

05 - Một số effect creator nên biết trong Redux Saga

Agenda

1. Phân biệt Effect vs Effect Creator
2. Chia sẻ về một số

Phân biệt Effect vs Effect Creator

- **Effect**: chỉ đơn giản là một javascript object có chứa thông tin để saga middleware biết cần phải làm gì.
- **Effect Creator**: là một function trả về một Effect. Và nó ko thực thi cái Effect này, người thực thi là saga middleware, chứ ko phải Effect Creator nhé.
- Các hàm các bạn dùng trong Redux Saga: takeEvery, takeLatest, ... đây là những Effect Creator.

Các Effect Creator nên biết

#	Effect Creator	Trả về Effect chứa thông tin giúp middleware biết phải làm gì
1	takeEvery (pattern, saga, ...args)	Chạy saga mỗi lần khi có action pattern được dispatch, dispatch bao nhiêu sẽ chạy bấy nhiêu cái saga
2	takeLatest (pattern, saga, ...args)	Chạy saga , nhưng khi có action pattern mới được dispatch, thì cái saga trước đó sẽ bị cancel
3	takeLeading (pattern, saga, ...args)	Chạy saga khi action pattern được dispatch, những action tiếp theo sẽ bị cancel cho đến khi saga trước đó chạy xong.
4	put (action)	Dispatch action từ saga
5	call (fn, ...args)	Gọi hàm fn và truyền tham số args vào hàm đó
6	all ([...effects])	Chạy tất cả effects cùng một lúc
7	take (pattern) and fork (fn, ...args)	Mô hình watcher ... worker, đợi dispatch action pattern thì sẽ thực hiện một cái task nào đó
8	throttle (ms, pattern, saga, ...args)	Throttle cái action pattern , đảm bảo chỉ chạy saga 1 lần trong khoảng thời gian ms
9	debounce (ms, pattern, saga, ...args)	Debounce cái action pattern , đảm bảo chỉ chạy saga 1 lần sau khi đã đợi khoảng thời gian ms
10	retry (maxTries, delay, fn, ...args)	Cố gắng gọi lại hàm fn nếu fail, sau mỗi delay milliseconds, và tối đa chỉ thử maxTries lần.

Ngoài ra còn có các creator khác, tham khảo thêm tại đây: <https://redux-saga.js.org/docs/api#effect-creators>