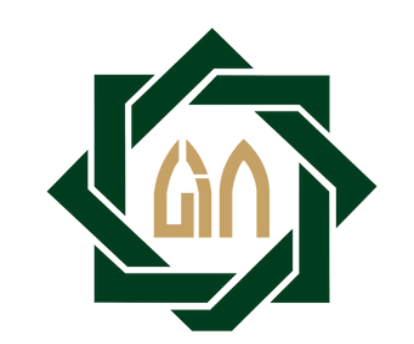
**LAPORAN PRAKTIKUM MATA KULIAH**

**LATIHAN VISUALISASI DATA DI WEB APP BERBASIS PHYTON MENGGUNAKAN STREAMLIT**

****

**Dosen Pengampu :**

**Khalid, M.Kom.**

**Disusun Oleh :**

**MUHAMMAD FARHAN MAHESWARA 09010623010**

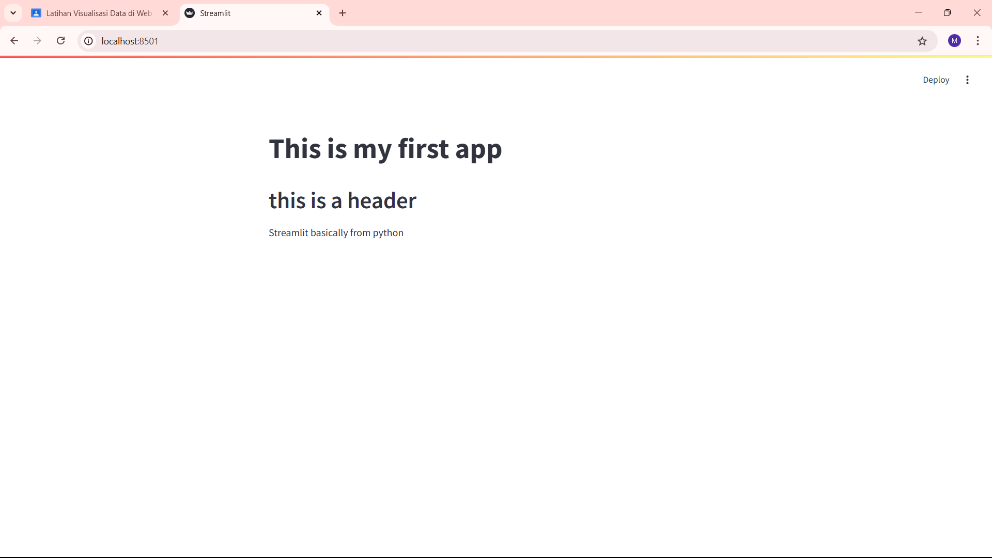
