

Lampiran Source Code 1

```
import RouteLine from '@components/RouteLine';
import AOS from 'aos';
import 'aos/dist/aos.css';
import { LeafletMouseEvent } from 'leaflet';
import 'leaflet/dist/leaflet.css';
import { ChevronDown } from 'lucide-react';
import { useEffect, useState } from 'react';
import { MapContainer, Marker, Popup, TileLayer, useMap, useMapEvents } from 'react-leaflet';
import Footer from '../components/Footer';
import MainLayout from '../layouts/MainLayout';

interface Wisata {
  id: number;
  nama: string;
  lokasi: string;
  gambar: string;
  latitude: number;
  longitude: number;
  deskripsi?: string;
  views?: number;
  distance?: number;
}

interface Props {
  wisataPopuler: Wisata[];
}

const Main = ({ wisataPopuler = [] }: Props) => {
  const [showRecommendations, setShowRecommendations] = useState(false);
  const [showMapModal, setShowMapModal] = useState(false);
  const [marker, setMarker] = useState<[number, number] | null>(null);
  const [locationText, setLocationText] = useState('');
  const [open, setOpen] = useState(false);
  const [selected, setSelected] = useState('Alam');
  const [, setShowEmptyMessage] = useState(false);
  const [recommendations, setRecommendations] = useState<Wisata[]>([]);

  interface Wisata {
    id: number;
    nama: string;
    lokasi: string;
    gambar: string;
    latitude: number;
    longitude: number;
    deskripsi?: string;
    views?: number;
    distance?: number;
  }

  useEffect(() => {
    AOS.init({ duration: 1000, once: false });
    const handleScroll = () => {
      const scrollTop = window.scrollY;
      const docHeight = document.body.scrollHeight;
      const winHeight = window.innerHeight;
```

```

        const scrolled = scrollTop / (docHeight - winHeight);

        if (scrolled > 0.7) {
            setShowEmptyMessage(true);
        }
    };

    window.addEventListener('scroll', handleScroll);
    return () => window.removeEventListener('scroll', handleScroll);
}, []);

function FixMapSize() {
    const map = useMap();
    useEffect(() => {
        // Pastikan map dirender setelah modal benar-benar tampil
        const timeout = setTimeout(() => {
            map.invalidateSize();
        }, 500);
        return () => clearTimeout(timeout);
    }, [map]);
    return null;
}

function LocationMarker({
    setMarker,
    setLocationText,
}): {
    setMarker: React.Dispatch<React.SetStateAction<[number, number] |
null>>;
    setLocationText: React.Dispatch<React.SetStateAction<string>>;
} {
    useMapEvents({
        click(e: LeafletMouseEvent) {
            const { lat, lng } = e.latlng;
            setMarker([lat, lng]);
            setLocationText(`Lat:  ${lat.toFixed(5)},  Lng:
${lng.toFixed(5)}`);
        },
    });
    return null;
}

function getDistance(lat1: number, lon1: number, lat2: number, lon2:
number) {
    const R = 6371; // km
    const dLat = (lat2 - lat1) * (Math.PI / 180);
    const dLon = (lon2 - lon1) * (Math.PI / 180);
    const a =
        Math.sin(dLat / 2) * Math.sin(dLat / 2) +
        Math.cos(lat1 * (Math.PI / 180)) * Math.cos(lat2 * (Math.PI /
180)) * Math.sin(dLon / 2) * Math.sin(dLon / 2);
    const c = 2 * Math.atan2(Math.sqrt(a), Math.sqrt(1 - a));
    return R * c; // jarak dalam km
}

const handleExplore = async () => {
    if (!marker) {
        alert('Silakan pilih lokasi terlebih dahulu.');
```

```

    type WisataAPI = {
      id: number;
      nama: string;
      latitude: string;
      longitude: string;
      deskripsi?: string;
      gambar?: string;
    };

    try {
      const res = await
fetch(`http://127.0.0.1:8000/api/wisata/jenis/${selected}`);
      const json = await res.json();

      const data: WisataAPI[] = json.data || [];

      // Mapping data
      const mapped = data.map((item) => ({
        id: item.id,
        name: item.nama,
        lat: parseFloat(item.latitude),
        lng: parseFloat(item.longitude),
        desc: item.deskripsi || 'Tidak ada deskripsi',
        imageUrl: item.gambar ?
`http://127.0.0.1:8000/storage/${item.gambar}` : '/images/noimage.jpg',
      }));

      // Hitung jarak dari user ke masing-masing wisata
      const withDistance = mapped.map((rec) => ({
        ...rec,
        distance: getDistance(marker[0], marker[1], rec.lat, rec.lng),
      }));

      // Urutkan dari jarak terdekat ke terjauh
      // Urutkan dari jarak terdekat
      const sorted = withDistance.sort((a, b) => a.distance -
b.distance);

      // Ambil 5 terdekat
      const nearestFive = sorted.slice(0, 5);

      // Mapping ke format Wisata
      const mappedNearest: Wisata[] = nearestFive.map((item) => ({
        id: item.id,
        nama: item.name,
        lokasi: item.desc,
        gambar: item.imageUrl,
        latitude: item.lat,
        longitude: item.lng,
        deskripsi: item.desc,
        distance: item.distance,
      }));

      setRecommendations(mappedNearest);
    } catch (error) {
      console.error('Gagal mengambil data wisata:', error);
      setRecommendations([]);
    }

    setShowRecommendations(true);
  };

