

TUGAS INDIVIDU
Redux
Pemrograman Berbasis Framework

Oleh:

Alfalah Anugara

1741720159

TI-3A / 02



PROGRAM STUDI – D4 TEKNIK INFORMATIKA

JURUSAN – TEKNOLOGI INFORMASI

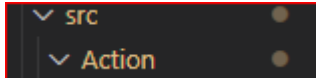
POLITEKNIK NEGERI MALANG

2020/2021

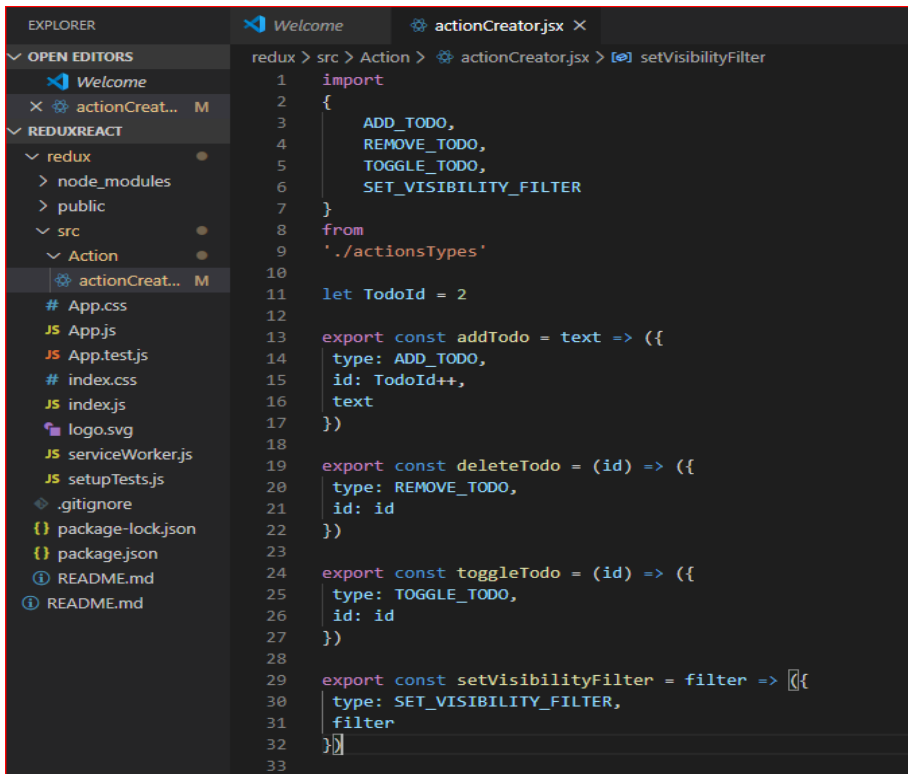
PRAKTIKUM REDUX

Step :

- 1) Buat Folder dalam src yaitu dengan nama Action.

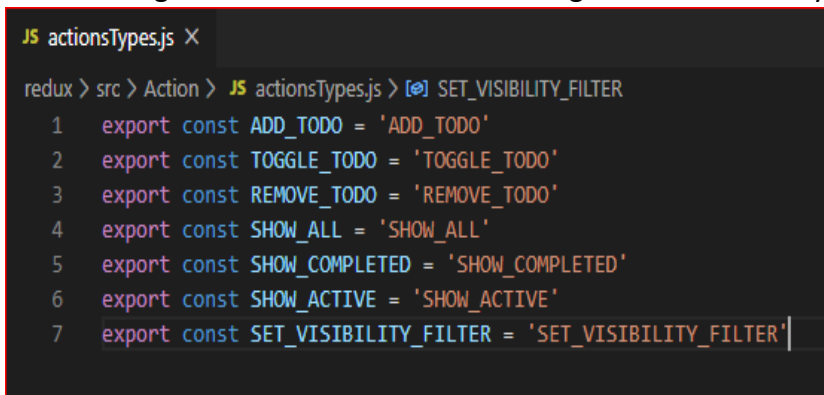


- 2) Lalu buat file dengan nama ActionCreator.jsx dan isi syntax seperti gambar dibawah ini