**PROGRAM STUDI SISTEM INFORMASI FAKULTAS SAINS DAN TEKNOLOGI**

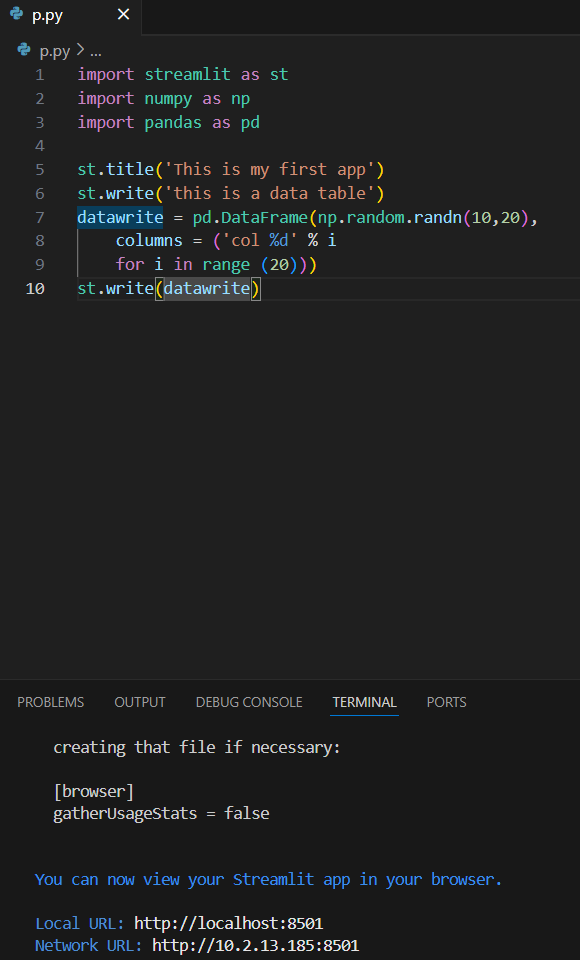
**UNIVERSITAS ISLAM NEGERI SUNAN AMPEL SURABAYA 2025**

# Data Visualization dengan Streamlit – Part I

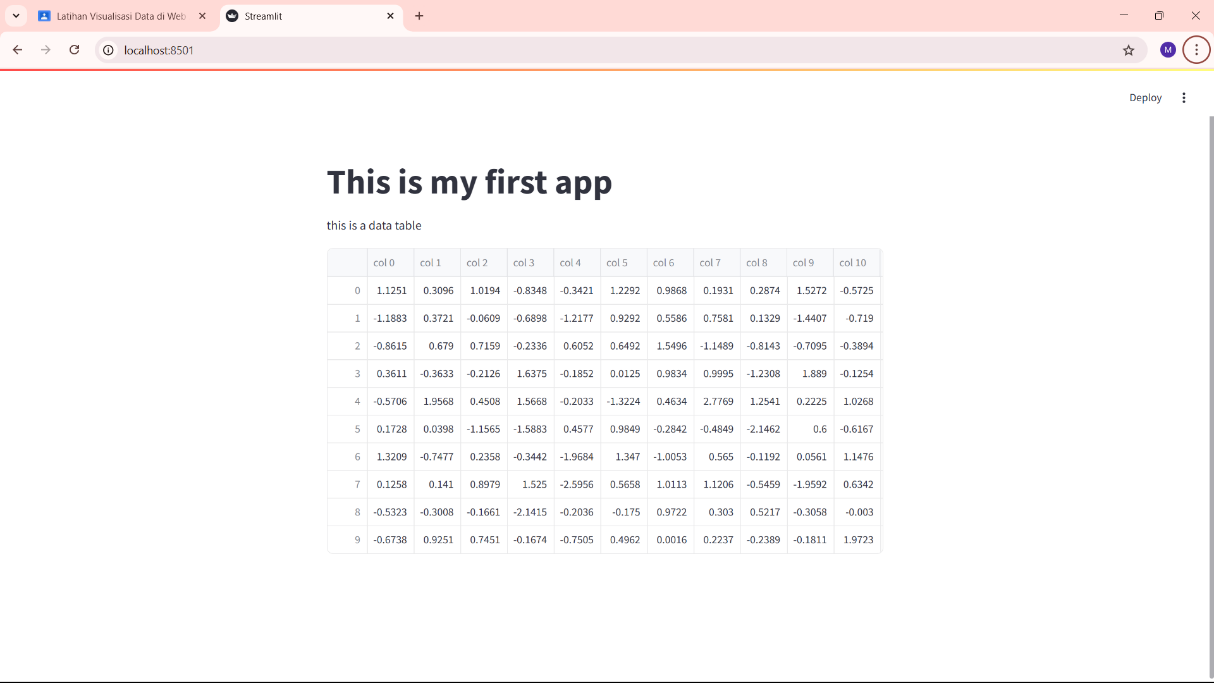
1. ****Add Text Data

Berikut hasilnya :

