектов

Структура массива для хранения задач в данном случае не подходит, т.к. теперь рядом с текстом задачи будут находиться поля `isCompleted` и `id`. `isCompleted` будет хранить информацию (`true/false`) о выполнении задачи, а `id` уникальный номер задачи. Поэтому организуем объект для хранения этих свойств.

Для `id` уникальность зададим с помощью внешнего npm-пакета, который и обеспечит уникальность

<https://www.npmjs.com/package/uuid?activeTab=readme>

(!!!Предварительно остановить сервер)

```
npm i uuid
```

Запустим сервер и подключим библиотеку

```
import { v4 as uuidv4 } from 'uuid';
```

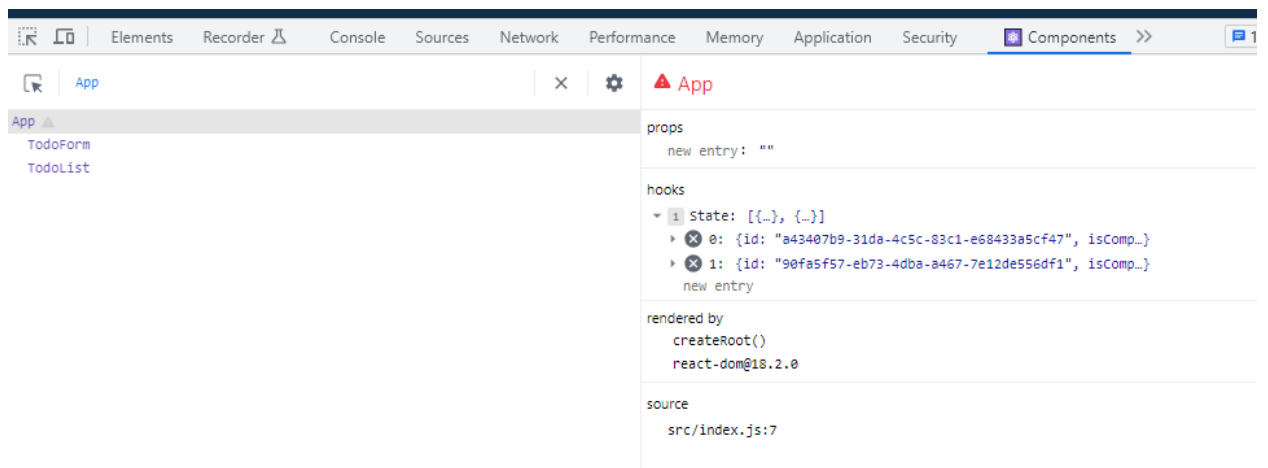
Поскольку у нас появился объект с идентификатор `id`, то предыдущий идентификатор `index` необходимо заменить во всех компонентах проекта

```
JS App.js U X JS TodoList.js U JS Todo.js U
25-todo-app-v2 > src > JS App.js > App > addTodoHandler > newTodo > id
6
7 function App() {
8   const [todos, setTodos] = useState([]);
9
10  const addTodoHandler = (text) => {
11    const newTodo = {
12      text: text,
13      isCompleted: false,
14      id: uuidv4(),
15    };
16
17    setTodos([...todos, newTodo]);
18  };
19
20  const deleteTodoHandler = (id) => {
21    setTodos(todos.filter((todo) => todo.id !== id));
22  };
23
24  return (
25    <div className="App">
26      <h1>Todo App</h1>
27      <TodoForm addTodo={addTodoHandler} />
28      <TodoList todos={todos} deleteTodo={deleteTodoHandler} />
29    </div>
30  );
31 }
32
33 export default App;
```

```
JS App.js U JS TodoList.js U JS Todo.js U X
25-todo-app-v2 > src > components > Todos > JS Todo.js > ...
1 import { RiTodoFill } from 'react-icons/ri';
2 import styles from './Todo.module.css';
3
4 function Todo({ todo, deleteTodo }) {
5   return (
6     <div className={styles.todo} onDoubleClick={() => deleteTodo(todo.id)}>
7       <RiTodoFill className={styles.todoIcon} />
8       <div className={styles.todoText}>{todo.text}</div>
9     </div>
10  );
11 }
12
13 export default Todo;
14
```

```
JS App.js U JS TodoList.js U X JS Todo.js U
25-todo-app-v2 > src > components > Todos > JS TodoList.js > TodoList > todos.map() callba
1  import Todo from './Todo';
2  import styles from './TodoList.module.css';
3
4  function TodoList({ todos, deleteTodo }) {
5    return (
6      <div className={styles.todoListContainer}>
7        {!todos.length && <h2>Todo list is empty</h2>}
8        {todos.map((todo) => (
9          <Todo key={todo.id} todo={todo} deleteTodo={deleteTodo} />
10       ))}
11      </div>
12    );
13  }
14
15  export default TodoList;
16
```