```

```

const routeByJenis = (jenis: string) => {
  switch (jenis) {
    case 'Alam':
      return '/wisata-alam';
    case 'Buatan dan Taman':
      return '/wisata-buatan';
    case 'Budaya dan Sejarah':
      return '/wisata-budaya';
    case 'Religi':
      return '/wisata-religi';
    default:
      return '/';
  }
};

return (
  <MainLayout>
    {/* Hero Section */}
    <section className="relative -mt-14 bg-cover bg-center pb-36"
style={{ backgroundImage: "url('/images/gslamet.png')" }}>
      <div className="absolute inset-0 bg-sky-800 opacity-70"></div>
      <div className="mx-auto -mt-6.5 flex max-w-7xl items-center
justify-between px-8 pt-95">
        {/* Left Content */}
        <div className="max-w-xl">
          <h1 className="mb-4 text-5xl leading-tight font-
extrabold text-white" data-aos="fade-up">
            Jelajahi Keindahan <span className="text-sky-
300">Banyumas</span>
          </h1>
          <p className="mb-4 font-semibold text-white" data-
aos="fade-up">
            Banyumas memiliki banyak destinasi wisata menarik
dari alam, budaya, taman hingga religi. Saatnya berpetualang dan temukan
keindahan lokal yang memukau!
          </p>
          <p className="font-semibold text-white" data-
aos="fade-up">
            Masih bingung mau wisata ke mana? Pilih jenis
wisata dan lokasi kamu, nanti sistem akan merekomendasikan lokasi wisata
terdekat dan yang mungkin kamu sukai.
          </p>
        </div>
      </div>

    {/* Explore Form - diposisikan di perbatasan biru-putih */}
    <div className="absolute bottom-0 left-1/2 z-10 w-full max-
w-5xl -translate-x-1/2 translate-y-1/2 rounded-xl bg-white shadow-lg ring-1
ring-black/10">
      <div className="flex flex-col justify-between px-6 py-6
sm:flex-row sm:items-center sm:gap-6">
        {/* Jenis Wisata */}
        <div className="relative mb-4 flex flex-col sm:mb-
0">
          <label className="mb-1 text-sm font-extrabold
text-sky-800">Jenis Wisata</label>
          <div className="relative">
            <button
              type="button"
              onClick={() => setOpen(!open)}

```

```

                                className="flex w-48 items-center justify-
between rounded-md border border-sky-500 bg-white px-4 py-2 text-sm font-
semibold text-gray-900 shadow-sm transition focus:ring-2 focus:ring-sky-600
focus:outline-none"
                                >
                                    <span>{selected}</span>
                                    <ChevronDown className={`h-4 w-4
transition-transform ${open ? 'rotate-180' : ''}`} />
                                </button>

                                {open && (
                                    <div className="absolute z-20 mt-1 w-48
rounded-md border border-sky-500 bg-white font-black shadow-lg">
                                        {['Alam', 'Buatan dan Taman', 'Budaya
dan Sejarah', 'Religi'].map((opt) => (
                                            <button
                                                key={opt}
                                                onClick={() => {
                                                    setSelected(opt);
                                                    setOpen(false);
                                                }}
                                                className={`block w-full px-
4 py-2 text-left text-sm font-medium ${
                                                    selected === opt ? 'bg-
sky-100 text-sky-800' : 'font-black text-gray-900 hover:bg-sky-50'
                                                }}`
                                            >
                                                {opt}
                                            </button>
                                        ))}
                                    </div>
                                )}
                            </div>
                        </div>

                        {/* Lokasi Awal */}
                        <div className="relative mb-4 flex flex-col sm:mb-
0">
                            <label className="mb-1 text-sm font-extrabold
text-sky-800">Lokasi Awal</label>
                            <input
                                type="text"
                                placeholder="Klik untuk pilih lokasi di peta"
                                value={locationText}
                                readOnly
                                onClick={() => setShowMapModal(true)}
                                className="cursor-pointer rounded-md border
border-sky-400 bg-gray-50 px-4 py-2 text-sm text-gray-900 shadow-sm
focus:ring-2 focus:ring-sky-600 focus:outline-none"
                            />
                        </div>

                        {/* Modal Map Fullscreen */}
                        {showMapModal && (
                            <div className="fixed inset-0 z-50 flex items-
center justify-center bg-black/60">
                                <div className="w-full max-w-5xl overflow-
hidden rounded-lg bg-white shadow-2xl">
                                    {/* Header Modal */}
                                    <div className="flex items-center justify-
between bg-white px-6 py-4 text-center">

```

```

<h2 className="text-lg font-semibold
text-sky-600">Pilih Lokasi Anda</h2>
<button onClick={() =>
setShowMapModal(false)} className="text-xl font-bold text-sky-600
hover:text-sky-800">
X
</button>
</div>

{/* Map Container */}
<div className="relative h-[80vh] w-
full">
<MapContainer
center={[-7.4314, 109.2478]}
zoom={12}
scrollWheelZoom={true}
className="z-0 h-full w-full
border border-black"
>
<FixMapSize />
<TileLayer
attribution='&copy; <a
href="https://osm.org/copyright">OpenStreetMap</a>'
url="https://{s}.tile.opens
treetmap.org/{z}/{x}/{y}.png"
/>
{marker && <Marker position={marker}
/>}
<LocationMarker
setMarker={setMarker} setLocationText={setLocationText} />
</MapContainer>

{/* Tombol Konfirmasi Lokasi */}
<div className="absolute bottom-4
left-1/2 -translate-x-1/2">
<button
onClick={() =>
setShowMapModal(false)}
className="rounded-md bg-
sky-800 px-6 py-3 text-sm font-semibold text-white shadow-md hover:bg-sky-
700"
>
Konfirmasi Lokasi
</button>
</div>
</div>
</div>
</div>
)}

{/* Tombol Explore */}
<div className="flex justify-center sm:justify-end">
<button
onClick={handleExplore}
className="mt-2 rounded-md bg-sky-600 px-6
py-3 text-sm font-semibold text-white transition hover:bg-sky-800"
>
Explore Now
</button>
</div>
</div>

```

```

        </div>
    </section>

    {/ * Rekomendasi Wisata */}
    {showRecommendations && (
        <section className="shadow-in bg-white px-6 py-25 pt-23 pb-
2" data-aos="fade-up">
            {/ * Header */}
            <div className="mb-4 flex items-center justify-between">
                <button onClick={() => setShowRecommendations(false)}
className="text-xl font-bold text-sky-600 hover:text-sky-800">
                    X
                </button>
            </div>
            {/ * JIKA DATA KOSONG */}
            {recommendations.length === 0 ? (
                <div className="flex flex-col items-center justify-
center py-20 text-center">
                    {/ * Icon atau ilustrasi */}
                    <svg
                        xmlns="http://www.w3.org/2000/svg"
                        className="mb-6 h-24 w-24 text-gray-400"
                        fill="none"
                        viewBox="0 0 24 24"
                        stroke="currentColor"
                        strokeWidth={1.5}
                    >
                        <path
                            strokeLinecap="round"
                            strokeLinejoin="round"
                            d="M3 17a4 4 0 014-4h10a4 4 0 014 4M7
7a5 5 0 1010 0 5 5 0 00-10 0z"
                        />
                    </svg>

                    <h3 className="mb-2 text-2xl font-bold text-
gray-700">Wisata Belum Ada</h3>

                    <p className="max-w-md text-gray-500">
                        Maaf, data wisata untuk kategori ini belum
tersedia. Coba pilih jenis wisata lain untuk melihat rekomendasi terdekat.
                    </p>
                </div>
            ) : (
                <>
                    {/ * MAP */}
                    <div className="mb-6 h-[400px] w-full overflow-
hidden rounded-xl shadow-lg">
                        <MapContainer center={[-7.4314, 109.2478]}
zoom={12} scrollWheelZoom={true} className="h-full w-full">
                            <TileLayer attribution="&copy;
OpenStreetMap" url="https://{s}.tile.openstreetmap.org/{z}/{x}/{y}.png" />

                            {marker && <Marker
position={marker}></Marker>}

                            {recommendations.map((rec) => (
                                <Marker key={rec.id}
position={[rec.latitude, rec.longitude]}>
                                    <Popup>
                                        <strong>{rec.nama}</strong>