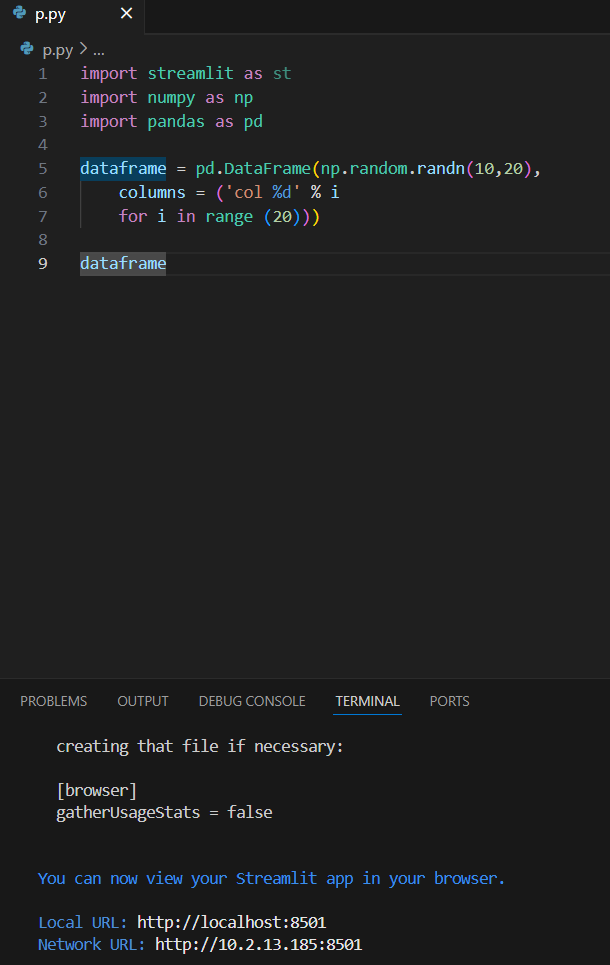


1. z`Display Data

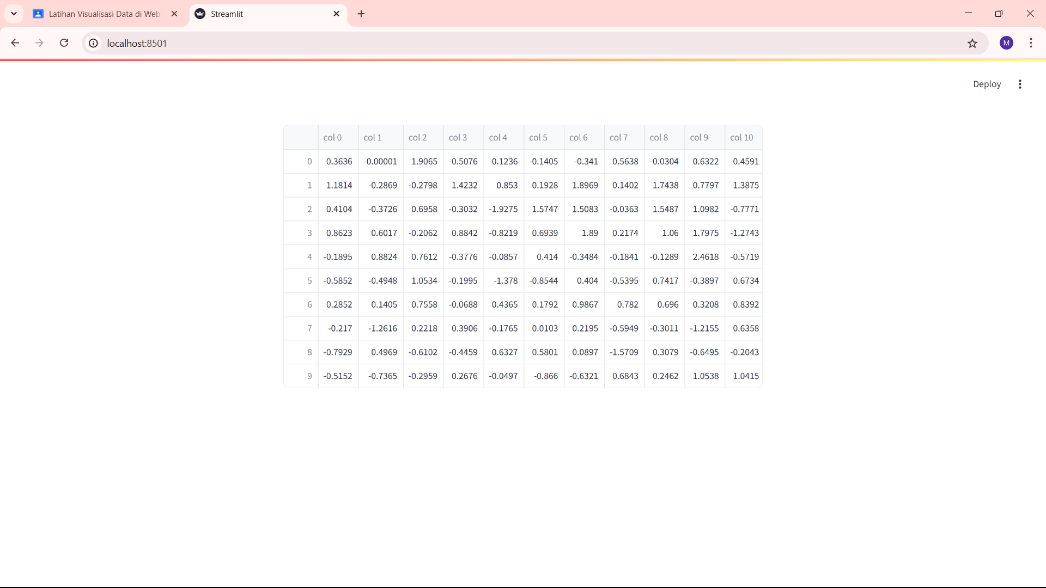
Berikut hasilnya :

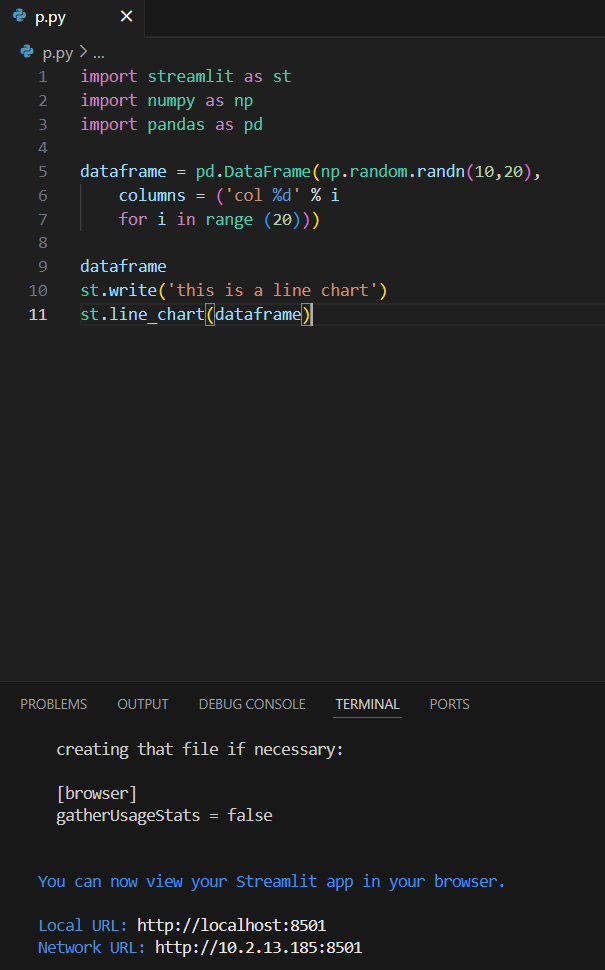


Sebenarnya, kita tidak selalu harus menggunakan fungsi **st.write** untuk menampilkan data di Streamlit. Streamlit mendukung fitur yang disebut ***magic commands*** atau perintah ajaib.

