Redux에 관하여

Redux란?

- Redux는 JS 어플리케이션에서 date-state와 UI-state를 관리해주는 도구이다.

- 싱글 페이지 어플(SPA)에 매우 유용하다.

- 상태적 데이터 관리가 시간의 흐름에 따라 복잡해질 수 있는 점을 방지

- React를 제외한 jQuery나 Angular에서도 유용하게 사용될 수 있다.

\* 알아두기 \*

- React에서 데이터 흐름은 단방향이다.

- parent-child 관계를 통하여 데이터를 교류한다.

- 만약 컴포넌트 개수가 많아질 시 parent-child 관계가 아니라면 웹 페이지가 훨씬 복잡해진다.

- 컴포넌트의 개수가 많고 서로 연결되는 구조가 복잡할 시, React로도 완벽히 해결되지는 않는다.

- 이를 해결하기 위해 React Document에서는 컴포넌트끼리 데이터를 교류할 때에는 글로벌 이벤트 시스템을 설정하는 방법이 있다고 명시했다. Flux 패턴이 그 예시이다.

- 다양한 패턴들

1. MVC 패턴

- Controller, Model, View 3가지로 이루어져있다.

- 설명 –

1. Action이 입력

2. Controller가 Model의 데이터를 조회 및 업데이트

3. 이러한 변화들은 View에 반영된다.

4. View에서 Model의 데이터에 접근 가능하다.

5. (중요!) 작은 어플리케이션에서는 잘 작동하지만

Model과 View가 늘어날 시 관리가 매우 힘들어진다.

2. FLUX 패턴

- Action, Dispatcher, Store, View로 볼 수 있다.

- 설명 -

: 시스템에서 특정 Action을 받았을 시, Dispatcher가 받은

Action들을 통제하여 Store에 있는 데이터를 업데이트한다. 변동된 데이터가 있을 시 View에 리렌더링 시킨다.

: 또한 View에서 Dispatcher로 Action을 보낼 수도 있다.

-Dispatcher의 역할-

: 수행되는 작업들이 중첩되지 않도록 해준다.

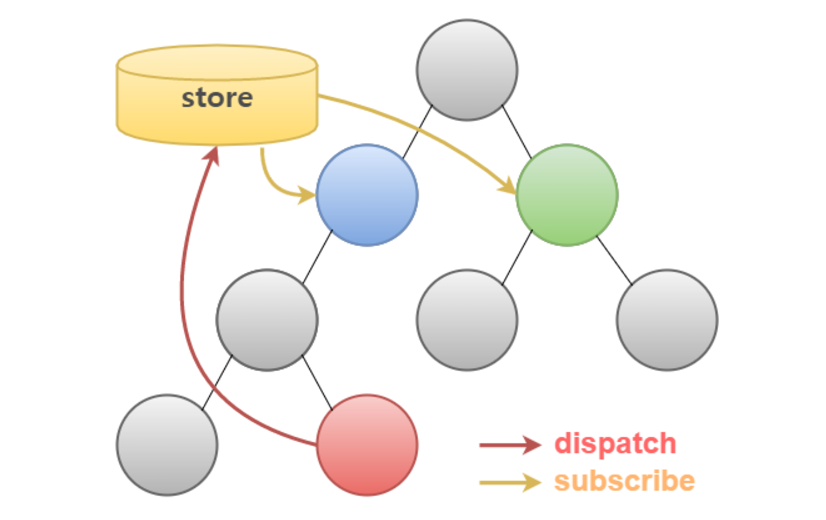
어떠한 Action이 Dispatcher를 통해 Store에 있는 데이터를

처리를 하고, 작업이 끝날 때까지 다른 Action들을 대기

시킨다.

3. Redux!

- Redux는 Flux를 좀 더 편하게 사용할 수 있게 하는 라.이.브.러.리 이다!



* Store에 모든 데이터가 담겨져있다. (모든 state는 한 곳에 있다.)
* 컴포넌트끼리는 직접 교류하지 않으며 store의 중간자?를 통하여 교류한다.
* Dispatch는 Store에 있는 데이터를 업데이트 하는 것을 가리키며 Subscribe는 Store에 있는 특정 데이터가 변동이 있을 시 즉시 반영을 하는 역할을 한다.
* Dispatch는 내보낸다라는 뜻이다.
* Action은 어떤 변화가 일어나야 할 지 알려주는 객체이다.
* Redux의 3가지 원칙 -.

1. Single Source of Truth (단 하나의 진실의 근원)

Redux는 어플리케이션의 state를 위해 단 한 개의 store를 사용한다. 아까 말했던 것처럼 모든 state가 한 곳에 있기 때문에 이런 식으로 부른다.

* 반면, Flux는 여러 개의 store를 사용한다.

1. State is read-only (State는 읽기 전용이다.)

Redux에서는 어플리케이션에서 state를 직접 변경할 수 없다. State를 변경하기 위해서는 action이 dispatch되어야 한다.

1. Changes are made with Pure Functions (변화는 순수 함수로 만들어 져야한다.)

Action을 dispatch하여 상태 값을 변경한다고 했는데 이 때 action 객체를 처리하는 함수를 ‘Reducer’라고 부른다.

Reducer는 쉽게 action의? 정보를 받고 어플리케이션의 상태를 ‘어떻게 바꿔야 할지’ 정의해주는 것이라 보면 될 것 같다.

* 순수 함수에 대한 설명 –

1. 외부 네트워크 혹은 DB에 접근하지 않아야 한다.
2. Return 값은 오직 파라미터 값에만 의존되어야 한다.
3. 인수는 변경되질 않아야 한다.
4. 같은 인수로 실행된 함수는 언제나 같은 결과를 반환해야 한다.
5. Date나 Math와 같은 순수하지 않은 API 호출을 해서는 안된다.

**간단한 설명에 대해 마무리**

**다음은 실제 사용에 대해 설명**