In this lesson you:

* Learned about Redux middleware and wrote your own simple logging middleware
* Encountered thunks and learned about how valuable thunks are for deferring computation

const remindMeLater = task => {   
  return () => {  
    remindMeTo(task)  
  }   
}

* Discovered redux-thunk, a middleware that allows you to write asynchronous action creators that return thunks instead of objects
* Automatically enabled redux-thunk by using configureStore

import { configureStore } from '@reduxjs/toolkit;

* Took a deep dive into the middleware’s source code, in order to understand how the middleware actually works
* Wrote your own asynchronous action creators in the format that redux-thunk expects

const getUser = (id) => {  
  return async (dispatch, getState) => {  
    const payload = await fetchUser(id)  
    dispatch({type: 'users/addUser', payload: payload})  
  }  
}

If you’d like to learn more, you can read the [redux-thunk documentation](https://github.com/reduxjs/redux-thunk) and [visit the Redux Toolkit site to see how configureStore includes redux-thunk by default](https://redux-toolkit.js.org/api/getDefaultMiddleware#getdefaultmiddleware).