Так же обратим внимание на состояние компонента App.js на вкладке Components



Добавление и стилизация кнопок удаления и завершения задач Todo.module

```
JS App.js U JS TodoList.js U JS Todo.js U X Todo.module.css U
25-todo-app-v2 > src > components > Todos > JS Todo.js > Todo
1 import { RiTodoFill } from 'react-icons/ri';
2 import { RiDeleteBin2Line } from 'react-icons/ri';
3 import { FaCheck } from 'react-icons/fa';
4 import styles from './Todo.module.css';
5
6 function Todo({ todo, deleteTodo }) {
7   return (
8     <div className={styles.todo} onDoubleClick={() => deleteTodo(todo.id)}>
9       <RiTodoFill className={styles.todoIcon} />
10      <div className={styles.todoText}>{todo.text}</div>
11      <RiDeleteBin2Line className={styles.deleteIcon} />
12      <FaCheck className={styles.checkIcon} />
13    </div>
14  );
15 }
16
17 export default Todo;
```

```

1  .todo {
2      display: flex;
3      align-items: center;
4      padding: 10px 20px;
5      margin: 15px 0;
6      font-size: 1.5rem;
7      border-radius: 5px;
8      border: 2px solid #555;
9      color: #112d49;
10     background-color: #fbfef9;
11 }
12
13 .todoText {
14     width: 100%;
15     text-align: left;
16 }
17
18 .todoIcon {
19     font-size: 1.8rem;
20     margin-right: 10px;
21     color: teal;
22 }
23
24 .completedTodo {
25     background-color: unset;
26     border-color: gray;
27     color: gray;
28 }
29
30 .todo.completedTodo .todoIcon,
31 .todo.completedTodo .checkIcon,
32 .todo.completedTodo .deleteIcon {
33     color: gray;
34 }
35
36 .checkIcon,
37 .deleteIcon {
38     cursor: pointer;
39     color: lightgrey;
40     padding: 0 7px;
41     font-size: 1.8rem;
42     transition: transform 0.3s;
43 }
44
45 .checkIcon:hover,
46 .deleteIcon:hover {
47     cursor: pointer;
48     transform: scale(1.3);
49 }
50
51 .checkIcon:hover {
52     color: green;
53 }
54
55 .deleteIcon:hover {
56     color: red;
57 }
58

```

Функционал удаления отдельных задач

Т.к. функционал удаления задачи у нас уже реализован, то скопируем эту функцию и добавим ее через свойство `onClick` в компонент `RiDeleteBin2Line`, а из общего `<div>` удалим свойство `onDubleClick`

```
JS Todo.js U X JS App.js U JS TodoList.js U Todo.module.css U
25-todo-app-v2 > src > components > Todos > JS Todo.js > [e] default
1  import { RiTodoFill } from 'react-icons/ri';
2  import { RiDeleteBin2Line } from 'react-icons/ri';
3  import { FaCheck } from 'react-icons/fa';
4  import styles from './Todo.module.css';
5
6  function Todo({ todo, deleteTodo }) {
7    return (
8      <div className={styles.todo}>
9        <RiTodoFill className={styles.todoIcon} />
10       <div className={styles.todoText}>{todo.text}</div>
11       <RiDeleteBin2Line
12         className={styles.deleteIcon}
13         onClick={() => deleteTodo(todo.id)}
14       />
15       <FaCheck className={styles.checkIcon} />
16     </div>
17   );
18 }
19
20 export default Todo;
```