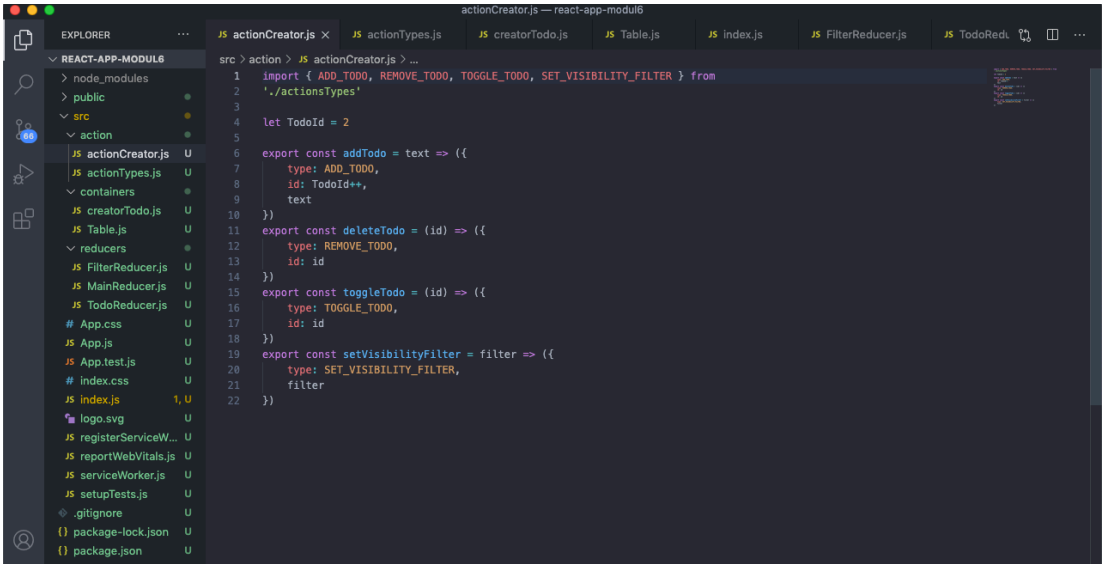
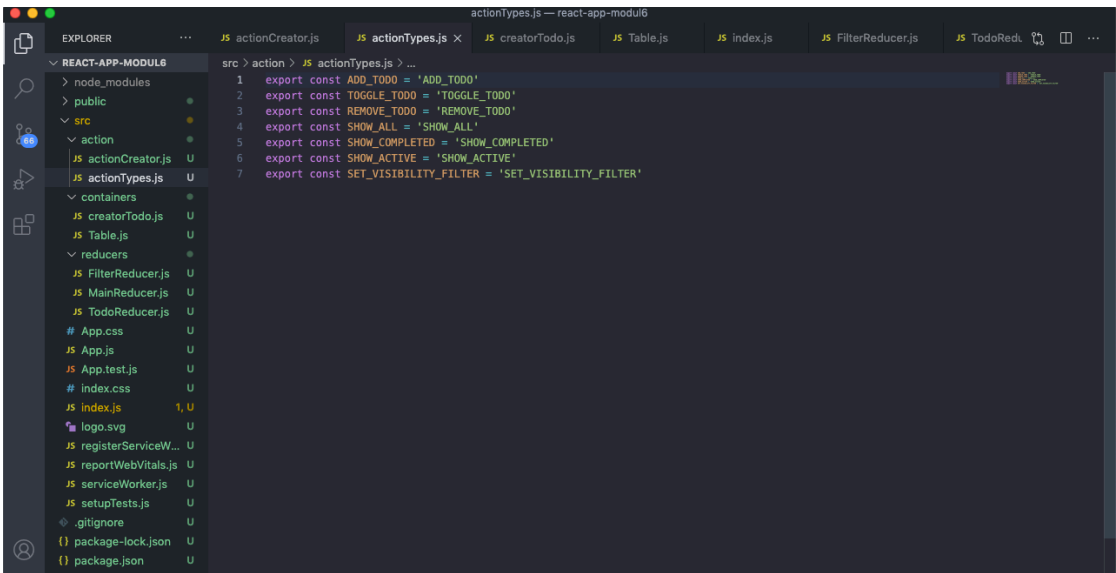




