

## REDUX

```
① README.md ×
① README.md > # React Router > ## Redux
10
11 ## Redux
12
13 <https://redux.js.org/introduction/getting-started> (installasi)
14
15 Why redux
16 <https://redux.js.org/understanding/thinking-in-redux/motivation>
17
18 3 Prinsip utama redux
19 <https://redux.js.org/understanding/thinking-in-redux/three-principles>
20
21 Redux Flow
22 <https://redux.js.org/assets/images/ReduxDataFlowDiagram-49fa8c3968371d9ef6f2a1486bd40a26.gif>
23
24 ## Basic Example
25
26 <https://redux.js.org/introduction/getting-started>
27
28 ### React Redux
29
30 <https://react-redux.js.org/api/provider>
31
32 ### Kamus
33
34 - Reducer
35 - Pure Function
36 - Action
37 - Action Type
38 - Action Creator
39
```

## PUSAT DATA = STORES

- mkdir stores
- npm install redux

### Basic Example

The whole global state of your app is stored in an `object tree` inside a single store. The only way to change the state tree is to create an *action*, an object describing what happened, and *dispatch* it to the store. To specify how state gets updated in response to an action, you write pure *reducer* functions that calculate a new state based on the old state and the action.

Action = object describe “what happened”

Reducer = kalkulasi dan membalikan state baru

```
import { createStore } from 'redux'

/**
 * This is a reducer - a function that takes a current state value and an
 * action object describing "what happened", and returns a new state value.
 * A reducer's function signature is: (state, action) => newState
 *
 * The Redux state should contain only plain JS objects, arrays, and primitives.
 * The root state value is usually an object. It's important that you should
 * not mutate the state object, but return a new object if the state changes.
 *
 * You can use any conditional logic you want in a reducer. In this example,
 * we use a switch statement, but it's not required.
 */
function counterReducer(state = { value: 0 }, action) {
  switch (action.type) {
    case 'counter/incremented':
      return { value: state.value + 1 }
    case 'counter/decremented':
      return { value: state.value - 1 }
    default:
      return state
  }
}
```

```
// Create a Redux store holding the state of your app.
// Its API is { subscribe, dispatch, getState }.
let store = createStore(counterReducer)

// You can use subscribe() to update the UI in response to state changes.
// Normally you'd use a view binding library (e.g. React Redux) rather than subscribe() directly.
// There may be additional use cases where it's helpful to subscribe as well.

store.subscribe(() => console.log(store.getState()))

// The only way to mutate the internal state is to dispatch an action.
// The actions can be serialized, logged or stored and later replayed.
store.dispatch({ type: 'counter/incremented' })
// {value: 1}
store.dispatch({ type: 'counter/incremented' })
// {value: 2}
store.dispatch({ type: 'counter/decremented' })
// {value: 1}
```

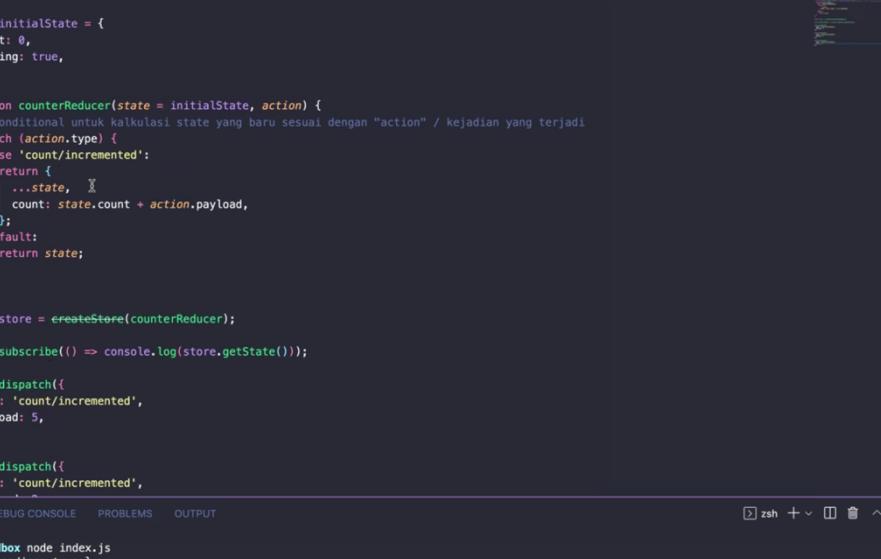
```
JS index.js x {} package.json

JS index.js > ...
1 const { createStore } = require('redux');
2
3 const initialState = {
4   count: 0,
5   loading: true,
6 };
7
8 function counterReducer(state = initialState, action) {
9   // conditional untuk kalkulasi state yang baru sesuai dengan "action" / kejadian yang terjadi
10  switch (action.type) {
11    case 'count/incremented':
12      return {
13        ...state,
14        count: state.count + 1,
15      };
16    default:
17      return state;
18  }
19}
20
21 const store = createStore(counterReducer);
22
23 store.subscribe(() => console.log(store.getState()));
24
25 store.dispatch({
26   type: 'count/incremented',
27 });
28
29 store.dispatch({
30   type: 'count/incremented',
31 });

TERMINAL DEBUG CONSOLE PROBLEMS OUTPUT
zsh + × □ ^ ×

● ➔ redux-sandbox node index.js
{ count: 1, loading: true }
{ count: 2, loading: true }
{ count: 3, loading: true }
○ ➔ redux-sandbox
```