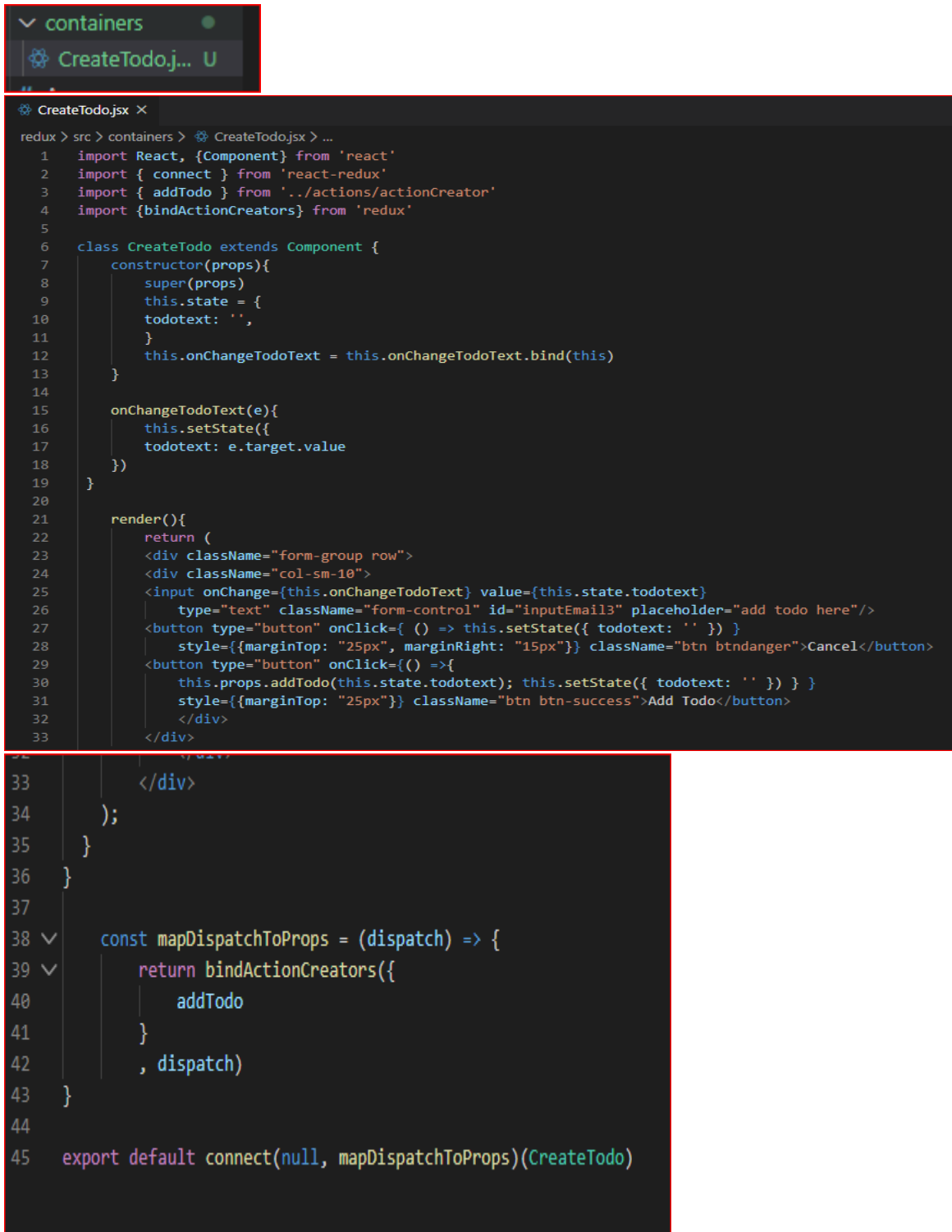
```
1 import
2 {
3   ADD_TODO,
4   REMOVE_TODO,
5   TOGGLE_TODO,
6   SET_VISIBILITY_FILTER
7 }
8 from
9 './actionsTypes'
10
11 let TodoId = 2
12
13 export const addTodo = text => ({
14   type: ADD_TODO,
15   id: TodoId++,
16   text
17 })
18
19 export const deleteTodo = (id) => ({
20   type: REMOVE_TODO,
21   id: id
22 })
23
24 export const toggleTodo = (id) => ({
25   type: TOGGLE_TODO,
26   id: id
27 })
28
29 export const setVisibilityFilter = filter => ({
30   type: SET_VISIBILITY_FILTER,
31   filter
32 })
33
```

- 3) Lalu buat lagi file didalam folder Action dengan nama actionsTypes.js



```
1 export const ADD_TODO = 'ADD_TODO'
2 export const TOGGLE_TODO = 'TOGGLE_TODO'
3 export const REMOVE_TODO = 'REMOVE_TODO'
4 export const SHOW_ALL = 'SHOW_ALL'
5 export const SHOW_COMPLETED = 'SHOW_COMPLETED'
6 export const SHOW_ACTIVE = 'SHOW_ACTIVE'
7 export const SET_VISIBILITY_FILTER = 'SET_VISIBILITY_FILTER'
```

- 4) Kemudian didalam folder src buatlah lagi folder dengan nama containers dan buat file dengan judul CreateTodo.jsx dan isi Kode program seperti dibawah ini.



