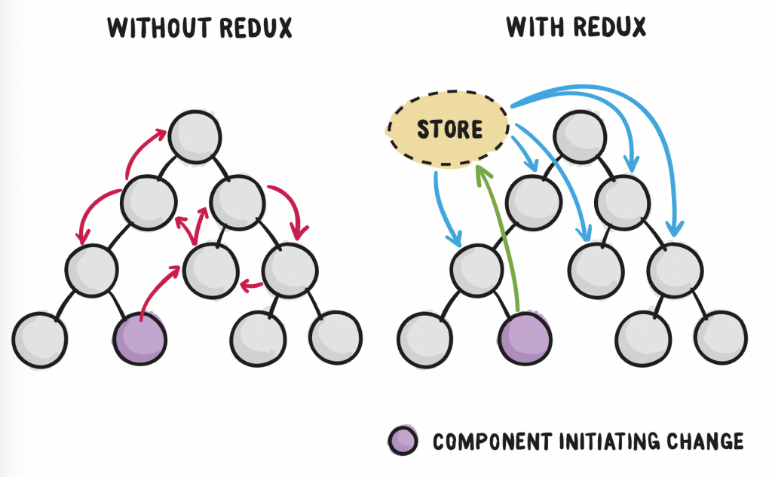
* **Redux**
  + Là 1 container quản lý **global state** của cả application thông qua Store (1 object global duy nhất)
  + Cho phép Component đọc data (global state) trực tiếp từ Redux Store và update data thông qua reducer
  + Có thể sử dụng được cả trong React, Angular, Vue
  + Rất nhẹ (2kB)



* **Nguyên lý**
  + **Single Source of Truth**

Single store, chỉ có 1 đối tượng store lưu trữ toàn bộ state của ứng dụng

* + **State is Read-only**
  + **Changes are made with functions** (reducers)
* **Các thành phần**
  + **Store** (object)
    - Lưu trữ, quản lý state của toàn bộ <App> (giống context)

Có các phương thức để update state (thông qua reducer)

* + - Các method
      * **store.getState()**

Lấy đối tượng state được lưu trong store

* + - * **store.dispatch(action)**

Gửi một action gồm một hành động (type) và dữ liệu (payload) lên reducer.

* + - * **store.subscribe(callback function)**

Đăng ký nhận sự kiện global state trong store thay đổi

|  |  |  |
| --- | --- | --- |
|  |  |  |

*Redux core*

*Nhận vào 1* ***function reducer*** *🡺 thường phải tạo Root Reducer để khai báo nhiều Child Reducer*

|  |  |  |
| --- | --- | --- |
|  |  |  |

*Redux toolkit*

*Nhận vào 1* ***object reducer 🡺 không cần phải tạo Root Reducer***

* + **Action** (object)
    - 1 component muốn thay đổi state trên store thì phải gửi action tới reducer (thông qua dispatch)
    - Action là 1 object (mô tả hành động muốn thực hiện). Gồm có:
      * **type** (required)

Để phân biệt các action với nhau

* + - * **payload** (optional)

Data gửi kèm

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | *Action có data* | *Action không có data* |   *Redux core* | |  |  | | --- | --- | | *Action không có data* | *Action có data* |   *Redux Toolkit* |

* + **Reducer** (function)
    - Là nơi tiếp nhận action và thay đổi state trong store nếu cần (thường gọi là dispatch một action – gửi 1 hành động tới reducer)
    - Không nên chỉnh state 🡺 copy ra 1 bản clone riêng và chỉnh sửa

|  |  |
| --- | --- |
|  |  |

* + - Luôn trả về 1 state mới



* + - Có 2 loại
      * **Root Reducer**

Quản lý tất cả các reducer con

* + - * **Child Reducer**

Mỗi 1 Child Reducer chịu trách nhiệm thay đổi 1 thuộc tính trong state

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

*Redux Core*

|  |  |  |
| --- | --- | --- |
|  |  |  |

*Không cần Root Reducer*

*Có thể chỉnh sửa state trực tiếp*

*Redux Toolkit*

* + **Các đối tượng khác**
    - **Action Creators** (function)