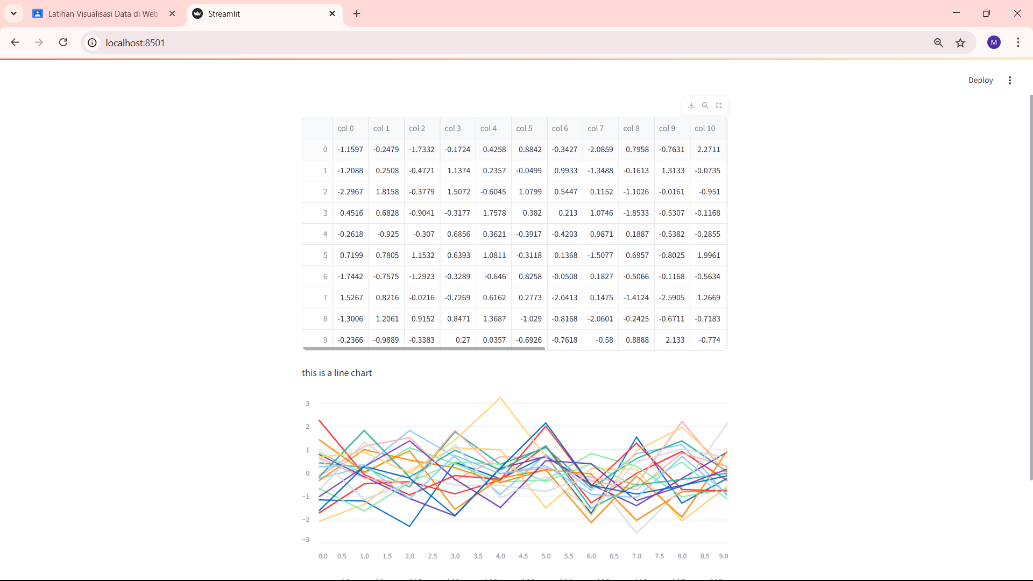
Dengan fitur ini, kita bisa langsung mengetik nama variabel seperti **(dataframe)**, dan Streamlit secara otomatis akan menampilkannya tanpa perlu dipanggil melalui fungsi khusus. Berikut contoh kode nya :

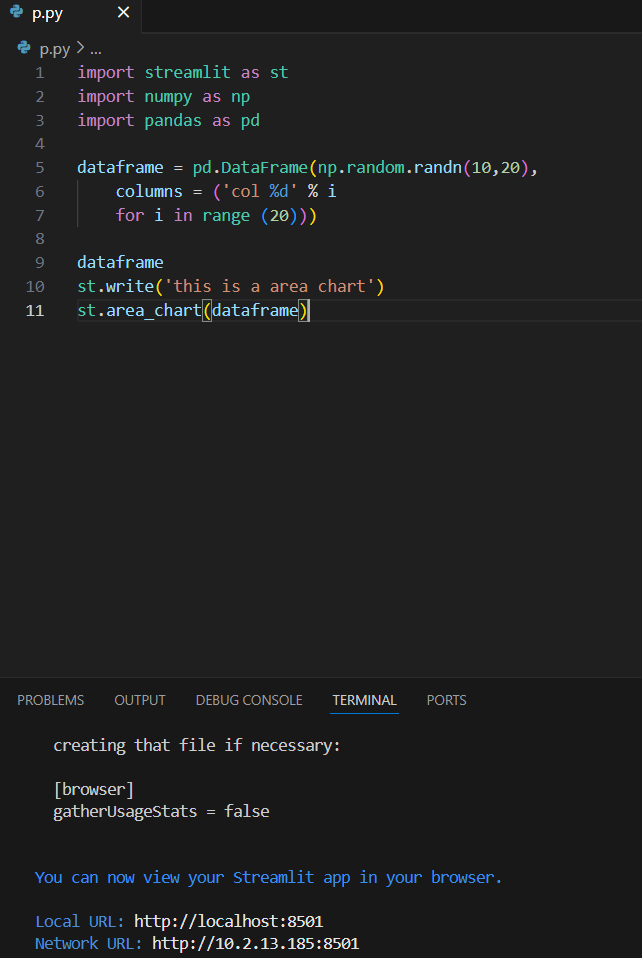
Berikut hasilnya :



