

# TodoList Component

In this lesson, we develop a component to display the list of ToDo items.

## WE'LL COVER THE FOLLOWING ^

- TodoList
- Next

## TodoList #

Here is the code for listing ToDo items.

```
import React from 'react';
import { useDispatch, useTrackedState } from './store';

import TodoItem from './TodoItem';
import NewTodo from './NewTodo';

const TodoList = () => {
  const dispatch = useDispatch();
  const state = useTrackedState();
  const setQuery = event => {
    dispatch({ type: 'SET_QUERY', query: event.target.value });
  };
  return (
    <div>
      <ul>
        {state.todos.map(({ id, title, completed }) => (
          <TodoItem key={id} id={id} title={title} completed={completed} />
        ))}
      <NewTodo />
    </ul>
    <div>
      Highlight Query for incomplete items:
      <input value={state.query} onChange={setQuery} />
    </div>
  )
}
```

```
    </div>
  );
};

export default TodoList;
```

The `useDispatch` returns a function `dispatch`. When we dispatch an action it will invoke the reducer.

The `useTrackedState` returns a global state.

Notice this signature is exactly the same as the usage of `useReducer`.

```
const [state, dispatch] = useReducer(...);
```

We have another hook in React Tracked that can be used like this.

```
const [state, dispatch] = useTracked();
```

There are two components this imports from other files, `TodoItem` and `NewTodo` which we will cover later.

## Next #

In the next lesson, we create a `TodoItem` component.