## CUSTOM PAYLOAD



The screenshot shows a terminal window with the following content:

```
redux-sandbox node index.js
{ count: 5, loading: true }
{ count: 7, loading: true }
{ count: 8, loading: true }
redux-sandbox
```

## PRINSIP UTAMA

- Single source of truth = global state disimpan di 1 objek
- State is readonly = untuk mengubah dari action, karna kalau ga readonly (ga akan kecatat dan tidak reaktif)
- Changes made with pure functions counterReducer = () => {}
- http request, async process tidak boleh dilakukan

npm install react-redux

The screenshot shows a code editor with a dark theme. The top tab bar has tabs for README.md, index.js (src), and index.js .../store 1, U. The main editor area contains the following code:

```
src > store > JS index.js > ...
1 import { createStore } from 'redux'
2
3 const initialState = {
4   count: 0,
5 };
6
7 // pure function
8 // Function tidak memiliki side effect ke variabel di luar fn itu
9 // Diberikan parameter yang sama kembalian harus sama
10
11 function rootReducer(state = initialState, action) {
12   // conditional untuk kalkulasi state yang baru sesuai dengan "action" / kejadian yang terjadi
13   switch (action.type) {
14     case 'count/incremented':
15       // state.count += 5; // state read only ga boleh langsung diubah
16       return {
17         ...state,
18         count: state.count + action.payload,
19       };
20     default:
21       return state;
22   }
23 }
24
25 const store = createStore(rootReducer);
26
```

Below the editor is a terminal window with the following output:

```
TERMINAL DEBUG CONSOLE PROBLEMS 1 OUTPUT
+
You can now view router6 in the browser.
Local: http://localhost:3000
On Your Network: http://192.168.100.14:3000
Note that the development build is not optimized.
To create a production build, use npm run build.
```

On the right side of the interface, there is a sidebar with a list of open terminals:

- node
- zsh redux-sandbox
- zsh

```
src > JS index.js
1 import React from 'react';
2 import ReactDOM from 'react-dom';
3 import './index.css';
4 import App from './App';
5 import reportWebVitals from './reportWebVitals';
6 import { BrowserRouter } from 'react-router-dom';
7 import { Provider } from 'react-redux';
8
9 ReactDOM.render(
10   <Provider>
11     <BrowserRouter>
12       <App />
13     </BrowserRouter>
14   </Provider>,
15   document.getElementById('root')
16 );
17
18 // If you want to start measuring performance in your app, pass a function
19 // to log results (for example: reportWebVitals(console.log))
20 // or send to an analytics endpoint. Learn more: https://bit.ly/CRA-vitals
21 reportWebVitals();
22
```

TERMINAL DEBUG CONSOLE PROBLEMS 1 OUTPUT

You can now view router6 in the browser.

Local: http://localhost:3000  
On Your Network: http://192.168.100.14:3000

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

+ ^ x

src > pages > JS Home.js M X JS index.js src M JS Home.js M X JS > ⌂ ⌂ ⌂ ...

