

Week 10

# Visualisasi Data

Arif Wicaksono Septyanto, S.Kom.,M.Kom

Sistem Informasi, Institut Teknologi Kalimantan

03/11/2025



# Mengapa Visualisasi Data Penting

- Membantu memahami pola dan tren dari data besar.
- Mempermudah pengambilan keputusan berbasis data.
- Meningkatkan efektivitas komunikasi hasil analisis.

# Mengapa Memilih Python

- Bahasa utama di bidang Data Science dan Visualisasi.
- Memiliki pustaka lengkap seperti Matplotlib, Seaborn, Plotly, dan Streamlit.
- Mudah dipelajari dan terintegrasi dengan berbagai database.

# Pustaka Populer untuk Visualisasi

- Matplotlib: grafik 2D dasar.
- Seaborn: visualisasi statistik yang indah.
- Plotly: grafik interaktif.
- Streamlit dan Dash: dashboard interaktif.
- Geopandas & Folium: visualisasi data geospasial.

# Miniconda dan Anaconda

- Mengelola paket dan environment Python secara terpisah.
- Menghindari konflik versi pustaka.
- Memudahkan instalasi pustaka kompleks seperti GIS atau Machine Learning.

# Kombinasi Ideal: Python + Miniconda

---

- Python untuk pemrograman utama.
- Miniconda untuk pengelolaan environment dan dependensi.
- Stabil, aman, dan lintas platform.

# Pengenalan Streamlit

- Framework Python untuk membuat aplikasi web interaktif.
- Tidak perlu HTML/CSS/JS.
- Cocok untuk dashboard data, machine learning, dan analisis.

# Instalasi Streamlit

- Gunakan perintah: pip install streamlit pandas matplotlib seaborn plotly.
- Verifikasi dengan perintah: streamlit hello.
- Buka di browser lokal otomatis.

# Struktur Program Streamlit

- File utama: app.py.
- Contoh kode sederhana menggunakan st.title() dan st.write().
- Jalankan dengan streamlit run app.py.

# Visualisasi Data Dasar

- Gunakan `bar_chart`, `line_chart`, dan `matplotlib`.
- `DataFrame` dapat ditampilkan langsung di Streamlit.
- Cocok untuk analisis donasi, kampanye, atau data lingkungan.

# Menambahkan Interaktivitas

- Gunakan selectbox untuk memilih jenis grafik.
- Gunakan slider untuk filter data.
- Tambahkan elemen dinamis sesuai kebutuhan pengguna.

# Visualisasi Geospasial

- Gunakan `st.map()` untuk menampilkan data lokasi.
- Cocok untuk data dengan latitude dan longitude.
- Dapat menampilkan persebaran lokasi kegiatan.

# Dashboard Sederhana

- Gunakan metric() untuk menampilkan angka penting.
- Gunakan progress() untuk menunjukkan pencapaian target.
- Kombinasikan grafik batang dan data tabel.

# Menambahkan Gambar dan Teks

---

- Gunakan `st.image()` untuk menampilkan gambar pendukung.
- Gunakan markdown untuk teks yang lebih terformat.
- Meningkatkan nilai presentasi dan estetika.

# Deployment ke Cloud

- Gunakan Streamlit Cloud untuk publikasi aplikasi.
- Hubungkan repository GitHub dan jalankan langsung di browser.
- Cocok untuk presentasi proyek akhir berbasis web.

# Integrasi dengan Database

- Gunakan SQLAlchemy untuk koneksi ke MySQL atau PostgreSQL.
- Gunakan pandas.read\_sql() untuk mengambil data.
- Data langsung divisualisasikan di dashboard Streamlit.

# Struktur Proyek Akhir

- Folder utama: app.py, config.py, requirements.txt, data/.
- Pisahkan logika, konfigurasi, dan data agar mudah dikelola.
- Struktur ini membuat proyek lebih profesional dan terukur.

# Index di Database

- Gunakan indeks untuk mempercepat query data.
- `CREATE INDEX idx_kampanye ON donasi(kampanye);`
- Mengoptimalkan performa dashboard saat menampilkan data besar.

# Index di Pandas

- Gunakan `set_index()` untuk mempermudah visualisasi.
- Index berguna untuk labeling otomatis grafik.
- Memudahkan filter dan manipulasi data.

# Index di Dashboard

- Buat indikator performa seperti Index Donasi dan Partisipasi.
- Tampilkan dalam bentuk progress bar atau metric.
- Meningkatkan pemahaman pengguna terhadap data.

# Kesimpulan

- Python dan Streamlit memudahkan pembuatan visualisasi interaktif.
- Integrasi database mendukung analisis data real-time.
- Cocok untuk proyek akhir Administrasi Basis Data.

# THANKS!

**arif.wicaksono@lecturer.itk.ac.id**  
**+62 852 1308 1309**