Jurusan Teknologi Informasi Politeknik Negeri Malang
Tugas Minggu: Modul 6
Mata Kuliah Pemrograman Berbasis Framework
Pengampu:
Februari 2021

Nama : Muhammad Iqbaluddin Al Huda
Kelas : TI-3E
NIM : 1841720013

Langkah	Keterangan
1	<p>Praktikum :</p> <ol style="list-style-type: none">1. Buat Folder dalam src yaitu Action2. Beri nama actionCreator 
2	<p>Masih dalam Folder Action buatlah file actionTypes.js</p> 

Kemudian dalam folder src buatlah folder dengan nama containers dan buatlah file dengan judul CreateTodo

3

The screenshot shows the VS Code interface with the Explorer sidebar on the left. The 'src' folder is expanded, showing a new folder named 'containers' and a new file named 'creatorTodo.js'. The main editor displays the code for 'creatorTodo.js'. The code imports React, connect from 'react-redux', and addTodo and bindActionCreators from 'redux'. It defines a class 'CreateTodo' extending 'Component'. The constructor sets the initial state to { todotext: ''}. The onChangeTodoText method updates the state. The render method returns a JSX element with a form group containing an input field and two buttons: 'Cancel' and 'Add Todo'. The 'Add Todo' button has a success class and calls the addTodo action creator. The mapDispatchToProps function is defined to bind the action creators to the dispatch prop. The file is exported as the default export.

```
1 import React, { Component } from "react";
2 import { connect } from "react-redux";
3 import { addTodo } from "../action/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   onChangeTodoText(e) {
15     this.setState({
16       todotext: e.target.value,
17     });
18   }
19   render() {
20     return (
21       (JSX attribute) React.HTMLAttributes<HTMLDivElement>.className?: string
22       <div className="form-group row">
23         <input onChange={this.onChangeTodoText} value={this.state.todotext} type="text" className="form-control" id="
24           style={{
25             marginTop: "25px",
26             marginRight: "15px",
27           }}
28         <button type="button" className="btn btn-danger" > Cancel </button>
29         <button type="button"
30           onClick={() => {
31             this.props.addTodo(this.state.todotext);
32             this.setState({
33               todotext: "",
34             });
35           }}
36         </button>
37       </div>
38     );
39   }
40 }
41
42 const mapDispatchToProps = (dispatch) => {
43   return bindActionCreators(
44     {
45       addTodo,
46     },
47     dispatch
48   );
49 };
50
51 export default connect(null, mapDispatchToProps)(CreateTodo);
```

4

Masih dalam folder containers buatlah file dengan judul Table.js

```
src > containers > JS Table.js > ...
1 import React, { Component } from "react";
2 import { connect } from "react-redux";
3 import {
4   deleteTodo,
5   toggleTodo,
6   setVisibilityFilter,
7 } from "../action/actionCreator";
8 import { SHOW_ALL, SHOW_COMPLETED, SHOW_ACTIVE } from "../Action/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={
19                 "breadcrumb-item " +
20                 (this.props.visibilityFilter === SHOW_ALL ? "active" : "")
21               }
22             >
23               <a href="#">All</a>
24             </li>
25             <li
26               className={
27                 "breadcrumb-item " +
28                 (this.props.visibilityFilter === SHOW_COMPLETED ? "active" : "")
29               }
30             >
31               <a href="#">Completed</a>
32             </li>
33           </ol>
34         </div>
35       </div>
36     );
37   }
38 }
39
40 export default connect(mapStateToProps, mapDispatchToProps)(Table);
```