JS index.js U X JS Home.js M

```
src > store > JS index.js > [o] initialState > ↴ count
1 import { legacy_createStore as createStore } from
  'redux';
2
3 const initialState = {
4   count: 0,
5 };
6
7 function rootReducer(state = initialState, action) {
8   // conditional untuk kalkulasi state yang baru sesuai
   dengan "action" / kejadian yang terjadi
9   switch (action.type) {
10     case 'count/incremented':
11       // state.count += 5; // state.read only ga boleh
         langsung diubah
12       return {
13         ...state,
14         count: state.count + action.payload,
15       };
16     default:
17       return state;
18   }
19 }
20
21 const store = createStore(rootReducer);
22
23 export default store;
```

TERMINAL DEBUG CONSOLE PROBLEMS OUTPUT

Compiled with warnings.

src/index.js
Line 8:8: 'store' is defined but never used `no-unused-vars`

Search for the `keywords` to learn more about each warning.  
To ignore, add `// eslint-disable-next-line` to the line before.

+ ^ x

node
zsh redux-sandbox
zsh

```
src > pages > JS Home.js > ⚡ Home > ⚡ useEffect() callback
1 import { useSelector, useDispatch } from 'react-redux';
2 import { useEffect } from 'react';
3 export default function Home() {
4   const count = useSelector((state) => {
5     return state.count;
6   });
7   const dispatch = useDispatch();
8
9   useEffect(() => {
10     // data
11     const data = [{ id: 1, name: 'apa' }];
12     dispatch({
13       type: 'todo/fetchSuccess',
14       payload: data,
15     });
16   }, []);
17
18   return (
19     <div>
20       <h1>Home {count}</h1>
21       <button
22         onClick={() =>
23           dispatch({
24             type: 'count/incremented',
25             payload: 5,
26           })
27         }
28       >
29         INCREMENT
30       </button>
31     </div>
32   );
33 }
```

```
src > store > JS index.js > ⚡ rootReducer
1 import { legacy_createStore as createStore } from
2   'redux';
3
4 const initialState = {
5   count: 0,
6   todos: [],
7 };
8
8 function rootReducer(state = initialState, action) {
9   // conditional untuk kalkulasi state yang baru sesuai
10  dengan "action" / kejadian yang terjadi
11  switch (action.type) {
12    case 'count/incremented':
13      // state.count += 5; // state read only ga boleh
14      langsung diubah
15      return {
16        ...state,
17        count: state.count + action.payload,
18      };
19    case 'todos/fetchSuccess':
20      return {
21        ...state,
22        todos: action.payload,
23      };
24    default:
25      return state;
26  }
27
28 const store = createStore(rootReducer);
29
```

TERMINAL DEBUG CONSOLE PROBLEMS 1 OUTPUT

```
src/pages/Home.js
Line 16:6: React Hook useEffect has a missing dependency: 'dispatch'. Either include it or remove the dependency array react-hooks/exhaustive-deps
```

Search for the [keywords](#) to learn more about each warning.  
To ignore, add `// eslint-disable-next-line` to the line before.

```
node
zsh redux-sandbox
zsh
```

```
src > pages > JS Home.js > ⚡ Home > ⚡ submitTodo > ⚡ payload
1 import { legacy_createStore as createStore } from
2   'redux';
3
4 const initialState = {
5   count: 0,
6   todos: [],
7 };
8
8 function rootReducer(state = initialState, action) {
9   // conditional untuk kalkulasi state yang baru sesuai
10  dengan "action" / kejadian yang terjadi
11  switch (action.type) {
12    case 'count/incremented':
13      // state.count += 5; // state read only ga boleh
14      langsung diubah
15      return {
16        ...state,
17        count: state.count + action.payload,
18      };
19    case 'todos/fetchSuccess':
20      return {
21        ...state,
22        todos: action.payload,
23      };
24    case 'todos/addSuccess':
25      return {
26        ...state,
27        todos: [...state.todos, action.payload],
28      };
29    default:
30      return state;
31  }
32
33 const submitTodo = () => {
34   // fetch server untuk POST
35   const newTodo = {
36     id: 2,
37     name: "Test"
38   }
39   dispatch({
40     type: 'todos/addSuccess',
41     payload: newTodo
42   })
43 }
```

```
src > store > JS index.js > ⚡ rootReducer > ⚡ todos
1 import { legacy_createStore as createStore } from
2   'redux';
3
4 const initialState = {
5   count: 0,
6   todos: [],
7 };
8
8 function rootReducer(state = initialState, action) {
9   // conditional untuk kalkulasi state yang baru sesuai
10  dengan "action" / kejadian yang terjadi
11  switch (action.type) {
12    case 'count/incremented':
13      // state.count += 5; // state read only ga boleh
14      langsung diubah
15      return {
16        ...state,
17        count: state.count + action.payload,
18      };
19    case 'todos/fetchSuccess':
20      return {
21        ...state,
22        todos: action.payload,
23      };
24    case 'todos/addSuccess':
25      return {
26        ...state,
27        todos: [...state.todos, action.payload],
28      };
29    default:
30      return state;
31  }
32
```