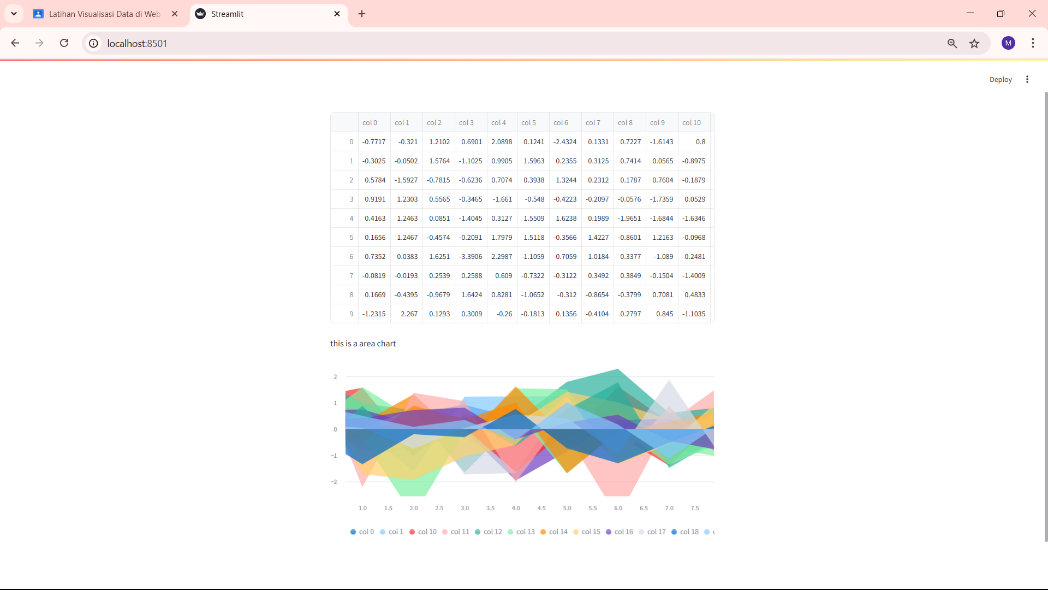
1. **Visualization of The Chart**
   1. Menggunakan line\_chart di Streamlit untuk menampilkan grafik berupa garis

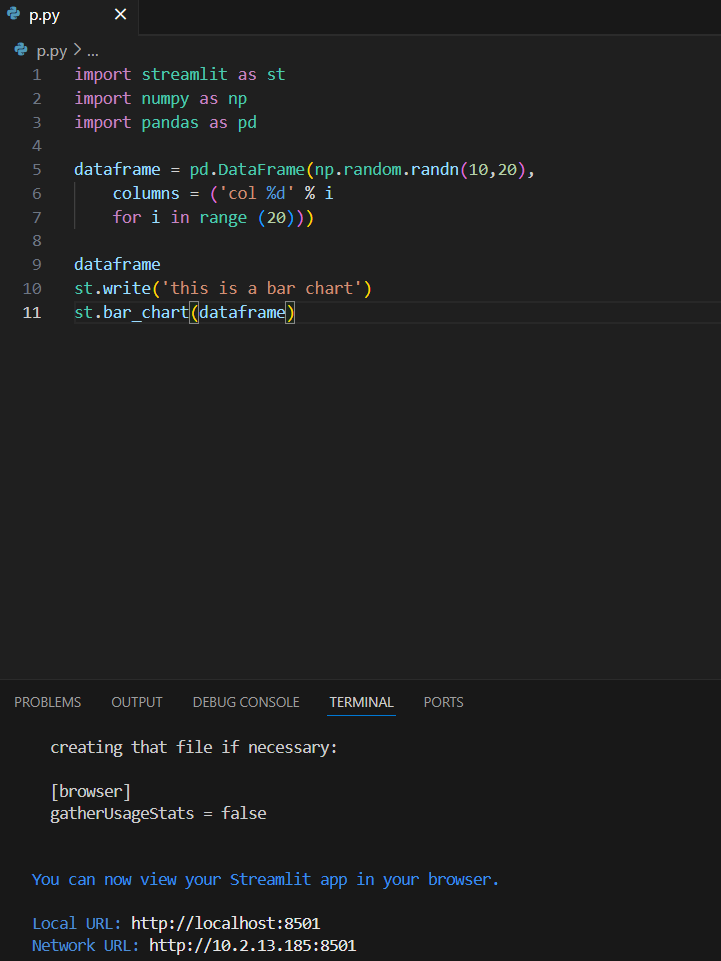
Berikut hasilnya :