Là 1 function để tạo và return ra đối tượng action

|  |  |
| --- | --- |
|  |  |

*Redux core*

|  |  |
| --- | --- |
|  |  |

*Redux Toolkit*

* + - **Selectors** (function)

Là các function để extract thông tin từ đối tượng state

|  |  |
| --- | --- |
|  |  |

*Redux core*

|  |  |
| --- | --- |
|  |  |

*Redux Toolkit*

*Tìm hiểu thêm:* [*https://github.com/reduxjs/reselect*](https://github.com/reduxjs/reselect)

* + - **Dispatch** (function)

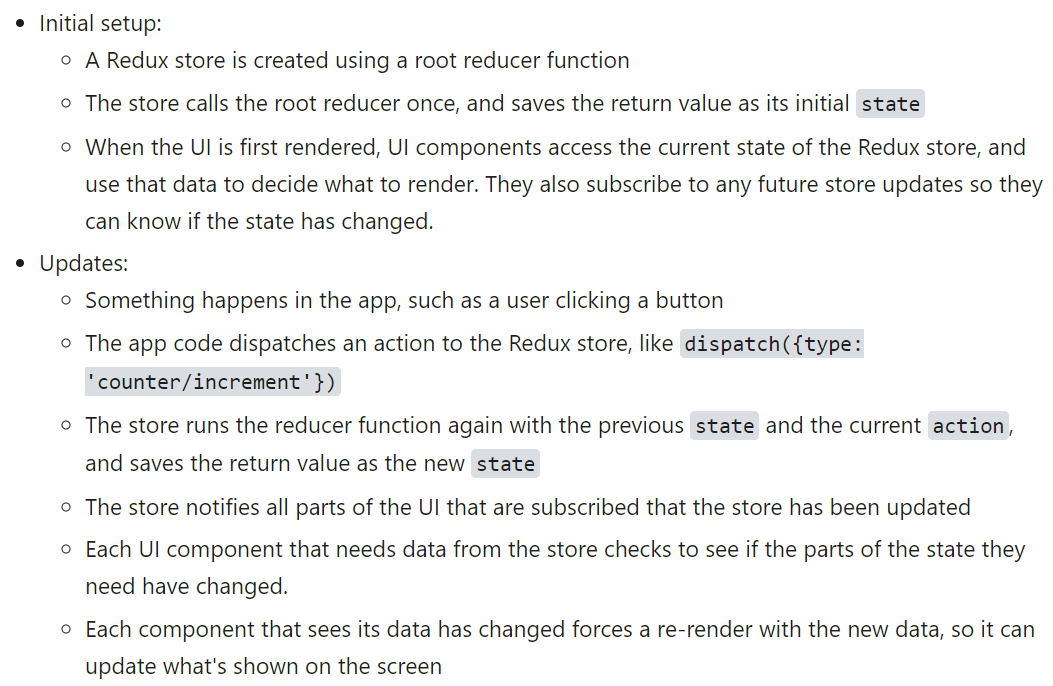
Chỉ có 1 cách để update state trong store là gọi qua store.dispatch() method và truyền đối tượng action vào

Là các function ở trong store

|  |  |
| --- | --- |
| *Redux core* | *Redux Toolkit* |

* **Flow**
  + Có 1 store lưu trữ toàn bộ state của App

Mỗi component có thể access trực tiếp đến state được lưu trữ thay vì phải gửi từ component này đến component khác.



|  |  |
| --- | --- |
|  |  |

[*https://redux.js.org/assets/images/ReduxDataFlowDiagram-49fa8c3968371d9ef6f2a1486bd40a26.gif*](https://redux.js.org/assets/images/ReduxDataFlowDiagram-49fa8c3968371d9ef6f2a1486bd40a26.gif)

* **Best Practice**
  + Redux store chỉ nên lưu global state, React Component sẽ lưu local state

VD: state trong form thì lưu local tại Component

* + Chỉ nên lưu global state khi có nhiều component cùng sử dụng

VD: theme

* **Reference**

<https://redux.js.org/introduction/getting-started>