TERMINAL DEBUG CONSOLE PROBLEMS 1 OUTPUT

```
src/pages/Home.js
Line 16:6: React Hook useEffect has a missing dependency: 'dispatch'. Either include it or remove the dependency array react-hooks/exhaustive-deps
```

Search for the [keywords](#) to learn more about each warning.  
To ignore, add `// eslint-disable-next-line` to the line before.

```
node
zsh redux-sandbox
zsh
```

① README.md    JS index.js src M    JS Home.js 1, M ×    J ⌂ ⌂ ⌂ ...

src > pages > JS Home.js > ⚡ Home

```

1 import { useSelector, useDispatch } from 'react-redux';
2 import { useEffect } from 'react';
3 export default function Home() {
4   const { count, todos } = useSelector((state) => {
5     return state;
6   });
7
8   const dispatch = useDispatch();
9
10  useEffect(() => {
11    // ini ceritanya datanya dari fetch
12    const data = [{ id: 1, name: 'apap' }];
13    dispatch({
14      type: 'todos/fetchSuccess',
15      payload: data,
16    });
17  }, []);
18
19  const submitTodo = () => {
20    // fetch server untuk POST
21    const newTodo = {
22      id: 2,
23      name: 'Test',
24    };
25    dispatch({
26      type: 'todos/addSuccess',
27      payload: newTodo,
28    });
29  };
30
31  return (
32

```

JS index.js U ×    JS Home.js 1, M

src > store > JS index.js > ⚡ rootReducer > todos

```

1 import { legacy_createStore as createStore } from
2   'redux';
3
4 const initialState = {
5   count: 0,
6   todos: [],
7 };
8
9 function rootReducer(state = initialState, action) {
10  // conditional untuk kalkulasi state yang baru sesuai
11  // dengan "action" / kejadian yang terjadi
12  switch (action.type) {
13    case 'count/incremented':
14      // state.count += 5; // state read only ga boleh
15      // langsung diubah
16      return {
17        ...state,
18        count: state.count + action.payload,
19      };
20    case 'todos/fetchSuccess':
21      return {
22        ...state,
23        todos: action.payload,
24      };
25    case 'todos/addSuccess':
26      return {
27        ...state,
28        todos: [...state.todos, action.payload],
29      };
30    default:
31      return state;
32

```

TERMINAL    DEBUG CONSOLE    PROBLEMS 1    OUTPUT

src/pages/Home.js

Line 17:6: React Hook useEffect has a missing dependency: 'dispatch'. Either include it or remove the dependency array react-hooks/exhaustive-deps

Search for the keywords to learn more about each warning.  
To ignore, add // eslint-disable-next-line to the line before.

① README.md    JS index.js src M    JS Users.js ●    JS index.js .../store U ● ...

src > pages > JS Users.js > ⚡ Users > ⚡ useEffect() callback > ⚡ then() callback

```

1 import { useEffect, useState } from 'react';
2 import { useSelector, useDispatch } from 'react-redux';
3 import './assets/table.css';
4 import UserRow from '../components/UserRow';
5
6 export default function Users() {
7   // const [users, setUsers] = useState([]);
8   // const [error, setError] = useState(null);
9   // const [loading, setLoading] = useState(true);
10  const { users } = useSelector((state) => state);
11  const dispatch = useDispatch();
12
13  useEffect(() => {
14    fetch('https://jsonplaceholder.typicode.com/users')
15      .then((res) => {
16        if (res.ok) {
17          return res.json();
18        } else {
19          throw new Error('Something went wrong');
20        }
21      })
22      .then((data) => dispatch({
23        type: 'users/fetchSuccess',
24        payload: data
25      }))
26      .catch((err) => console.log(err))
27  }, []);
28
29  // if (loading) {
30  //   return <div>Loading...</div>;
31  // }