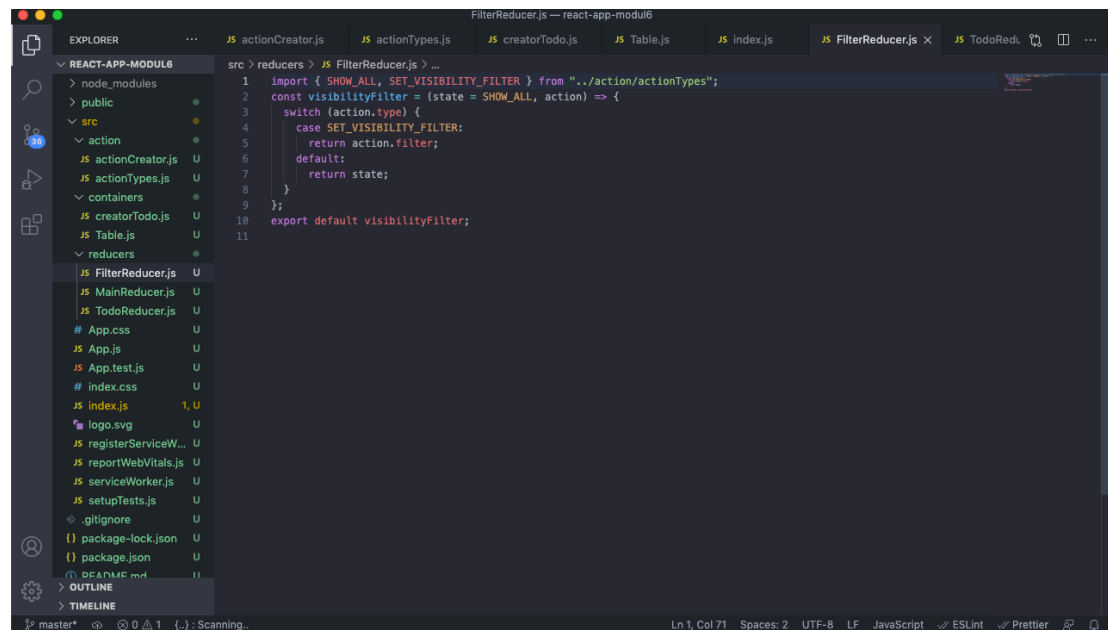
Edit index.js dengan source code berikut ini

```
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 { register as registerServiceWorker } from "../registerServiceWorker";
8
9 const store = compose(
10   window.devToolsExtension ? window.devToolsExtension() : f => f
11 )(createStore)(MainReducer);
12 ReactDOM.render(
13   <Provider store={store}>
14     <App />
15   </Provider>,
16   document.getElementById("root")
17 );
18
19 // If you want to start measuring performance in your app, pass a function
20 // to log results (for example: reportWebVitals(console.log))
21 // or send to an analytics endpoint. Learn more: https://bit.ly/CRA-vitals
22 registerServiceWorker();
```

5

Buat folder dalam src dengan nama reducers dan beri nama file FilterReducer.js dan ketikkan program dibawah ini :