The image shows a code editor interface. At the top, a file explorer sidebar displays a folder named 'containers' which contains a file named 'CreateTodo.j...'. The main editor area shows the code for 'CreateTodo.jsx'. The code is a class-based component that uses React, Redux, and React-Redux. It has a state for 'todotext' and two buttons: 'Cancel' and 'Add Todo'. The 'Add Todo' button calls the 'addTodo' action creator from Redux. The component is wrapped in a 'connect' function from 'react-redux'.

```
1 import React, {Component} from 'react'
2 import { connect } from 'react-redux'
3 import { addTodo } from '../actions/actionCreator'
4 import { bindActionCreators } from 'redux'
5
6 class CreateTodo extends Component {
7   constructor(props){
8     super(props)
9     this.state = {
10       todotext: '',
11     }
12     this.onChangeTodoText = this.onChangeTodoText.bind(this)
13   }
14
15   onChangeTodoText(e){
16     this.setState({
17       todotext: e.target.value
18     })
19   }
20
21   render(){
22     return (
23       <div className="form-group row">
24         <div className="col-sm-10">
25           <input onChange={this.onChangeTodoText} value={this.state.todotext}
26             type="text" className="form-control" id="inputEmail3" placeholder="add todo here"/>
27           <button type="button" onClick={ () => this.setState({ todotext: '' }) }
28             style={{marginTop: "25px", marginRight: "15px"}} className="btn btn-danger">Cancel</button>
29           <button type="button" onClick={() =>{
30             this.props.addTodo(this.state.todotext); this.setState({ todotext: '' }) } }
31             style={{marginTop: "25px"}} className="btn btn-success">Add Todo</button>
32         </div>
33       </div>
34     );
35   }
36 }
37
38 const mapDispatchToProps = (dispatch) => {
39   return bindActionCreators({
40     addTodo
41   }, dispatch)
42 }
43
44 export default connect(null, mapDispatchToProps)(CreateTodo)
```

5) Masih didalam folder containers, buatlah file dengan nama Table.js

```
JS Tablejs X
redux > src > containers > JS Tablejs > Table > render > props.todos.map() callback > fontSize
1 import React, { Component } from "react";
2 import { connect } from "react-redux";
3 import {
4   deleteTodo,
5   toggleTodo,
6   setVisibilityFilter
7 } from "../actions/actionCreator";
8 import { SHOW_ALL, SHOW_COMPLETED, SHOW_ACTIVE } from "../actions/actionTypes";
9 import { bindActionCreators } from "redux";
10
11 class Table extends Component {
12   render() {
13     return (
14       <div className="col-lg-10 offset-lg-1 col-md-10 col-sm-12 col-xs-12">
15         <nav style={{ marginTop: "60px" }}>
16           <ol className="breadcrumb">
17             <li
18               className={"breadcrumb-item " + (this.props.visibilityFilter === SHOW_ALL ?
19                 'active' : '') }
20               onClick={() => this.props.setVisibilityFilter(SHOW_ALL)}
21             >
22               All
23             </li>
24             <li
25               className={"breadcrumb-item " + (this.props.visibilityFilter === SHOW_COMPLETED ? 'active' : '') }
26               onClick={() => this.props.setVisibilityFilter(SHOW_COMPLETED)}
27             >
28               Completed
29             </li>
30             <li
31               className={"breadcrumb-item " + (this.props.visibilityFilter === SHOW_ACTIVE ?
32                 'active' : '') }
33               onClick={() => this.props.setVisibilityFilter(SHOW_ACTIVE)}
34             >
35               Active
36             </li>
37           </ol>
38         </nav>
39
40         {this.props.todos.length !== 0 ? (
41           <table
42             style={{ marginTop: "60px" }}
43             className="table table-hover table-dark">
44             <thead>
45               <tr>
46                 <th scope="col">Todos</th>
47                 <th scope="col">Actions</th>
48               </tr>
49             </thead>
50             <tbody>
51               {this.props.todos.map(todo => (
52                 <tr key={todo.id}>
53                   <td
54                     style={{
55                       textDecoration: todo.completed ? "line-through" : "none"
56                     }}
57                   >{todo.text} {todo.completed === true ? "(completed)" : ""}
58                 </td>
59                 <td>
60                   <span
61                     className="fas fa-minus-circle"
62                     onClick={() => this.props.deleteTodo(todo.id)}
63                     style={{
64                       color: "white",
65                       fontSize: "20pt",
66                       marginRight: "20px"
67                     }}
68                   />
69                   <span
70                     className="fas fa-check-circle"
71                     onClick={() => this.props.toggleTodo(todo.id)}
72                     style={{ color: "white", fontSize: "20pt" }}
73                   />
74                 </td>
75               </tr>
76             </tbody>
77           </table>
78         ) : ""}
79       </div>
80     );
81   }
82 }
```

```
JS Table.js X
redux > src > containers > JS Table.js > Table > render > props.todos.map() callback
75     </tr>
76   }}
77 </tbody>
78 </table>
79 ) : (
80
81   <div
82     style={{ marginTop: "50px" }}
83     className="col-lg-10 col-md-10 col-xs-12 col-sm-12 offset-lg-1">
84     <div className="alert alert-danger" role="alert">
85       Todo List is empty or Filter results show no results
86     </div>
87   </div>
88   ){ " "}
89 </div>
90 );
91 }
92 }
93 const getVisibleTodos = (todos, filter) => {
94   switch (filter) {
95     case SHOW_ALL:
96       return todos;
97
98     case SHOW_COMPLETED:
99       return todos.filter(t => t.completed);
100
101     case SHOW_ACTIVE:
102       return todos.filter(t => !t.completed);
103     default:
104       throw new Error("Unknown filter: " + filter);
105   }
106 };
107
108 const mapStateToProps = state => {
109   return { todos: getVisibleTodos(state.todos, state.visibilityFilter),
110     visibilityFilter: state.visibilityFilter
111   };
112 }
```

```
JS Table.js X
redux > src > containers > JS Table.js > getVisibleTodos
112   };
113
114   const mapDispatchToProps = dispatch => {
115     return bindActionCreators(
116       {
117         deleteTodo,
118         toggleTodo,
119         setVisibilityFilter
120       },
121       dispatch
122     );
123   };
124
125   export default connect(mapStateToProps, mapDispatchToProps)(Table);
```

