

Soal 1: Todo List dengan TypeScript dan React Hooks

Deskripsi:

Buatlah komponen daftar tugas menggunakan React dan TypeScript. Gunakan React Hooks untuk mengelola state dan TypeScript untuk pengetikan yang ketat.

Tugas:

1. Buatlah komponen React bernama `TodoList` dengan:
 - Input untuk menambahkan tugas baru.
 - Daftar tugas yang sudah ditambahkan.
 - Tombol untuk menghapus setiap tugas.
2. Gunakan TypeScript untuk mendefinisikan tipe data.
3. Gunakan Hooks (`useState`) untuk mengelola state.

Kriteria Penilaian:

- Fungsi `add` dan `delete` harus bekerja dengan baik.
- TypeScript digunakan dengan benar.
- Hooks digunakan dengan efisien.
- UI sederhana dan fungsional.

Soal 2: Formulir Pendaftaran dengan Validasi dan React Hook Form

Deskripsi:

Buatlah komponen formulir pendaftaran pengguna dengan React dan TypeScript. Gunakan react-hook-form untuk validasi dan manajemen formulir.

Tugas:

1. Buatlah komponen React bernama RegistrationForm dengan input berikut:
 - **Nama:** Input teks.
 - **Email:** Input teks.
 - **Password:** Input password.
 - **Kirim:** Tombol kirim.
2. Gunakan TypeScript untuk pengetikan yang ketat.
3. Gunakan react-hook-form untuk menangani validasi.

Kriteria Penilaian:

- Validasi input harus berfungsi dengan baik.
- TypeScript digunakan dengan benar.
- react-hook-form digunakan untuk manajemen formulir.

Soal 3: Implementasi Routing dengan React Router

Deskripsi:

Buatlah aplikasi React sederhana dengan dua halaman menggunakan React Router. Halaman pertama adalah daftar tugas dan halaman kedua adalah detail tugas.

Tugas:

1. Instal dan konfigurasi react-router-dom.
2. Buat dua komponen: TodoList (daftar tugas) dan TaskDetail (detail tugas).
3. Implementasikan routing sehingga pengguna dapat beralih antara halaman daftar tugas dan detail tugas.

Kriteria Penilaian:

- Routing harus berfungsi dengan baik.
- Struktur komponen harus bersih dan terpisah.
- UI harus mudah dinavigasi.