```

```

                                <br />
                                {rec.deskripsi}
                            </Popup>
                        </Marker>
                    )))

                    {marker && recommendations.map((rec) =>
<RouteLine start={marker} end=[[rec.latitude, rec.longitude]] />)}
                </MapContainer>
            </div>

            {/* CARD */}
            <div className="relative">
                <h2 className="mb-4 text-2xl font-extrabold
text-sky-800">Wisata Terdekat dari Lokasi Kamu</h2>
                <div className="scrollbar-hide flex gap-6
overflow-x-auto pb-4">
                    {recommendations.map((rec) => (
                        <div
                            key={rec.id}
                            className="max-w-[300px] min-w-
[300px] flex-shrink-0 overflow-hidden rounded-xl border border-gray-200 bg-
white shadow-md transition hover:shadow-lg"
                        >
                            {/* Gambar */}
                            <img src={rec.gambar} alt={rec.nama}
className="h-40 w-full object-cover" />

                            {/* Isi Card */}
                            <div className="p-4">
                                <h3 className="text-lg font-
bold text-sky-800">{rec.nama}</h3>
                                <p className="mt-2 line-
clamp-3 text-sm text-gray-700">{rec.deskripsi}</p>

                                {/* Tombol Detail */}
                                <a
                                    href={`\wisata/${rec.id}`}
                                    className="mt-4 inline-
block rounded-md bg-sky-600 px-4 py-2 text-sm font-semibold text-white
hover:bg-sky-800"
                                >
                                    Lihat Detail
                                </a>
                            </div>
                        </div>
                    )))
                </div>
            </div>
        </>
    ))
    {/* WISATA POPULER */}
    <section className="mt-12">
        <div className="relative">
            <h2 className="mb-4 text-2xl font-extrabold text-
sky-800">Wisata Populer</h2>

            <div className="scrollbar-hide flex gap-6
overflow-x-auto pb-4">
                {wisataPopuler.map((item) => (
                    <div

```

```

                                key={item.id}
                                className="max-w-[300px] min-w-
[300px] flex-shrink-0 overflow-hidden rounded-xl border border-gray-200 bg-
white shadow-md transition hover:shadow-lg"
                                >
                                { /* Gambar */ }
                                <img
                                src={item.gambar ?
`/storage/${item.gambar}` : '/images/default.jpg'}
                                alt={item.nama}
                                className="h-40 w-full object-
cover"
                                />

                                { /* Isi Card */ }
                                <div className="p-4">
                                <h3 className="text-lg font-bold
text-sky-800">{item.nama}</h3>

                                <p className="text-sm text-gray-
500">{item.lokasi}</p>

                                <p className="mt-1 text-xs text-
gray-400">{item.views} views</p>

                                { /* Tombol Detail */ }
                                <a
                                href={` /wisata/${item.id}` }
                                className="mt-4 inline-block
rounded-md bg-sky-600 px-4 py-2 text-sm font-semibold text-white hover:bg-
sky-800"
                                >
                                Lihat Detail
                                </a>
                                </div>
                                </div>
                                )))
                                </div>
                                </div>
                                </section>

                                <div className="mt-12 text-center">
                                <a
                                href={routeByJenis(selected)}
                                className="inline-block rounded-lg bg-sky-700
px-8 py-3 text-lg font-bold text-white hover:bg-sky-800"
                                >
                                Lihat Wisata Lainnya →
                                </a>
                                </div>
                                </section>
                                )}

                                { /* About Section */ }
                                <section className="mt-[-10px] bg-white px-6 py-24">
                                <div className="mx-auto max-w-5xl text-center">
                                <h2 className="mb-6 text-3xl font-extrabold text-sky-
800">Tentang Banyumas</h2>
                                <p className="text-justify text-lg leading-relaxed font-
black text-gray-600">

```

```

        Banyumas adalah permata tersembunyi di Jawa Tengah
yang kaya akan pesona alam, budaya, dan sejarah. Terletak di kaki Gunung
        Slamet, daerah ini memiliki udara sejuk dan pemandangan
hijau yang memanjakan mata. Dari curug-curug yang memukau seperti
        Curug Cipendok, hingga keindahan Lokawisata Baturraden
yang menjadi magnet wisata keluarga dan pecinta alam, Banyumas adalah
        destinasi yang layak dijelajahi oleh siapa pun yang
mendambakan liburan penuh petualangan dan relaksasi.
        <br />
        <br />
        Tidak hanya menyajikan keindahan alam, Banyumas juga
menyimpan kekayaan budaya seperti tradisi lengger dan batik Banyumasan
        yang unik. Kuliner khas seperti mendoan, soto, dan
gethuk goreng siap memanjakan lidah pengunjung. Banyumas bukan hanya
        tempat untuk berwisata, melainkan pengalaman yang
menyatu antara keindahan alam, kearifan lokal, dan keramahan masyarakatnya.
        <br />
        <br />
        Jadi, tunggu apa lagi? Mari temukan cerita dan
keajaiban yang hanya bisa kamu rasakan langsung di Banyumas!
        </p>
    </div>
</section>
<Footer />
</MainLayout>
    );
};

export default Main;

```

Lampiran Source Code 2

```

import { router, useForm } from '@inertiajs/react';
import { ArrowLeft } from 'lucide-react';
import React from 'react';

export default function LoginPage() {
    const { data, setData, processing, errors } = useForm({
        nama: '',
        password: '',
    });

    const handleSubmit = (e: React.FormEvent) => {
        e.preventDefault();
        router.post('/login', data, {
            onSuccess: () => {
                router.visit('/admindash/dashadmin');
            },
        });
    };

    const goBack = () => {
        router.visit('/');
    };

    return (
        <div className="flex h-screen items-center justify-center
bg-sky-400">
            <div className="flex w-4/5 max-w-5xl overflow-hidden
rounded-lg bg-white shadow-lg">

