08. REDUX Example Todo App.

This is the complete source code of the tiny **todo app** we built during the basics tutorial.

8.1. Entry Point

Example: index.js

```
import React from 'react'
import { render } from 'react-dom'
import { Provider } from 'react-redux'
import { createStore } from 'redux'
import rootReducer from './reducers'
import App from './components/App'

const store = createStore(rootReducer)

render(
  <Provider store={store}>
    <App />
  </Provider>,
  document.getElementById('root')
)
```

8.2. Action Creators

Example: actions/index.js

```
let nextTodoId = 0
export const addTodo = text => ({
  type: 'ADD_TODO',
  id: nextTodoId++,
  text
})

export const setVisibilityFilter = filter => ({
  type: 'SET_VISIBILITY_FILTER',
  filter
})

export const toggleTodo = id => ({
  type: 'TOGGLE_TODO',
  id
})

export const VisibilityFilters = {
  SHOW_ALL: 'SHOW_ALL',
  SHOW_COMPLETED: 'SHOW_COMPLETED',
  SHOW_ACTIVE: 'SHOW_ACTIVE'
}
```

8.3. Reducers

Example: reducers/todo.js

```
const todos = (state = [], action) => {
  switch (action.type) {
    case 'ADD_TODO':
      return [
        ...state,
        {
          id: action.id,
          text: action.text,
          completed: false
        }
      ]
    case 'TOGGLE_TODO':
      return state.map(todo =>
        todo.id === action.id ? { ...todo, completed: !
todo.completed } : todo
      )
    default:
      return state
  }
}

export default todos
```

Example: reducers/visibilityFilter.js

```
import { VisibilityFilters } from '../actions'

const visibilityFilter = (state = VisibilityFilters.SHOW_ALL,
action) => {
  switch (action.type) {
    case 'SET_VISIBILITY_FILTER':
      return action.filter
    default:
      return state
  }
}

export default visibilityFilter
```

Example: reducers/index.js

```
import { combineReducers } from 'redux'
import todos from './todos'
import visibilityFilter from './visibilityFilter'

export default combineReducers({
  todos,
  visibilityFilter
})
```

8.4. Presentational Components

Example: components/Todo.js

```
import React from 'react'
import PropTypes from 'prop-types'

const Todo = ({ onClick, completed, text }) => (
  <li
    onClick={onClick}
    style={{
      textDecoration: completed ? 'line-through' : 'none'
    }}
  >
    {text}
  </li>
)

Todo.propTypes = {
  onClick: PropTypes.func.isRequired,
  completed: PropTypes.bool.isRequired,
  text: PropTypes.string.isRequired
}
```

```
export default Todo
```

Example: components/TodoList.js

```
import React from 'react'
import PropTypes from 'prop-types'
import Todo from './Todo'

const TodoList = ({ todos, toggleTodo }) => (
  <ul>
    {todos.map(todo => (
      <Todo key={todo.id} {...todo} onClick={() =>
toggleTodo(todo.id)} />
    ))}
  </ul>
)

TodoList.propTypes = {
  todos: PropTypes.arrayOf(
    PropTypes.shape({
      id: PropTypes.number.isRequired,
      completed: PropTypes.bool.isRequired,
      text: PropTypes.string.isRequired
    }).isRequired
  ).isRequired,
  toggleTodo: PropTypes.func.isRequired
}

export default TodoList
```

Example: components/Link.js

```
import React from 'react'
import PropTypes from 'prop-types'

const Link = ({ active, children, onClick }) => (
  <button
    onClick={onClick}
    disabled={active}
    style={{
      marginLeft: '4px'
    }}
  >
    {children}
  </button>
)
```

```
Link.propTypes = {
  active: PropTypes.bool.isRequired,
  children: PropTypes.node.isRequired,
  onClick: PropTypes.func.isRequired
}

export default Link
```

Example: components/Footer.js

```
import React from 'react'
import FilterLink from '../containers/FilterLink'
import { VisibilityFilters } from '../actions'

const Footer = () => (
  <div>
    <span>Show: </span>
    <FilterLink filter={VisibilityFilters.SHOW_ALL}>All</FilterLink>
    <FilterLink
filter={VisibilityFilters.SHOW_ACTIVE}>Active</FilterLink>
    <FilterLink filter={VisibilityFilters.SHOW_COMPLETED}>Completed</
FilterLink>
  </div>
)

export default Footer
```

Example: components/App.js

```
import React from 'react'
import Footer from './Footer'
import AddTodo from '../containers/AddTodo'
import VisibleTodoList from '../containers/VisibleTodoList'

const App = () => (
  <div>
    <AddTodo />
    <VisibleTodoList />
    <Footer />
  </div>
)

export default App
```

8.5. Container Components.

Example: containers/VisibleTodoList.js

```
import { connect } from 'react-redux'
import { toggleTodo } from '../actions'
import TodoList from '../components/TodoList'
import { VisibilityFilters } from '../actions'

const getVisibleTodos = (todos, filter) => {
  switch (filter) {
    case VisibilityFilters.SHOW_ALL:
      return todos
    case VisibilityFilters.SHOW_COMPLETED:
      return todos.filter(t => t.completed)
    case VisibilityFilters.SHOW_ACTIVE:
      return todos.filter(t => !t.completed)
    default:
      throw new Error('Unknown filter: ' + filter)
  }
}

const mapStateToProps = state => ({
  todos: getVisibleTodos(state.todos, state.visibilityFilter)
})

const mapDispatchToProps = dispatch => ({
  toggleTodo: id => dispatch(toggleTodo(id))
})

export default connect(
  mapStateToProps,
  mapDispatchToProps
)(TodoList)
```

Example: containers/FilterLink.js

```
import { connect } from 'react-redux'
import { setVisibilityFilter } from '../actions'
import Link from '../components/Link'

const mapStateToProps = (state, ownProps) => ({
  active: ownProps.filter === state.visibilityFilter
})

const mapDispatchToProps = (dispatch, ownProps) => ({
  onClick: () => dispatch(setVisibilityFilter(ownProps.filter))
})
```

```
export default connect(
  mapStateToProps,
  mapDispatchToProps
)(Link)
```

8.6. Other Components

Example: containers/AddTodo.js

```
import React from 'react'
import { connect } from 'react-redux'
import { addTodo } from '../actions'

const AddTodo = ({ dispatch }) => {
  let input

  return (
    <div>
      <form
        onSubmit={e => {
          e.preventDefault()
          if (!input.value.trim()) {
            return
          }
          dispatch(addTodo(input.value))
          input.value = ''
        }}
      >
        <input ref={node => (input = node)} />
        <button type="submit">Add Todo</button>
      </form>
    </div>
  )
}

export default connect()(AddTodo)
```

This code is also in our repository of examples and can be run in your browser via CodeSandbox.

- https://github.com/reduxjs/redux/tree/master/examples/todos/src
- https://codesandbox.io/s/github/reduxjs/redux/tree/master/examples/todos