```

JS index.js U ●    JS Home.js 1, M

src > store > JS index.js > ⚡ rootReducer

```

6   users: []
7 }
8
9 function rootReducer(state = initialState, action) {
10  // conditional untuk kalkulasi state yang baru sesuai
11  // dengan "action" / kejadian yang terjadi
12  switch (action.type) {
13    case 'count/incremented':
14      // state.count += 5; // state read only ga boleh
15      // langsung diubah
16      return {
17        ...state,
18        count: state.count + action.payload,
19      };
20    case 'todos/fetchSuccess':
21      return [
22        ...state,
23        todos: action.payload,
24      ];
25    case 'todos/addSuccess':
26      return {
27        ...state,
28        todos: [...state.todos, action.payload],
29      };
30    case 'users/fetchSuccess':
31      return {
32        ...state,
33        users: action.payload
34      };
35    default:
36      return state;
37

```

TERMINAL    DEBUG CONSOLE    PROBLEMS 1    OUTPUT

Compiled with warnings.

src/index.js

Line 8:8: 'store' is defined but never used `no-unused-vars`

src/pages/Home.js

Line 17:6: React Hook useEffect has a missing dependency: 'dispatch'. Either include it or remove the dependency array `react-hooks/exhaustive-deps`

+ ⌂ ⌂ ⌂ ...

node  
zsh redux-sandbox  
zsh

+ ⌂ ⌂ ⌂ ...

node  
zsh redux-sandbox  
zsh

## ACTIONS AS FUNCTIONS

```
src > store > actions > index.js > ...
1 // action creator
2 // Function -> Object (action)
3
4 export const countIncremented = (payload) => {
5   return {
6     type: 'count/incremented',
7     payload,
8   };
9 }
10
11 export const todoAddSuccess = (payload) => {
12   return {
13     type: 'todos/addSuccess',
14     payload,
15   };
16 }
17

src > pages > Home.js > Home
1 import { useSelector, useDispatch } from 'react-redux';
2 import { countIncremented, todoAddSuccess } from '../store/actions/index.js';
3
4 export default function Home() {
5   const { count, todos } = useSelector((state) => {
6     return state;
7   });
8
9   const dispatch = useDispatch();
10
11   useEffect(() => {
12     // ini ceritanya datanya dari fetch
13     const data = [{ id: 1, name: 'apa' }];
14     dispatch({
15       type: 'todos/fetchSuccess',
16       payload: data,
17     });
18   }, []);
19
20   const submitTodo = () => {
21     // fetch server untuk POST
22     // { name: '' } body
23     // res.status(201).json(todoYangBarusanDibuat)
24     const newTodo = {
25       id: 2,
26       name: 'Test',
27     };
28     dispatch(todoAddSuccess(newTodo));
29   };
30
31   return (
32     <div>
33       <h1>Home</h1>
34       <p>Count: {count}</p>
35       <ul>
36         {todos.map((todo) => (
37           <li>{todo.name}</li>
38         ))}
39       </ul>
40       <button onClick={submitTodo}>Submit</button>
41     </div>
42   );
43 }
```

TERMINAL DEBUG CONSOLE PROBLEMS 1 OUTPUT

```
Compiled with warnings.

src/index.js
Line 8:8: 'store' is defined but never used no-unused-vars

src/pages/Home.js
Line 18:6: React Hook useEffect has a missing dependency: 'dispatch'. Either include it or remove the dependency array react-hooks/exhaustive-deps.
```