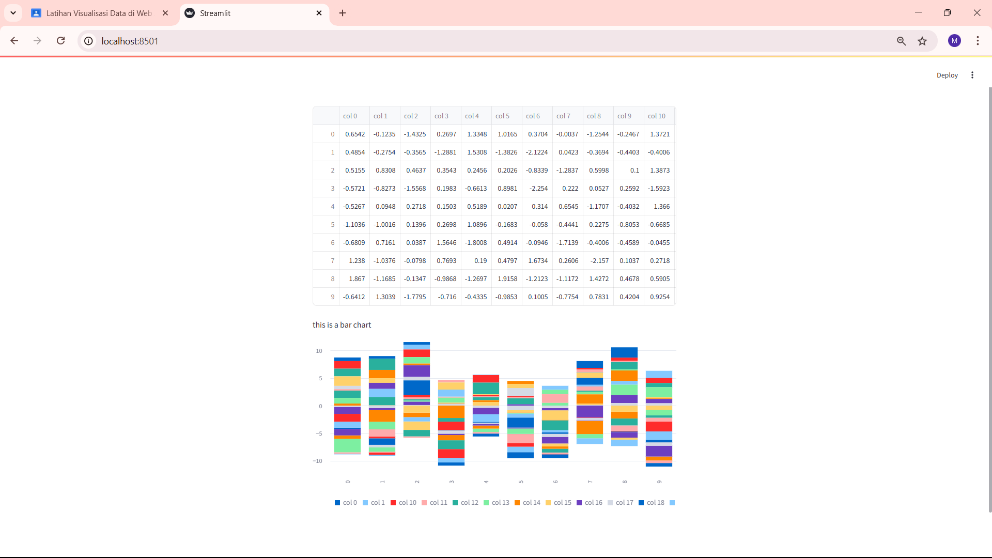
* 1. Menggunakan area\_chart di Streamlit untuk menampilkan grafik berupa area

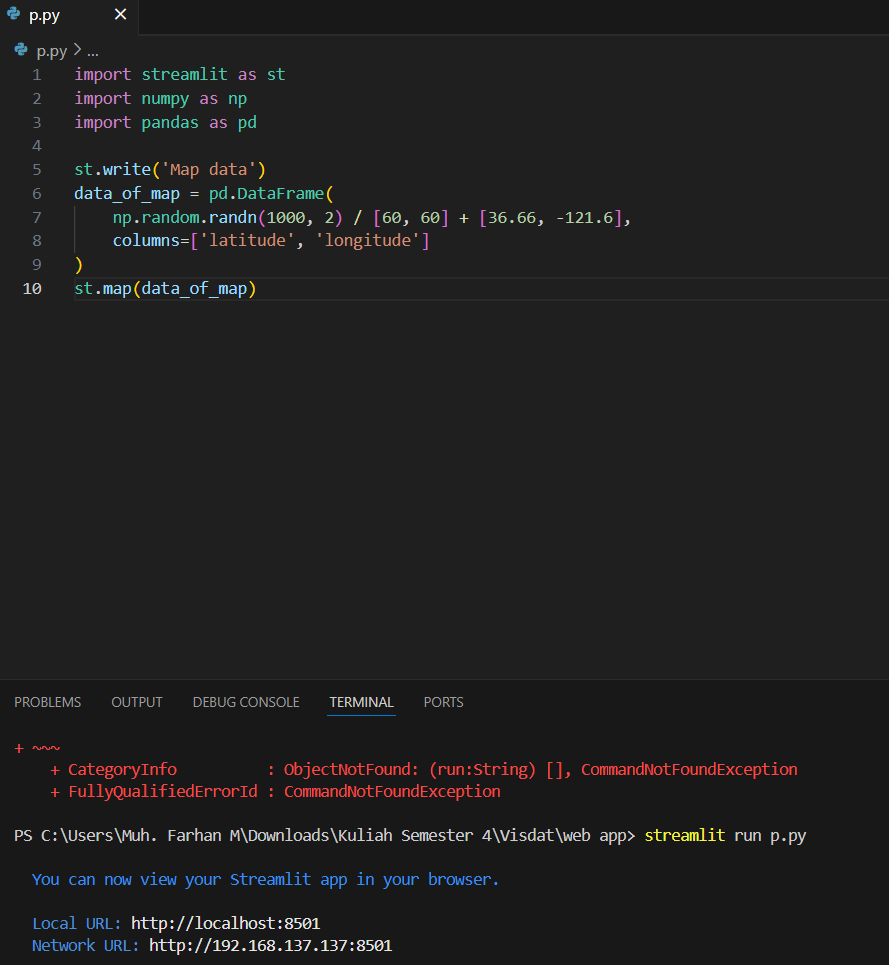
Berikut hasilnya :



