Source Code:

/src/App.jsx

```jsx
Pertemuan-3 > src > ⚛ App.jsx > ⓣ App
 1  import React from "react";
 2  import ReactDOM from "react-dom/client";
 3  import { Toaster } from "react-hot-toast";
 4  import { createBrowserRouter, RouterProvider } from "react-router-dom";
 5  import { AuthProvider } from "@/Pages/Layouts/Utils/Context/AuthContext";
 6  import "@/App.css";
 7  import { QueryClient, QueryClientProvider } from "@tanstack/react-query";
 8
 9  // Pages
10  import Login from "@/Pages/Auth/Login";
11  import Dashboard from "@/Pages/Admin/Dashboard";
12  import Mahasiswa from "@/Pages/Admin/Mahasiswa";
13  import MahasiswaDetail from "@/Pages/Admin/MahasiswaDetail";
14  import PageNotFound from "@/Pages/PageNotFound";
15  import RencanaStudi from "@/Pages/Admin/RencanaStudi";
16
17  // Layouts
18  import AuthLayout from "@/Pages/Layouts/AuthLayout";
19  import AdminLayout from "@/Pages/Layouts/AdminLayout";
20  import ProtectedRoute from "@/Pages/Layouts/ProtectedRoute";
21
22  const router = createBrowserRouter([
23    {
24      path: "/",
25      element: <AuthLayout />,
26      children: [
27        { index: true, element: <Login /> },
28        { path: "login", element: <Login /> },
29      ],
30    },
31    {
32      path: "/admin",
33      element: (
34        <ProtectedRoute>
35          <AdminLayout />
36        </ProtectedRoute>
37      ),
38      children: [
39        { index: true, element: <Dashboard /> },
40        { path: "dashboard", element: <Dashboard /> },
41        { path: "mahasiswa", element: <Mahasiswa /> },
42        { path: "mahasiswa/:id", element: <MahasiswaDetail /> },
43        { path: "rencana-studi", element: <RencanaStudi /> },
44      ],
45    }
```

```jsx
44      ],
45    },
46    { path: "*", element: <PageNotFound /> },
47  ]);
48
49  const queryClient = new QueryClient();
50
51  ReactDOM.createRoot(document.getElementById("root")).render(
52    <React.StrictMode>
53      <QueryClientProvider client={queryClient}>
54        <AuthProvider>
55          <Toaster position="top-center" />
56          <RouterProvider router={router} />
57        </AuthProvider>
58      </QueryClientProvider>
59    </React.StrictMode>
60  );
61
62  export default function App() {
63    return <Login />;
64  }
65
```

/src/Pages/Layouts/Utils/Hooks/useMahasiswa.jsx

```jsx
1    // Utils/Hooks/useMahasiswa.jsx
2    import { useQuery, useMutation, useQueryClient } from "@tanstack/react-query";
3    import {
4      getAllMahasiswa,
5      storeMahasiswa,
6      updateMahasiswa,
7      deleteMahasiswa,
8    } from "@/Pages/Layouts/Utils/Apis/MahasiswaApi";
9    import { toastSuccess, toastError } from "@/Pages/Layouts/Utils/Helpers/ToastHelpers";
10
11   // Ambil semua mahasiswa
12   export const useMahasiswa = () =>
13     useQuery({
14       queryKey: ["mahasiswa"],
15       queryFn: getAllMahasiswa,
16       select: (res) => res?.data ?? [],
17     });
18
19   // Tambah
20   export const useStoreMahasiswa = () => {
21     const queryClient = useQueryClient();
22     return useMutation({
23       mutationFn: storeMahasiswa,
24       onSuccess: () => {
25         queryClient.invalidateQueries({ queryKey: ["mahasiswa"] });
26         toastSuccess("Mahasiswa berhasil ditambahkan!");
27       },
28       onError: () => toastError("Gagal menambahkan mahasiswa."),
29     });
30   };
31
32   // Edit
33   export const useUpdateMahasiswa = () => {
34     const queryClient = useQueryClient();
35     return useMutation({
36       mutationFn: ({ id, data }) => updateMahasiswa(id, data),
37       onSuccess: () => {
38         queryClient.invalidateQueries({ queryKey: ["mahasiswa"] });
39         toastSuccess("Mahasiswa berhasil diperbarui!");
40       },
41       onError: () => toastError("Gagal memperbarui mahasiswa."),
42     });
43   };
```

```jsx
45   // Hapus
46   export const useDeleteMahasiswa = () => {
47     const queryClient = useQueryClient();
48     return useMutation({
49       mutationFn: deleteMahasiswa,
50       onSuccess: () => {
51         queryClient.invalidateQueries({ queryKey: ["mahasiswa"] });
52         toastSuccess("Mahasiswa berhasil dihapus!");
53       },
54       onError: () => toastError("Gagal menghapus mahasiswa."),
55     });
56   };
```

/src/Pages/Layouts/Utils/Hooks/useMataKuliah.jsx

```jsx
1    // Utils/Hooks/useMataKuliah.jsx
2    import { useQuery } from "@tanstack/react-query";
3    import { getAllMataKuliah } from "@/Pages/Layouts/Utils/Apis/MataKuliahApi";
4
5    export const useMataKuliah = () =>
6      useQuery({
7        queryKey: ["matakuliah"],
8        queryFn: getAllMataKuliah,
9        select: (res) => res?.data ?? [],
10     });
```

/src/Pages/Layouts/Utils/Hooks/useKelas.jsx

```
Pertemuan-3 > src > Pages > Layouts > Utils > Hooks > ⚛ useKelas.jsx > ...
 1    // Utils/Hooks/useKelas.jsx
 2    import { useQuery } from "@tanstack/react-query";
 3    import { getAllKelas } from "@/Pages/Layouts/Utils/Apis/KelasApi";
 4
 5    export const useKelas = () =>
 6      useQuery({
 7        queryKey: ["kelas"],
 8        queryFn: getAllKelas,
 9        select: (res) => res?.data ?? [],
10      });
```

/src/Pages/Admin/Mahasiswa.jsx

```
// ================= DATA RELASI =================
const [kelas, setKelas] = useState([]);
const [mataKuliah, setMataKuliah] = useState([]);


// ================= FETCH MAHASISWA =================
const {
  data: mahasiswa = [],
  isLoading: isLoadingMahasiswa,
} = useMahasiswa();
```

```
// ================= MUTATION =================
const { mutate: store } = useStoreMahasiswa();
const { mutate: update } = useUpdateMahasiswa();
const { mutate: remove } = useDeleteMahasiswa();
```

```
// ================= SUBMIT =================
const handleSubmit = (e) => {
  e.preventDefault();

  if (!form.nim || !form.name || !form.max_sks) {
    toastError("NIM, Nama, dan Max SKS wajib diisi!");
    return;
  }

  if (isEdit) {
    confirmUpdate(() => {
      update({ id: form.id, data: form });
      resetForm();
    });
    return;
  }

  const exists = mahasiswa.find((m) => m.nim === form.nim);
  if (exists) {
    toastError("NIM sudah terdaftar!");
    return;
  }

  store(form);
  resetForm();
};
```

```
// ================= DELETE =================
const handleDelete = (id) => {
  confirmDelete(() => {
    remove(id);
  });
};
```