6) Edit index.js dengan source code seperti berikut.

```
JS Table.js JS index.js X
redux > src > JS index.js > ...
1 import React from 'react';
2 import ReactDOM from 'react-dom';
3 import { Provider } from 'react-redux'
4 import { createStore, applyMiddleware, compose } from 'redux'
5 import MainReducer from '../reducers/MainReducer'
6 import App from './App';
7 import registerServiceWorker from './registerServiceWorker';
8 import './index.css';
9 import 'bootstrap/dist/css/bootstrap.min.css'
10
11 const store = compose(window.devToolsExtension ? window.devToolsExtension()
12 f)(createStore)(MainReducer)
13 |
14 ReactDOM.render(<Provider store={store}>
15 <App />
16 </Provider>
17 , document.getElementById('root'));
18 registerServiceWorker();
19
20 // If you want your app to work offline and load faster, you can change
21 // unregister() to register() below. Note this comes with some pitfalls.
22 // Learn more about service workers: https://bit.ly/CRA-PWA
23 serviceWorker.unregister();
24
```

7) Buatlah folder dalam src dengan nama reducers dan beri nama file FilterReducer.js

```
JS Table.js X JS index.js JS FilterReducer.js X
redux > src > JS FilterReducer.js > ...
1 import {
2   SHOW_ALL,
3   SET_VISIBILITY_FILTER
4 } from '../actions/actionTypes';
5
6 const visibilityFilter = (state = SHOW_ALL, action) => {
7   switch (action.type) {
8     case SET_VISIBILITY_FILTER:
9       return action.filter;
10     default:
11       return state;
12   }
13 };
14 |
15 export default visibilityFilter;
16
```

- 8) Masih didalam folder reducers, buatlah file dengan nama TodoReducer.js

```
JS TodoReducer.js X
redux > src > JS TodoReducer.js > [0] TodoReducer
12 import {
13   ADD_TODO,
14   REMOVE_TODO,
15   TOGGLE_TODO
16 } from '../actions/actionTypes'
17
18 const INITIAL_DATA = []
19
20 const TodoReducer = (state=INITIAL_DATA, action) => {
21   switch (action.type){
22     case ADD_TODO:
23       return [
24         ...state,{
25           id: action.id,
26           text: action.text,
27           completed: false,
28         }
29       ]
30
31     case TOGGLE_TODO:
32       return state.map(todo =>
33         (todo.id === action.id)
34         ? {...todo, completed: !todo.completed}
35         : todo
36       )
37
38     case REMOVE_TODO:
39       const numIndex = parseInt(action.id)
40       return state.filter(todo => todo.id !== numIndex);
41       default:
42         return state
43     }
44 }
45 export default TodoReducer
```

- 9) Buat file dengan nama MainReducer.js kemudian simpan

```
redux > src > reducers > JS MainReducer.js > ...
1 import { combineReducers } from 'redux'
2 import todos from './TodoReducer'
3 import visibilityFilter from './FilterReducer'
4
5 export default combineReducers({
6   todos,
7   visibilityFilter
8 })
```

- 10) Running semua program dengan npm start dan lihat hasilnya

```
Compiled successfully!

You can now view redux-falah in the browser.

Local:      http://localhost:3000
On Your Network:  http://192.168.56.1:3000

Note that the development build is not optimized.
To create a production build, use npm run build.
```

11) Hasil Output Running program

