Essay

1. Apa itu React?
2. Apa keunggulan React?
3. Apa itu SPA pada React?
4. Apa itu state dan props? Apa perbedaannya?
5. Kapan harus menggunakan state dan props? Berikan contoh studi kasus
6. Apa itu state management dan kapan harus menggunakannya?
7. Apa itu ORM/ODM?
8. Apa perbedaan SQL dan NoSQL?

Jawab

1. React JS adalah framework view library javascript untuk membuat tampilan (user interface) pada website
2. A. React JS is FAST => React JS membuat aplikasi front-end menjadi lebih cepat walaupun harus menghandle berbagai data.

B. React Js is MODULAR => Kita dapat menerapkan konsep Modular javascript pada React JS. React JS membagi 1 tampilan pada website menjadi komponen-komponen kecil.

C. React JS is Scalable => React JS dapat digunakan pada aplikasi bersakala kecil hingga besar dan kompleks.

D. React JS is Popular => Komunitas React JS diseluruh dunia sangat besar. Kebanyakan Perusahaan teknologi pun sudah menggunakan REACT JS.

1. SPA adalah singkatan dari Single Page Application. Ini adalah jenis aplikasi web yang memuat satu halaman HTML dan memperbarui konten yang ditampilkan secara dinamis saat pengguna berinteraksi dengan aplikasi tersebut, tanpa harus memuat ulang seluruh halaman dari server.
2. State dan Props adalah hal yang berhubungan dengan Statless dan Stateful Component.   
   Stateless berarti tidak memiliki State. Dia hanya memiliki props.  
   Statefull berarti memiliki state dan bisa mengirim state tersebut ke component.

Perbedaan :  
-State adalah data lokal yang dimiliki oleh komponen dan dikelola di dalam komponen itu sendiri, Sementara props digunakan agar component memiliki data yang dinamis yang dikirim dari component lain.

1. **State :**

* Gunakan state ketika data perlu diubah atau dikelola oleh komponen itu sendiri.
* Gunakan state untuk menyimpan data lokal yang hanya berlaku untuk komponen tersebut.
* Gunakan state untuk menyimpan data yang mungkin berubah seiring waktu, seperti data formulir yang diisi pengguna.

**Contoh Studi Kasus Penggunaan State:** Sebuah komponen Formulir Pendaftaran yang memiliki inputan untuk nama, email, dan nomor telepon. Data yang dimasukkan pengguna disimpan dalam state komponen Formulir. Ketika pengguna mengisi formulir, nilai state diperbarui, dan ketika formulir dikirim, nilai state digunakan untuk mengirim data ke server.

**Props :**

* Gunakan props ketika data perlu dipass dari komponen induk ke komponen anak.
* Gunakan props untuk membuat komponen lebih reusable dan dinamis.
* Gunakan props untuk mengirim fungsi dari komponen induk ke komponen anak untuk berinteraksi dengan komponen tersebut.

**Contoh Studi Kasus Penggunaan Props:** Sebuah komponen List yang menampilkan daftar item. Data daftar item dan fungsi untuk menambahkan item baru diteruskan sebagai props ke komponen List.

1. State management adalah proses mengelola dan menyimpan status atau data aplikasi secara keseluruhan, sehingga dapat diakses dan dikelola dengan mudah di seluruh aplikasi.

Kapan harus menggunakannya :

* + Aplikasi Memiliki State yang Kompleks dan Beragam
  + Banyak Komponen Membutuhkan Akses ke Data yang Sama
  + Memiliki Kebutuhan untuk Memelihara Status Aplikasi yang Konsisten

1. ORM dan ODM adalah singkatan dari dua konsep yang berbeda yang digunakan dalam pengembangan perangkat lunak:

**ORM (Object-Relational Mapping)**

* ORM adalah teknik yang digunakan dalam pengembangan perangkat lunak untuk mengonversi data antara tipe data yang terkait dengan objek dalam bahasa pemrograman dengan skema basis data relasional.
* ORM memungkinkan pengembang untuk bekerja dengan data dalam bentuk objek tanpa perlu berurusan langsung dengan kueri SQL dan struktur tabel dalam database relasional.
* ORM menyediakan antarmuka pemrograman tingkat tinggi (API) yang memungkinkan penggunaan metode objek untuk melakukan operasi database seperti pencarian, penambahan, penghapusan, dan pembaruan.
* Contoh ORM populer termasuk Hibernate untuk Java, Entity Framework untuk .NET, dan Sequelize untuk JavaScript (Node.js).

**ODM (Object-Document Mapping)**

* ODM adalah konsep serupa dengan ORM, tetapi digunakan dalam konteks basis data dokumen, seperti MongoDB, yang menggunakan format dokumen seperti JSON atau BSON.
* ODM memungkinkan pengembang untuk berinteraksi dengan basis data dokumen menggunakan objek dalam bahasa pemrograman.
* ODM mengonversi data antara dokumen dalam basis data dokumen dengan objek dalam kode program.
* ODM biasanya menyediakan fitur-fitur seperti validasi data, validasi skema, dan fitur-fitur lain yang terkait dengan basis data dokumen.
* Contoh ODM populer termasuk Mongoose untuk MongoDB dalam lingkungan Node.js.

1. SQL (Structured Query Language) dan NoSQL (Not Only SQL) adalah dua jenis basis data yang berbeda, dan ada beberapa perbedaan antara keduanya:

**Struktur Data:**

* SQL adalah basis data relasional yang menggunakan skema tetap dan tabel terstruktur dengan kolom yang didefinisikan sebelumnya. Data disimpan dalam tabel yang terhubung melalui kunci asing.
* NoSQL adalah basis data non-relasional yang tidak memiliki skema tetap dan dapat menyimpan data dalam bentuk yang lebih fleksibel, seperti dokumen, grafik, kolom, atau kunci-nilai.

**Query Language:**

* SQL menggunakan bahasa kueri yang disebut SQL untuk melakukan operasi seperti SELECT, INSERT, UPDATE, DELETE, dan JOIN untuk mengambil, menambahkan, memperbarui, dan menghapus data dari basis data relasional.
* NoSQL memiliki berbagai bahasa kueri tergantung pada jenis NoSQL yang digunakan. Beberapa NoSQL menggunakan bahasa kueri mirip SQL, sementara yang lain menggunakan API atau bahasa kueri yang berbeda.