## Page:1

# Redux Thunk

### What is Redux Thunk?

🚀 **Redux Thunk**: Middleware for Redux to handle async actions.

### Why Use Redux Thunk?

🔄 Handles async operations like API calls. 🔧 Allows dispatching functions instead of actions.

### Installation

📦 Install with npm or yarn:

bashCopy code

npm install redux-thunk

### Importing Redux Thunk

📥 Import in your project:

javascriptCopy code

import {thunk} from 'redux-thunk';

### Applying Middleware

🔗 Connect thunk to Redux store:

javascriptCopy code

import { createStore, applyMiddleware } from 'redux';

import thunk from 'redux-thunk';

import rootReducer from './reducers';

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

### Using Redux Thunk

🖊️ Write an async action creator:

javascriptCopy code

const fetchData = () => {

## Page:2

return (dispatch) => {

dispatch({ type: 'FETCH\_START' });

fetch('https://api.example.com/data')

.then(response => response.json())

.then(data => {

dispatch({ type: 'FETCH\_SUCCESS', payload: data });

})

.catch(error => {

dispatch({ type: 'FETCH\_ERROR', payload: error });

});

};

};

### Dispatching the Async Action

📡 Use the async action in your component:

javascriptCopy code

import React, { useEffect } from 'react';

import { useDispatch } from 'react-redux';

import { fetchData } from './actions';

const MyComponent = () => {

const dispatch = useDispatch();

useEffect(() => {

dispatch(fetchData());

}, []);

return <div>My Component</div>;

};

export default MyComponent;

### Summary

📦 **Install** redux-thunk 📥 **Import** thunk 🔗 **Apply** middleware to Redux store 🖊️ **Write** async action creator 📡 **Dispatch** async action in component