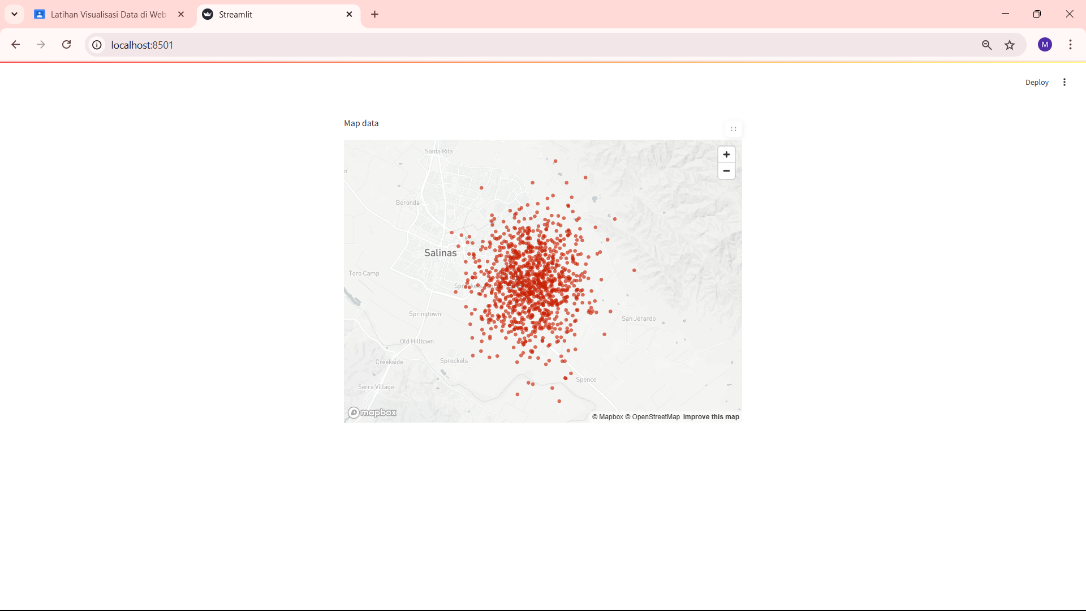
* 1. Menggunakan bar\_chart di Streamlit untuk menampilkan grafik histogram

Berikut hasilnya :