```

```

        <div className="relative w-1/2">
            

            <button
                onClick={goBack}
                className="absolute top-1/2 right-[-24px]
-translate-y-1/2 rounded-full bg-sky-800 p-3 text-white shadow-
lg transition hover:bg-sky-700"
            >
                <ArrowLeft size={20} />
            </button>
        </div>

        <div className="flex w-1/2 flex-col items-center
justify-center bg-sky-800 p-10">
            <h2 className="mb-2 text-2xl font-bold text-
white">WANDER BANYUMAS</h2>
            <p className="mb-6 text-sm text-white">Admin
Only</p>

            <form onSubmit={handleSubmit} className="w-
full max-w-sm">
                <div className="mb-4">
                    <label className="mb-2 block text-sm
text-white">Username</label>
                    <input
                        type="text"
                        value={data.nama}
                        onChange={(e) => setData('nama',
e.target.value)}
                        placeholder="Enter your username"
                        className="w-full rounded-md
border border-gray-300 px-4 py-2 text-black placeholder-gray-400
focus:ring-2 focus:ring-white focus:outline-none"
                    />
                    {errors.nama && <p className="mt-1
text-sm text-red-400">{errors.nama}</p>}
                </div>

                <div className="mb-4">
                    <label className="mb-2 block text-sm
text-white">Password</label>
                    <input
                        type="password"
                        value={data.password}
                        onChange={(e) => setData('password',
e.target.value)}
                        placeholder="Enter your password"
                        className="w-full rounded-md
border border-gray-300 px-4 py-2 text-black placeholder-gray-400
focus:ring-2 focus:ring-white focus:outline-none"
                    />
                    {errors.password && <p className="mt-
1 text-sm text-red-400">{errors.password}</p>}
                </div>

                <button
                    type="submit"
                    disabled={processing}

```

```

                                className="w-full rounded-full bg-
white py-2 font-semibold text-sky-800 shadow hover:bg-gray-100"
                                >
                                    {processing ? 'Loading...' : 'ENTER'}
                                </button>
                            </form>
                        </div>
                    </div>
                </div>
            </div>
        );
    }
}

```

Lampiran Source Code 3

```

import AdminLayout from '@layouts/AdminLayout';
import { Link } from '@inertiajs/react';

interface DashboardProps {
    totalWisata: number;
    totalViews: number;
}

export default function DashAdmin({ totalWisata, totalViews }:
DashboardProps) {
    return (
        <AdminLayout>
            { /* Header */ }
            <div className="mb-8 flex items-center justify-
between">
                <div>
                    <h1 className="text-3xl font-bold text-sky-
800">Dashboard Admin</h1>
                    <p className="text-gray-500">Selamat datang
di halaman admin</p>
                </div>

                <Link href={route('datawisata.create')}
className="rounded-lg bg-sky-600 px-5 py-2 text-white shadow-md
transition hover:bg-sky-700">
                    + Tambah Wisata
                </Link>
            </div>

            { /* Konten Dashboard */ }
            <div className="grid grid-cols-1 gap-6 md:grid-cols-
2">
                <div className="rounded-xl bg-white p-6 shadow-
sm">
                    <h3 className="text-sm font-medium text-gray-
500">Total Wisata</h3>
                    <p className="mt-2 text-3xl font-bold text-
sky-600">{totalWisata}</p>
                </div>

                <div className="rounded-xl bg-white p-6 shadow-
sm">

```

```

                <h3 className="text-sm font-medium text-gray-500">Total Pengunjung</h3>
                <p className="mt-2 text-3xl font-bold text-emerald-600">{totalViews}</p>
            </div>
        </div>
    </AdminLayout>
);
}

```

Lampiran Source Code 4

```

import AdminLayout from '@layouts/AdminLayout';
import { Link, router, usePage } from '@inertiajs/react';
import { useState } from 'react';

interface Wisata {
  id: number;
  nama: string;
  lokasi: string;
  gambar: string | null;
  deskripsi?: string;
  jenis_wisata?: string;
}

export default function DataWisata({ wisata }: { wisata: Wisata[] }) {
  const { url } = usePage();

  // Ambil query filter jenis wisata
  const currentQuery = new URLSearchParams(url.split('?')[1]);
  const selectedJenis = currentQuery.get('jenis') || '';

  // Sorting: none | az | za
  const [sortOrder, setSortOrder] = useState<string>('none');

  // Hapus data
  const handleDelete = (id: number) => {
    if (confirm('Yakin ingin menghapus data ini?')) {
      router.delete(route('admin.wisata.destroy', id), {
        onSuccess: () => alert('Data wisata berhasil dihapus!'),
        onError: () => alert('Gagal menghapus data wisata.'),
      });
    }
  };

  // Filter jenis wisata
  const handleFilterChange = (value: string) => {
    const query = value ? `?jenis=${value}` : '';
    router.visit(`/admindash/datawisata${query}`);
  };

  // Sorting final, hanya berjalan jika sortOrder bukan 'none'
  const finalData = [...wisata];

  if (sortOrder === 'az') {

```

```

        finalData.sort((a, b) => a.nama.localeCompare(b.nama));
    } else if (sortOrder === 'za') {
        finalData.sort((a, b) => b.nama.localeCompare(a.nama));
    }

    return (
        <AdminLayout>
            {/* FILTER DAN SORT */}
            <div className="mb-6 flex flex-col gap-4 md:flex-row
md:items-center md:justify-between">
                <div className="mt-3 flex flex-wrap items-center
gap-6">
                    {/* FILTER JENIS WISATA */}
                    <select
                        className="flex w-48 items-center justify-
between rounded-md border border-sky-500 bg-white px-4 py-2 text-
sm font-semibold text-gray-900 shadow-sm transition focus:ring-2
focus:ring-sky-600 focus:outline-none"
                        value={selectedJenis}
                        onChange={ (e) =>
handleFilterChange(e.target.value)}
                    >
                        <option value="">Semua Jenis
Wisata</option>
                        <option value="Alam">Alam</option>
                        <option value="Buatan dan Taman">Buatan
dan Taman</option>
                        <option value="Budaya dan Sejarah">Budaya
dan Sejarah</option>
                        <option value="Religi">Religi</option>
                    </select>

                    {/* SORTING RADIO */}
                    <div className="flex items-center gap-4 text-
sm font-semibold text-gray-700">
                        {/* NONE */}
                        <label className="flex cursor-pointer
items-center gap-2">
                            <input
                                type="radio"
                                name="sort"
                                value="none"
                                checked={sortOrder === 'none'}
                                onChange={ () =>
setSortOrder('none')}
                                className="h-4 w-4 accent-blue-
600"
                            />
                            Tidak aktif
                        </label>

                        {/* A-Z */}
                        <label className="flex cursor-pointer
items-center gap-2">
                            <input
                                type="radio"
                                name="sort"
                                value="az"
                                checked={sortOrder === 'az'}
                                onChange={ () => setSortOrder('az')}