6

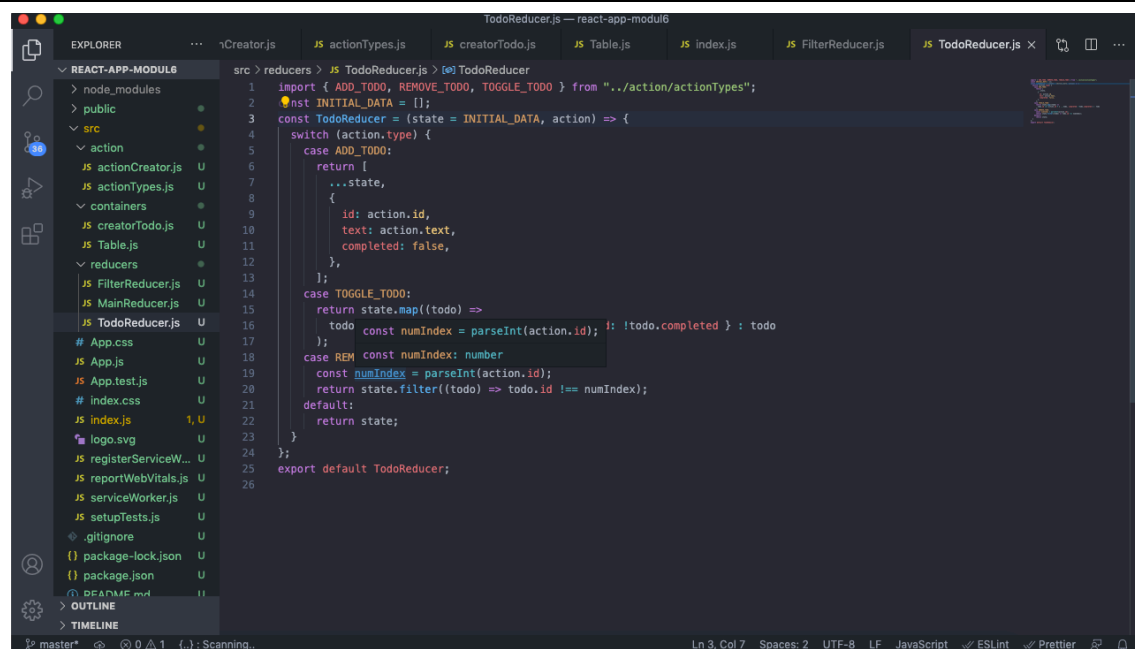


The screenshot shows the VS Code interface with the Explorer sidebar on the left displaying the project structure. The 'reducers' folder is expanded, showing 'FilterReducer.js' as the active file. The main editor displays the following code:

```
1 import { SHOW_ALL, SET_VISIBILITY_FILTER } from "../action/actionTypes";
2 const visibilityFilter = (state = SHOW_ALL, action) => {
3   switch (action.type) {
4     case SET_VISIBILITY_FILTER:
5       return action.filter;
6     default:
7       return state;
8   }
9 };
10 export default visibilityFilter;
```

