

Penjelasan kode

1. ...src/components/index.js

```
import React from "react";
import { useSelector, useDispatch } from "react-redux";

import {
  incrementNumber,
  decrementNumber,
} from "../store/counter/counter.action";

const Components = () => {
  const number = useSelector(state => state.counter.number);
  const dispatch = useDispatch();

  return (
    <div>
      <h2>Increment and Decrement Number App</h2>
      <button onClick={() => dispatch(incrementNumber(number))}>+</button>
      <h1>{number}</h1>
      <button onClick={() => dispatch(decrementNumber(number))}>-</button>
    </div>
  )
};

export default Components;
```

Mengatur komponen yang akan ditampilkan di layer seperti tulisan, button, dan angka.

2. ...src/store/counter/counter.action.js

```
import { INCREMENT, DECREMENT } from "../counter.action.type";

export const incrementNumber = (number) => {
  return {
    type: INCREMENT,
    data: number + 1,
  };
};

export const decrementNumber = (number) => {
  return {
    type: DECREMENT,
    data: number - 1,
  };
};
```

Mengatur aksi dari increment dan decrement. Dimana increment untuk menambahkan nilai angka dan decrement mengurangi nilai angka.

3. ...src/store/counter/counter.action.type.js

```
export const INCREMENT = "INCREMENT";  
export const DECREMENT = "DECREMENT";
```

Untuk menjelaskan aksi apa saja yang dapat dilakukan oleh program.

4. ...src/store/counter/reducer.js

```
import { INCREMENT, DECREMENT } from "../counter.action.type";  
  
const initialState = {  
  number : 0,  
};  
  
export default (state={...initialState}, action) => {  
  switch (action.type) {  
    case INCREMENT:  
      return {  
        ...state,  
        number: action.data,  
      };  
    case DECREMENT:  
      return {  
        ...state,  
        number: action.data,  
      };  
    default:  
      return state;  
  }  
};
```

Sebagai reducer

5. ...src/store/index.js

```

import { combineReducers, applyMiddleware, createStore } from "redux";
import { composeWithDevTools } from "redux-devtools-extension";
import thunk from "redux-thunk";

import counterReducer from './counter/counter.reducer.js'

const rootReducer = combineReducers ({
  counter: counterReducer,
});

const store = createStore(
  rootReducer,
  composeWithDevTools(applyMiddleware(thunk))
);

export default store;

```

6. ...src/store/App.js

```

import logo from "./logo.svg";
import "./App.css";

import Components from "./components";

function App() {
  return (
    <div className="App">
      <header className="App-header">
        <img src={logo} className="App-logo" alt="logo" />
        <Components />
      </header>
    </div>
  );
}

export default App;

```

7. ...src/store/index.js

```
import React from "react";
import ReactDOM from "react-dom";
import { Provider } from "react-redux";
import "./index.css";
import App from "./App";
//import * as serviceWorker from "/serviceWorker";
import reportWebVitals from './reportWebVitals';

import store from "./store";

ReactDOM.render(
  <Provider store={store}>
    <React.StrictMode>
      <App />
    </React.StrictMode>
  </Provider>,
  document.getElementById("root")
);

// If you want to start measuring performance in your app, pass a function
// to log results (for example: reportWebVitals(console.log))
// or send to an analytics endpoint. Learn more: https://bit.ly/CRA-vitals
reportWebVitals();
```