```

```

        className="h-4 w-4 accent-blue-
600"
        />
        A - Z
    </label>

    {/* Z-A */}
    <label className="flex cursor-pointer
items-center gap-2">
        <input
            type="radio"
            name="sort"
            value="za"
            checked={sortOrder === 'za'}
            onChange={() => setSortOrder('za')}
            className="h-4 w-4 accent-blue-
600"
            />
            Z - A
        </label>
    </div>
</div>

    {/* TOMBOL AKSI */}
    <div className="flex items-center gap-4">
        <Link href={route('datawisata.create')}
className="rounded-lg bg-blue-600 px-5 py-2 text-white shadow-md
hover:bg-blue-700">
            + Tambah Wisata
        </Link>

        <Link href={route('datawisata.import')}
className="rounded-lg bg-green-600 px-5 py-2 text-white shadow-
md hover:bg-green-700">
            Import CSV
        </Link>
    </div>
</div>

    {/* GRID DATA */}
    <div className="grid grid-cols-1 gap-6 sm:grid-cols-2
md:grid-cols-3">
        {finalData.map((item: Wisata) => (
            <div key={item.id} className="group relative
rounded-xl border border-gray-100 bg-white p-4 shadow-md">
                <img src={`\storage/${item.gambar}`}
alt={item.nama} className="h-40 w-full rounded-lg object-cover"
/>

                <h3 className="mt-3 font-semibold text-
gray-800">{item.nama}</h3>
                <p className="text-sm text-gray-
600">{item.lokasi}</p>
                <span className="mt-1 inline-block text-
xs text-slate-500">{item.jenis_wisata}</span>

                {/* TOMBOL EDIT & HAPUS */}
                <div className="absolute top-2 right-2
opacity-0 transition group-hover:opacity-100">
                    <button

```

```

                                onClick={() =>
router.visit(route('admin.wisata.edit', item.id))}
                                className="mx-1 rounded bg-blue-
500 px-2 py-1 text-xs text-white hover:bg-blue-600"
                                >
                                Edit
                                </button>

                                <button
                                onClick={() =>
handleDelete(item.id)}
                                className="mx-1 rounded bg-red-
500 px-2 py-1 text-xs text-white hover:bg-red-600"
                                >
                                Hapus
                                </button>
                                </div>
                                </div>
                                </div>
                                )))
                                </div>
                                </AdminLayout>
                                );
                                }

```

Lampiran Source Code 5

```

import AdminLayout from '@layouts/AdminLayout';
import { useForm } from '@inertiajs/react';
import React, { useState } from 'react';

export default function DataWisataCreate() {
  const { data, setData, post, reset, errors } = useForm({
    nama: '',
    jenis_wisata: '',
    deskripsi: '',
    lokasi: '',
    latitude: '',
    longitude: '',
    gambar: null as File | null,
    jam_operasional: '',
    harga_tiket: '',
    fasilitas: '',
    maps_url: '',
  });

  const [preview, setPreview] = useState<string | null>(null);

  const handleSubmit = (e: React.FormEvent) => {
    e.preventDefault();

    const formData = new FormData();
    Object.entries(data).forEach(([key, value]) => {
      if (value !== null && value !== undefined) {
        if (value instanceof File) {
          formData.append(key, value);
        } else {
          formData.append(key, String(value));
        }
      }
    });
  };
}

```

```

    });

    post(route('admin.wisata.store'), {
        forceFormData: true,
        onSuccess: () => {
            reset();
            setPreview(null);
            alert('Data wisata berhasil disimpan!');
        },
        onError: () => alert('Gagal menyimpan data wisata.'),
    });
};

return (
    <AdminLayout>
        { /* FORM TAMBAH DATA */ }
        <div className="mx-auto max-w-4xl rounded-2xl bg-white p-8 shadow-md">
            <h1 className="mb-6 text-2xl font-semibold text-sky-800">Form Tambah Wisata</h1>

            <form onSubmit={handleSubmit}
                encType="multipart/form-data" className="grid grid-cols-1 gap-6 md:grid-cols-2">
                { /* Nama Wisata */ }
                <div className="flex flex-col">
                    <label htmlFor="nama" className="text-sm font-bold text-gray-800">
                        Nama Wisata
                    </label>
                    <input
                        id="nama"
                        type="text"
                        value={data.nama}
                        onChange={(e) => setData('nama', e.target.value)}
                        className="rounded-lg border border-gray-300 p-2 font-bold text-gray-800 focus:ring-2 focus:ring-blue-400 focus:outline-none"
                        required
                    />
                    {errors.nama && <p className="text-sm text-red-500">{errors.nama}</p>}
                </div>

                { /* Latitude */ }
                <div className="flex flex-col">
                    <label htmlFor="latitude" className="mb-1 text-sm font-bold text-gray-800">
                        Latitude
                    </label>
                    <input
                        id="latitude"
                        type="text"
                        value={data.latitude}
                        onChange={(e) => setData('latitude', e.target.value)}
                        className="rounded-lg border border-gray-300 p-2 font-bold text-gray-900 focus:ring-2 focus:ring-blue-400 focus:outline-none"
                    />
                </div>
            </form>
        </div>
    </AdminLayout>
);

