

## Jawaban soal teori oleh Naufal Zufar S

1. State dan props adalah “model” tipe data pada React.

State bersifat mutable(dapat diubah) dan diperuntukkan untuk pengolahan data didalam suatu component. State berbentuk object terenkapsulasi sehingga tidak dapat diakses diluar componentnya. State hanya dapat digunakan di class component (sebelum Hooks)

Sedangkan Props bersifat immutable(tidak dapat diubah) dan diperuntukkan untuk passing data antar component (biasanya antar parent-child). Props tersedia di class ataupun functional component.

Penggunaan state dan props dapat digabungkan untuk menghasilkan data yang dapat diolah di parent maupun child. Cth: button yang state nya terdapat di parent

	State	Props
Mutability	Mutable (dapat diubah / modifikasi)	Immutable (tidak dapat diubah-ubah)
Fungsi	Pengolahan data (State management)	Passing data antar component
Access from	Class component	Class and Functional component

2. State management pada React adalah metode pengolahan dan sharing data pada component. State management yang baik perlu dibuat agar struktur data pada aplikasi lebih jelas dan rapih sehingga mempermudah pengolahan, maintenance, dan skalabilitas aplikasi.

Salah satu contoh state management adalah cart pada ecommerce, dimana ketika kita menambahkan barang ke keranjang maka state cart akan terupdate dimana jumlah barang dan barang yang dipilih bertambah

3. Store dalam redux adalah tempat menyimpan dan mengolah keseluruhan state dalam aplikasi. Dimana kita melakukan akses

pemanggilan (getState), perubahan nilai (dispatch) dan merekam perubahan (subscribe)

4. Key adalah sebuah attribute dengan nilai unik yg harus disertakan ketika merender “list of item” pada react. Key digunakan untuk identifikasi / pembeda antar elemen list. React hanya akan melakukan render ulang pada list yg mengalami perubahan dengan cara membandingkan elemen dengan key yang sama. Hal ini dilakukan react agar proses rendering lebih efisien.

5. Redux adalah global state management tools yang digunakan untuk membuat state universal pada aplikasi. Dimana state universal ini nantinya dapat diakses oleh component manapun (di dalam scopenya). Redux menjadi dibutuhkan ketika aplikasi yang dibangun sudah cukup besar dan kompleks, dimana fungsi-fungsi aplikasi terbagi-bagi antar component dan bahkan sudah mempunyai child componentnya dengan kedalaman yang tinggi (nesting diatas 3 level). Dengan menggunakan redux maka child terdalam di aplikasi dapat mengakses ataupun memodifikasi state secara langsung tanpa melakukan drilling props .

Thunk adalah middleware Redux yang digunakan agar fungsi action creator dapat me-return fungsi (tidak hanya object). Hal ini memungkinkan fungsi untuk melakukan asynchronous call pada Redux. Middleware ini digunakan apabila kita membutuhkan state redux yang nilainya berasal dari luar aplikasi, seperti pemanggilan API.

Ya, saya telah menggunakan Redux dan Thunk dalam project. Redux pada aplikasi booking eventori.id (pesan talent) dan Redux Thunk dalam project (latihan) pemanggilan data user melalui public API json placeholder.