Masih didalam folder reducers buatlah file dengan nama TodoReducer.js

7

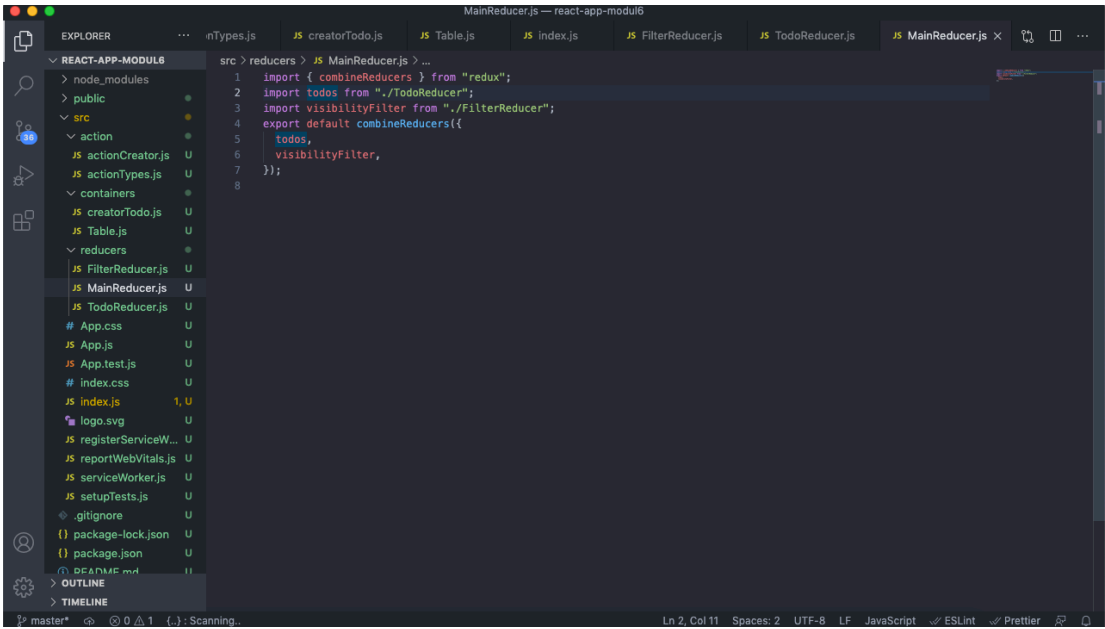
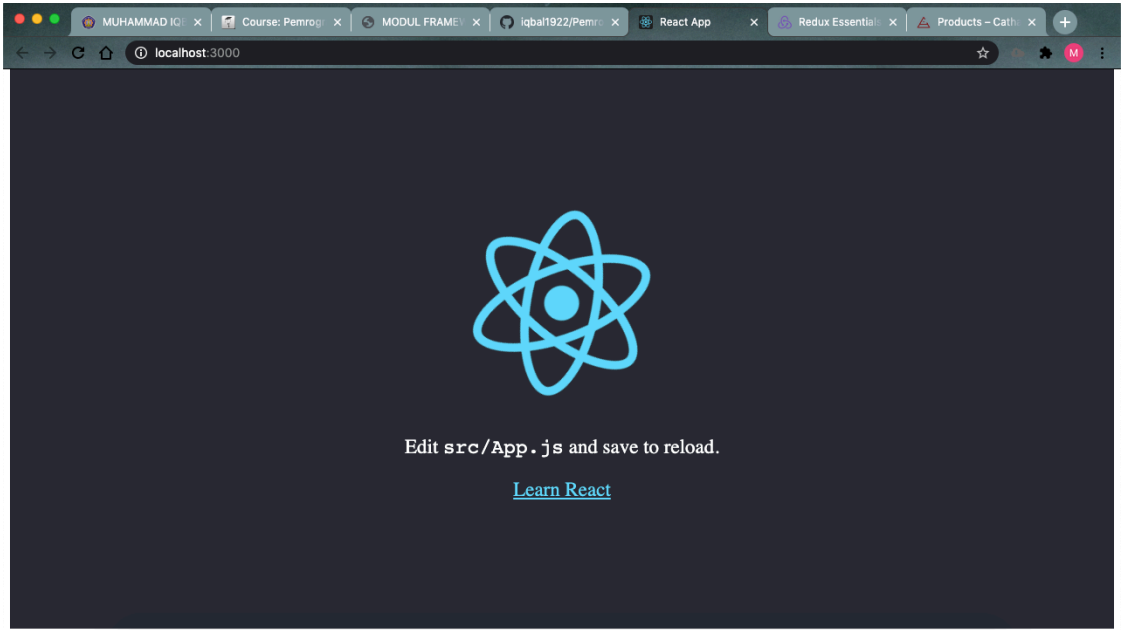


The screenshot shows the VS Code interface with the Explorer sidebar on the left. The 'reducers' folder is expanded, and 'TodoReducer.js' is the active file. The main editor displays the following code:

```
1 import { ADD_TODO, REMOVE_TODO, TOGGLE_TODO } from "../action/actionTypes";
2 const INITIAL_DATA = [];
3 const TodoReducer = (state = INITIAL_DATA, action) => {
4   switch (action.type) {
5     case ADD_TODO:
6       return [
7         ...state,
8         {
9           id: action.id,
10          text: action.text,
11          completed: false,
12        },
13       ];
14     case TOGGLE_TODO:
15       return state.map((todo) => {
16         const numIndex = parseInt(action.id);
17         return { ...todo, completed: !todo.completed };
18       });
19     case REMOVE_TODO:
20       const numIndex = parseInt(action.id);
21       return state.filter((todo) => todo.id !== numIndex);
22     default:
23       return state;
24   }
25 };
26 export default TodoReducer;
```

8

Buat file dengan nama MainReducer.js dan simpan

	
9	<p>Running semua program dengan npm start dan liat hasilnya dan laporkan didalam laporan</p> 
10	<p>https://github.com/iqbal1922/Pemrogramman-Berbasis-Framework/tree/master/Minggu%206%20-%20Redux</p>
11	<p>https://youtu.be/Ot0013QrELU</p>

-- Selamat Mengerjakan --