```

```

        />
    </div>

    { /* Longitude */}
    <div className="flex flex-col">
        <label htmlFor="longitude" className="mb-
1 text-sm font-bold text-gray-700">
            Longitude
        </label>
        <input
            id="longitude"
            type="text"
            value={data.longitude}
            onChange={(e) => setData('longitude',
e.target.value)}
            className="rounded-lg border border-
gray-300 p-2 font-bold text-gray-900 focus:ring-2 focus:ring-
blue-400 focus:outline-none"
        />
    </div>

    { /* Jenis Wisata */}
    <div className="flex flex-col">
        <label htmlFor="jenis_wisata"
className="mb-1 text-sm font-bold text-gray-800">
            Jenis Wisata
        </label>
        <select
            id="jenis_wisata"
            value={data.jenis_wisata || ''}
            onChange={(e) => setData('jenis_wisata',
e.target.value)}
            className="rounded-lg border border-
gray-300 p-2 font-bold text-gray-900 focus:ring-2 focus:ring-
blue-400 focus:outline-none"
        >
            <option value="">Pilih jenis
wisata</option>
            <option value="Alam">Alam</option>
            <option value="Buatan">Buatan dan
Taman</option>
            <option value="Budaya">Budaya dan
Taman</option>
            <option value="Religi">Religi</option>
        </select>
    </div>

    { /* Jam Buka */}
    <div className="flex flex-col">
        <label className="mb-1 text-sm font-bold
text-gray-800">Jam Operasional</label>
        <textarea
            rows={4}
            className="rounded-lg border border-
gray-300 p-2 font-bold text-gray-900 focus:ring-2 focus:ring-
blue-400 focus:outline-none"
            value={data.jam_operasional}
            onChange={(e) =>
setData('jam_operasional', e.target.value)}
        ></textarea>
    </div>

```

```

        {/* Harga Tiket */}
        <div className="flex flex-col">
            <label className="mb-1 text-sm font-bold
text-gray-800">Harga Tiket</label>
            <textarea
                className="rounded-lg border border-
gray-300 p-2 font-bold text-gray-900 focus:ring-2 focus:ring-
blue-400 focus:outline-none"
                rows={4}
                value={data.harga_tiket}
                onChange={(e) => setData('harga_tiket',
e.target.value)}
            ></textarea>
        </div>

        {/* Fasilitas */}
        <div className="flex flex-col">
            <label className="mb-1 text-sm font-bold
text-gray-800">Fasilitas</label>
            <textarea
                className="rounded-lg border border-
gray-300 p-2 font-bold text-gray-900 focus:ring-2 focus:ring-
blue-400 focus:outline-none"
                rows={4}
                value={data.fasilitas}
                onChange={(e) => setData('fasilitas',
e.target.value)}
            ></textarea>
        </div>

        {/* Google Maps URL */}
        <div className="flex flex-col">
            <label htmlFor="maps_url" className="mb-
1 text-sm font-bold text-gray-800">
                Link Google Maps
            </label>
            <input
                id="maps_url"
                type="text"
                value={data.maps_url}
                onChange={(e) => setData('maps_url',
e.target.value)}
                className="rounded-lg border border-
gray-300 p-2 font-bold text-gray-900 focus:ring-2 focus:ring-
blue-400 focus:outline-none"
                placeholder="Tempel link Google Maps"
            />
            {errors.maps_url && <p className="text-
sm text-red-500">{errors.maps_url}</p>}
        </div>

        {/* Lokasi */}
        <div className="flex flex-col">
            <label htmlFor="lokasi" className="mb-1
text-sm font-bold text-gray-800">
                Lokasi
            </label>
            <input
                id="lokasi"
                type="text"

```

```

        value={data.lokasi}
        onChange={(e) => setData('lokasi',
e.target.value)}
        className="rounded-lg border border-
gray-300 p-2 font-bold text-gray-900 focus:ring-2 focus:ring-
blue-400 focus:outline-none"
      />
    </div>

    {/* Deskripsi */}
    <div className="flex flex-col md:col-span-2">
      <label htmlFor="deskripsi" className="mb-
1 text-sm font-bold text-gray-700">
        Deskripsi
      </label>
      <textarea
        id="deskripsi"
        value={data.deskripsi}
        onChange={(e) => setData('deskripsi',
e.target.value)}
        className="rounded-lg border border-
gray-300 p-3 font-bold text-gray-900 focus:ring-2 focus:ring-
blue-400 focus:outline-none"
        placeholder="Tuliskan deskripsi singkat
tentang wisata..."
        rows={4}
      ></textarea>
    </div>

    {/* Gambar */}
    <div className="flex flex-col md:col-span-2">
      <label htmlFor="gambar" className="mb-1
text-sm font-bold text-gray-700">
        Gambar
      </label>
      <input
        id="gambar"
        type="file"
        onChange={(e) => {
          const file = e.target.files?.[0];
          setData('gambar', file || null);
          if (file)
setPreview(URL.createObjectURL(file));
        }}
        className="rounded-lg border border-
gray-300 p-2 font-bold text-gray-900 focus:ring-2 focus:ring-
blue-400 focus:outline-none"
      />
      {preview && <img src={preview} alt="Preview"
className="mt-3 w-48 rounded-lg border border-gray-200" />}
    </div>

    {/* Tombol Simpan */}
    <div className="flex justify-end md:col-span-
2">
      <button type="submit" className="rounded-
lg bg-blue-600 px-6 py-2 text-white shadow-md transition
hover:bg-blue-700">
        Simpan
      </button>
    </div>

```

```

        </form>
      </div>
    </AdminLayout>
  );
}

```

Lampiran Source Code 6

```

import AdminLayout from '@layouts/AdminLayout';
import { router, useForm, usePage } from '@inertiajs/react';
import React, { useEffect } from 'react';

interface Wisata {
  id: number;
  nama: string;
  jenis_wisata: string;
  deskripsi: string;
  lokasi: string;
  latitude: string;
  longitude: string;
  gambar: string;
  jam_operasional: string;
  harga_tiket: string;
  fasilitas: string;
  maps_url: string;
}

export default function DataWisataEdit() {
  const { props } = usePage<{ wisata?: Wisata }>();
  const wisata = props.wisata;
  console.log('Props dari Laravel:', props);

  const { data, setData, errors } = useForm({
    nama: '',
    jenis_wisata: '',
    deskripsi: '',
    lokasi: '',
    latitude: '',
    longitude: '',
    gambar: null as File | null,
    jam_operasional: '',
    harga_tiket: '',
    fasilitas: '',
    maps_url: '',
  });

  useEffect(() => {
    if (wisata) {
      setData({
        nama: wisata.nama || '',
        jenis_wisata: wisata.jenis_wisata || '',
        deskripsi: wisata.deskripsi || '',
        lokasi: wisata.lokasi || '',
        latitude: wisata.latitude || '',
        longitude: wisata.longitude || '',
        gambar: null,
        jam_operasional: wisata.jam_operasional || '',