+ ^ x

node  
zsh redux-sandbox  
zsh

## ACTION TYPE

**TOP IMAGE (Screenshot 1):**

The screenshot shows a VS Code interface with two main panes. The left pane displays the file structure of a React-Redux application, including `src` (with `store`, `actions`, and `reducers`), `pages` (with `About.js`, `Home.js`, `Login.js`, `Profile.js`, `ProfileGraduate.js`, `ProfileHiringPartner.js`, `UserDetail.js`, and `Users.js`), and `#` (with `App.css`, `App.js`, `App.test.js`, `index.css`, and `index.js`). The right pane shows the code editor with three tabs: `JS actionTypes.js` (containing action creators for `COUNT\_INCREMENTED` and `todoAddSuccess`), `JS index.js` (containing reducers for `rootReducer`), and `JS Home.js` (containing the `Home` component). The terminal at the bottom shows compilation warnings.

```

src > store > actions > JS actionTypes.js ...
1 // action creator
2 // Function => Object (action)
3 import { COUNT_INCREMENTED } from './actionType';
4
5 export const countIncremented = (payload) => {
6   return {
7     type: COUNT_INCREMENTED,
8     payload,
9   };
10};

11
12 export const todoAddSuccess = (payload) => {
13   return {
14     type: 'todos/addSuccess',
15     payload,
16   };
17};

18
19 export const todoFetchSuccess = (payload) => {
20   return {
21     type: 'todos/fetchSuccess',
22     payload,
23   };
24};

25

```

```

JS index.js ...
1 import { COUNT_INCREMENTED } from '../actions/actionType';
2
3 const initialState = {
4   count: 0,
5   todos: [],
6   users: [],
7   loadingUsers: true,
8 };
9
10 export default function rootReducer(state =
initialState, action) {
11   // conditional untuk kalkulasi state yang baru sesuai
dengan "action" / kejadian yang terjadi
12   switch (action.type) {
13     case COUNT_INCREMENTED:
14       // state.count += 5; // state read only ga boleh
langsung diubah
15       return {
16         ...state,
17         count: state.count + action.payload,
18       };
19     case 'todos/fetchSuccess':
20       return {
21         ...state,
22         todos: action.payload,
23       };
24     case 'todos/addSuccess':
25       return {
26         ...state,
27         todos: [...state.todos, action.payload],

```

**MIDDLE IMAGE (Screenshot 2):**

This screenshot is identical to the first one, showing the same file structure and code editor content. The terminal at the bottom shows the same compilation warnings.

**Bottom of Screenshot 2:**

```

src > store > actions > JS actionTypes.js ...
1 export const COUNT_INCREMENTED = 'count/incremented';
2

```

**Bottom of Screenshot 1:**

```

src > store > reducers > JS index.js > rootReducer
1 import { COUNT_INCREMENTED } from '../actions/actionType';
2
3 const initialState = {
4   count: 0,
5   todos: [],
6   users: [],
7   loadingUsers: true,
8 };
9
10 export default function rootReducer(state =
initialState, action) {
11   // conditional untuk kalkulasi state yang baru sesuai
dengan "action" / kejadian yang terjadi
12   switch (action.type) {
13     case COUNT_INCREMENTED:
14       // state.count += 5; // state read only ga boleh
langsung diubah
15       return {
16         ...state,
17         count: state.count + action.payload,
18       };
19     case 'todos/fetchSuccess':
20       return {
21         ...state,
22         todos: action.payload,
23       };
24     case 'todos/addSuccess':
25       return {
26         ...state,
27         todos: [...state.todos, action.payload],

```

The screenshot shows a VS Code interface with the following details:

- EXPLORER**: Shows the project structure with files grouped into "GROUP 1" and "GROUP 2". "GROUP 1" includes `index.js`, `store`, `actionType.js`, and `Users.js`. "GROUP 2" includes `index.js`, `store/redux`, and `Home.js`. A folder named "LECTURE-H8-REAC..." is also listed.
- EDITOR**: Two tabs are open:
  - `index.js` (U) in the first tab, showing code for creating a store using `legacy_createStore` and `rootReducer`.
  - `index.js` (U) in the second tab, showing the `rootReducer` function which handles actions like `COUNT_INCREMENTED` and `todos/fetchSuccess`.
- TERMINAL**: Shows compilation warnings and React Hook errors.

```
Compiled with warnings.

src/index.js
  Line 8:8: 'store' is defined but never used  no-unused-vars

src/pages/Home.js
  Line 19:6: React Hook useEffect has a missing dependency: 'dispatch'. Either include it or remove the dependency array  react-hooks/exhaustive-deps
```
- SIDE BAR**: Shows recent files: `App.css`, `App.js`, `App.test.js`, `index.css`, and `index.js`.
- STATUS BAR**: Shows file status (U for unstaged changes) and modification counts (e.g., 1, M).