```

```

        harga_tiket: wisata.harga_tiket || '',
        fasilitas: wisata.fasilitas || '',
        maps_url: wisata.maps_url || '',
    });
}
}, [wisata, setData]);

const handleSubmit = (e: React.FormEvent) => {
    e.preventDefault();

    if (!wisata?.id) {
        alert('Data wisata belum siap.');
```

return;

```
    }

    formData.append('_method', 'PUT');
```

// Kirim hanya field yang tidak kosong

```
    Object.entries(data).forEach(([key, value]) => {
        if (value !== '' && value !== null && value !==
undefined) {
            formData.append(key, value as Blob | string);
        }
    });

    console.log('Data yang dikirim ke server:',
Object.fromEntries(formData.entries()));

    // Tetap pakai POST, tapi Laravel akan anggap ini PUT
    (karena ada _method)
    router.post(route('admin.wisata.update', wisata.id),
formData, {
        onStart: () => console.log('Mengirim data ke
server...'),
        onSuccess: () => alert('Data wisata berhasil
diperbarui!'),
        onError: (errors) => {
            console.error('Gagal update:', errors);
            alert('Terjadi kesalahan saat memperbarui data.');
```

});

```
    });
};

if (!wisata) {
    return (
        <AdminLayout>
            <div className="flex min-h-screen items-center
justify-center text-gray-600">
                <p>Memuat data wisata...</p>
            </div>
        </AdminLayout>
    );
}

return (
    <AdminLayout>
        <div className="mx-auto max-w-4xl rounded-2xl bg-
white p-8 shadow-md">
            <h1 className="mb-6 text-2xl font-semibold text-
sky-800">Edit Data Wisata</h1>

```

```

                                <form    onSubmit={handleSubmit}
encType="multipart/form-data" className="grid grid-cols-1 gap-6
md:grid-cols-2">
    {/* Nama Wisata */}
    <div className="flex flex-col">
        <label className="text-sm font-bold text-
gray-800">Nama Wisata</label>
        <input
            type="text"
            value={data.nama}
            onChange={(e) => setData('nama',
e.target.value)}
            className="rounded-lg border border-
gray-300 p-2 font-semibold text-gray-800"
            required
        />
        {errors.nama && <p className="text-sm
text-red-500">{errors.nama}</p>}
    </div>

    {/* Latitude */}
    <div className="flex flex-col">
        <label className="text-sm font-bold text-
gray-800">Latitude</label>
        <input
            type="text"
            value={data.latitude}
            onChange={(e) => setData('latitude',
e.target.value)}
            className="rounded-lg border border-
gray-300 p-2 font-semibold text-gray-800"
        />
    </div>

    {/* Longitude */}
    <div className="flex flex-col">
        <label className="text-sm font-bold text-
gray-800">Longitude</label>
        <input
            type="text"
            value={data.longitude}
            onChange={(e) => setData('longitude',
e.target.value)}
            className="rounded-lg border border-
gray-300 p-2 font-semibold text-gray-800"
        />
    </div>

    {/* Jenis Wisata */}
    <div className="flex flex-col">
        <label className="text-sm font-bold text-
gray-800">Jenis Wisata</label>
        <select
            value={data.jenis_wisata}
            onChange={(e) => setData('jenis_wisata',
e.target.value)}
            className="rounded-lg border border-
gray-300 p-2 font-semibold text-gray-800"
        >
            <option value="">Pilih jenis
wisata</option>

```

```

        <option value="Alam">Alam</option>
        <option value="Buatan">Buatan dan
Taman</option>
        <option value="Budaya">Budaya dan
Sejarah</option>
        <option value="Religi">Religi</option>
    </select>
</div>

    { /* Jam Buka */ }
    <div className="flex flex-col">
        <label className="mb-1 text-sm font-bold
text-gray-800">Jam Operasional</label>
        <textarea
            rows={4}
            className="rounded-lg border border-
gray-300 p-2 font-bold text-gray-900 focus:ring-2 focus:ring-
blue-400 focus:outline-none"
            value={data.jam_operasional}
            onChange={(e) =>
setData('jam_operasional', e.target.value)}
        ></textarea>
    </div>

    { /* Harga Tiket */ }
    <div className="flex flex-col">
        <label className="mb-1 text-sm font-bold
text-gray-800">Harga Tiket</label>
        <textarea
            className="rounded-lg border border-
gray-300 p-2 font-bold text-gray-900 focus:ring-2 focus:ring-
blue-400 focus:outline-none"
            rows={4}
            value={data.harga_tiket}
            onChange={(e) => setData('harga_tiket',
e.target.value)}
        ></textarea>
    </div>

    { /* Fasilitas */ }
    <div className="flex flex-col">
        <label className="mb-1 text-sm font-bold
text-gray-800">Fasilitas</label>
        <textarea
            className="rounded-lg border border-
gray-300 p-2 font-bold text-gray-900 focus:ring-2 focus:ring-
blue-400 focus:outline-none"
            rows={4}
            value={data.fasilitas}
            onChange={(e) => setData('fasilitas',
e.target.value)}
        ></textarea>
    </div>

    { /* Google Maps URL */ }
    <div className="flex flex-col">
        <label htmlFor="maps_url" className="mb-
1 text-sm font-bold text-gray-800">
            Link Google Maps
        </label>
        <input

```

```

                                id="maps_url"
                                type="text"
                                value={data.maps_url}
                                onChange={(e) => setData('maps_url',
e.target.value)}}
                                className="rounded-lg border border-
gray-300 p-2 font-bold text-gray-900 focus:ring-2 focus:ring-
blue-400 focus:outline-none"
                                placeholder="Tempel link Google Maps
(optional)"
                                />
                                {errors.maps_url && <p className="text-
sm text-red-500">{errors.maps_url}</p>}
                                </div>

                                {/* Lokasi */}
                                <div className="flex flex-col">
                                    <label className="text-sm font-bold text-
gray-800">Lokasi</label>
                                    <input
                                        type="text"
                                        value={data.lokasi}
                                        onChange={(e) => setData('lokasi',
e.target.value)}}
                                        className="rounded-lg border border-
gray-300 p-2 font-semibold text-gray-800"
                                    />
                                </div>

                                {/* Deskripsi */}
                                <div className="flex flex-col md:col-span-2">
                                    <label className="text-sm font-bold text-
gray-800">Deskripsi</label>
                                    <textarea
                                        value={data.deskripsi}
                                        onChange={(e) => setData('deskripsi',
e.target.value)}}
                                        className="rounded-lg border border-
gray-300 p-2 font-semibold text-gray-800"
                                        rows={4}
                                    ></textarea>
                                </div>

                                {/* Gambar */}
                                <div className="flex flex-col md:col-span-2">
                                    <label className="text-sm font-bold text-
gray-800">Gambar</label>
                                    <input
                                        type="file"
                                        onChange={(e) => setData('gambar',
e.target.files?.[0] || null)}}
                                        className="rounded-lg border border-
gray-300 p-2 font-semibold text-gray-800"
                                    />
                                </div>

                                <div className="flex justify-end md:col-span-
2">
                                    <button type="submit" className="rounded-
lg bg-blue-600 px-6 py-2 text-white shadow-md transition
hover:bg-blue-700">

```

```

                Simpan Perubahan
            </button>
        </div>
    </form>
</div>
</AdminLayout>
);
}

```

Lampiran Source Code 7

```

import AdminLayout from '@layouts/AdminLayout';
import { router } from '@inertiajs/react';
import React, { useState } from 'react';

export default function ImportWisata() {
    const [file, setFile] = useState<File | null>(null);

    function handleSubmit(e: React.FormEvent) {
        e.preventDefault();

        if (!file) {
            alert('Pilih file CSV terlebih dahulu.');
```

return;

```
        }

        const formData = new FormData();
        formData.append('file', file);

        router.post(route('datawisata.import'), formData, {
            onSuccess: () => alert('Berhasil import CSV!'),
            onError: () => alert('Gagal import CSV.'),
        });
    }

    return (
        <AdminLayout>
            <div className="mx-auto max-w-xl rounded-xl bg-white
p-8 shadow">
                <h1 className="mb-4 text-2xl font-bold text-sky-
700">Import Data Wisata (CSV)</h1>

                <form onSubmit={handleSubmit}>
                    <input
                        type="file"
                        accept=".csv"
                        className="mb-2 w-full rounded border p-
2 text-gray-700"
                        onChange={(e) =>
setFile(e.target.files?.[0] || null)}
                    />

                    {file && <p className="mb-4 text-sm font-
medium text-green-700">File dipilih: {file.name}</p>}

                    <button type="submit" className="rounded-lg
bg-green-600 px-6 py-2 text-white hover:bg-green-700">

```

```

        Import Sekarang
      </button>
    </form>
  </div>
</AdminLayout>
);
}

```

Lampiran Source Code 8

```

import AdminLayout from '@layouts/AdminLayout';
import { Head } from '@inertiajs/react';
import { Bar, BarChart, CartesianGrid, ResponsiveContainer,
Tooltip, XAxis, YAxis } from 'recharts';

interface WisataStat {
  id: number;
  nama: string;
  views: number;
}

interface StatistikProps {
  topWisata: WisataStat[];
  totalWisata: number;
  totalViews: number;
}

export default function Statistik({ topWisata, totalWisata,
totalViews }: StatistikProps) {
  const rataRataViews = totalWisata > 0 ? Math.round(totalViews
/ totalWisata) : 0;

  const wisataTerpopuler = topWisata.length > 0 ? topWisata[0]
: null;

  return (
    <AdminLayout>
      <Head title="Statistik Wisata" />

      {/* SUMMARY (contoh, boleh kamu pisahkan nanti) */}
      <div className="mb-6 grid grid-cols-1 gap-6 md:grid-
cols-4">
        <div className="rounded-xl bg-white p-6 shadow">
          <p className="text-sm text-gray-500">Total
Wisata</p>
          <p className="mt-2 text-3xl font-bold text-
sky-600">{totalWisata}</p>
        </div>

        <div className="rounded-xl bg-white p-6 shadow">
          <p className="text-sm text-gray-500">Total
Pengunjung</p>
          <p className="mt-2 text-3xl font-bold text-
emerald-600">{totalViews.toLocaleString('id-ID')}</p>
        </div>

        <div className="rounded-xl bg-white p-6 shadow">

```

```

        <p className="text-sm text-gray-500">Rata-
rata Views</p>
        <p className="mt-2 text-3xl font-bold text-
indigo-600">{rataRataViews.toLocaleString('id-ID')}</p>
    </div>
    <div className="rounded-xl bg-white p-6 shadow">
        <p className="text-sm text-gray-500">Wisata
Terpopuler</p>

        {wisataTerpopuler ? (
            <>
                <p className="mt-2 text-lg font-bold
text-gray-800">{wisataTerpopuler.nama}</p>
                <p className="text-sm text-gray-
500">{wisataTerpopuler.views.toLocaleString('id-ID')} views</p>
            </>
        ) : (
            <p className="mt-2 text-sm text-gray-
400">Belum ada data</p>
        )}
    </div>
    </div>

    {/* CHART */}
    <div className="rounded-xl bg-white p-6 shadow">
        <h2 className="mb-4 text-lg font-bold text-gray-
700">Wisata Paling Banyak Dilihat</h2>

        <div className="h-[600px]">
            <ResponsiveContainer width="100%"
height="100%">
                <BarChart data={topWisata}
layout="vertical" margin={{ top: 10, right: 30, left: 100, bottom:
10 }}>
                    <CartesianGrid strokeDasharray="3 3"
/>
                    <XAxis type="number" />
                    <YAxis type="category" dataKey="nama"
width={180} />
                    <Tooltip />
                    <Bar dataKey="views" fill="#0284c7"
radius={[0, 6, 6, 0]} />
                </BarChart>
            </ResponsiveContainer>
        </div>
    </div>

    {/* TABLE RANKING */}
    <div className="mt-8 rounded-xl bg-white p-6 shadow">
        <h2 className="mb-4 text-lg font-bold text-gray-
700">Ranking Wisata Berdasarkan Views</h2>

        <div className="overflow-x-auto">
            <table className="w-full text-sm">
                <thead>
                    <tr className="border-b text-left
text-gray-500">
                        <th className="py-3">No</th>
                        <th>Nama Wisata</th>
                        <th className="text-
right">Views</th>

```

```

right">Kontribusi</th>
    </tr>
</thead>

    <tbody>
        {topWisata.map((item, index) => {
            const persen = totalViews > 0 ?
            ((item.views / totalViews) * 100).toFixed(1) : '0';

            return (
                <tr key={item.id}
                className="border-b last:border-0">
                    <td className="py-3 text-
                    gray-800">{index + 1}</td>
                    <td className="font-medium
                    text-gray-800">{item.nama}</td>
                    <td className="text-right
                    text-gray-800">{item.views.toLocaleString('id-ID')}</td>
                    <td className="text-right
                    text-gray-800">{persen}%</td>
                </tr>
            );
        })}
    </tbody>
</table>
</div>
</div>
</AdminLayout>
);
}

```