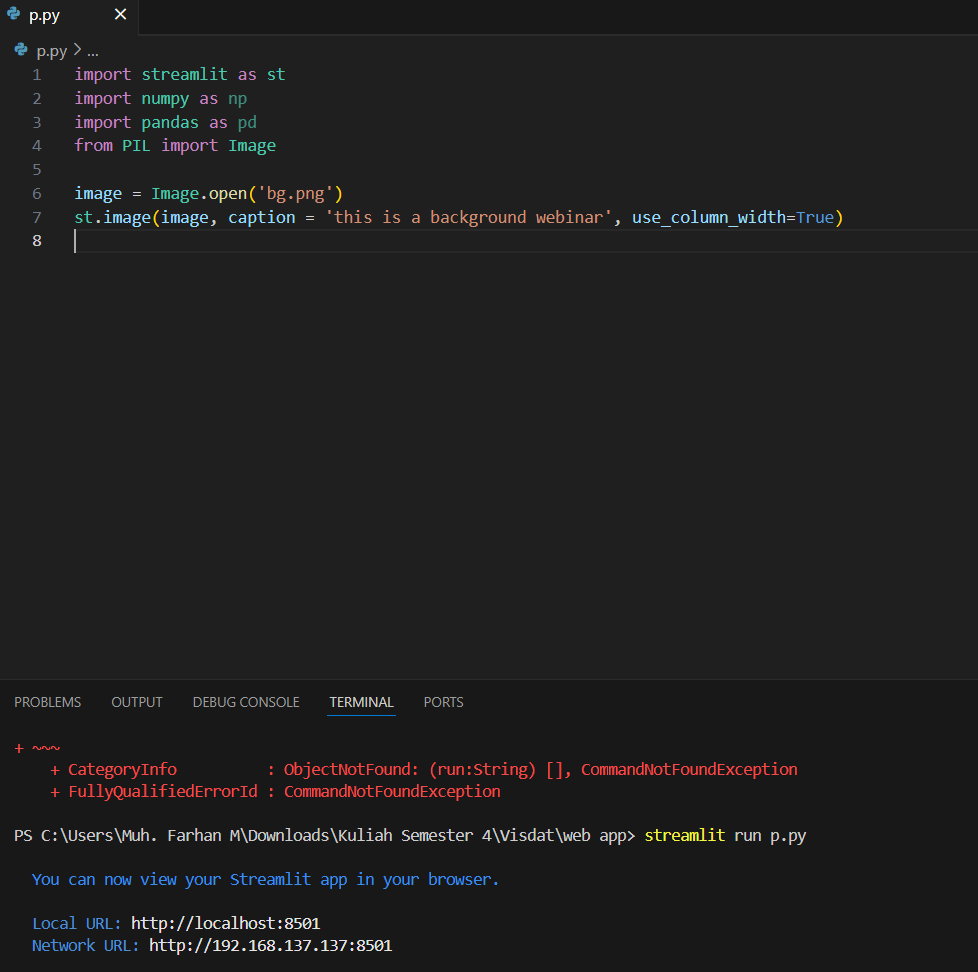


* 1. Menggunakan **st.map()** untuk menampilkan titik data pada sebuah peta

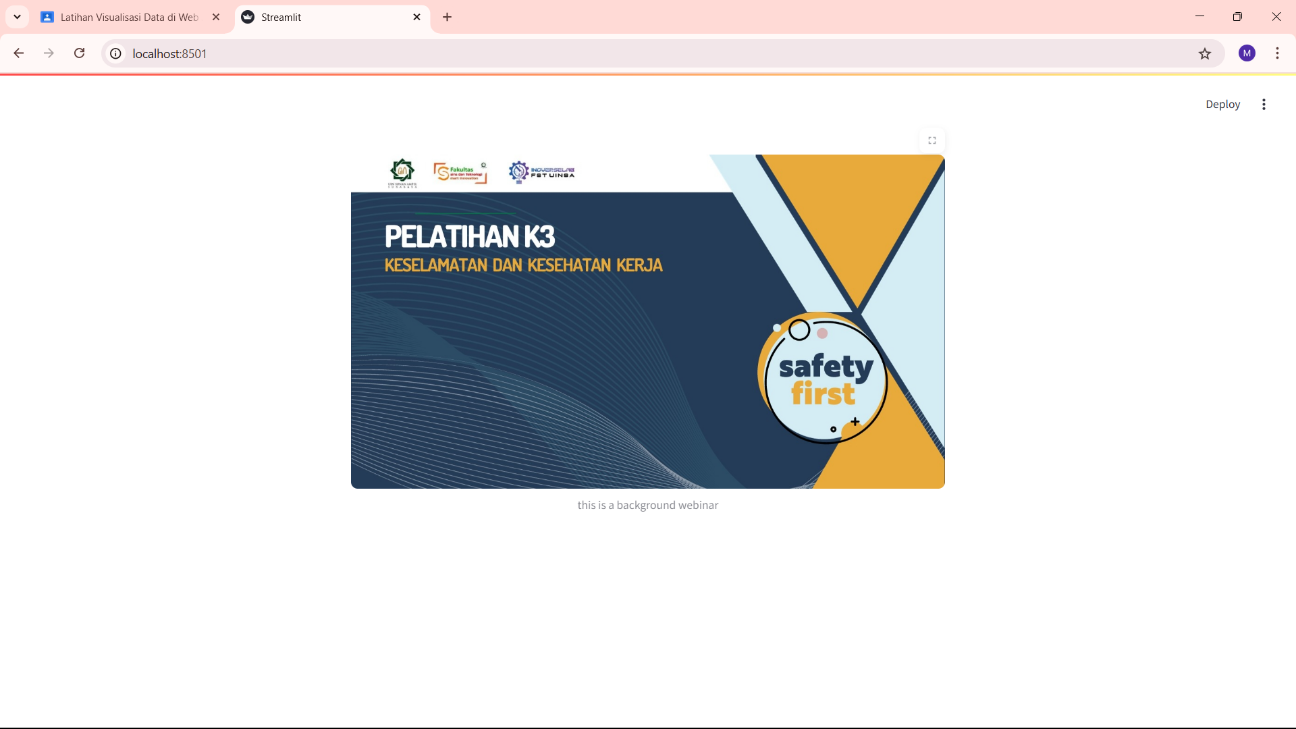
Berikut hasilnya :



1. Display Pictures



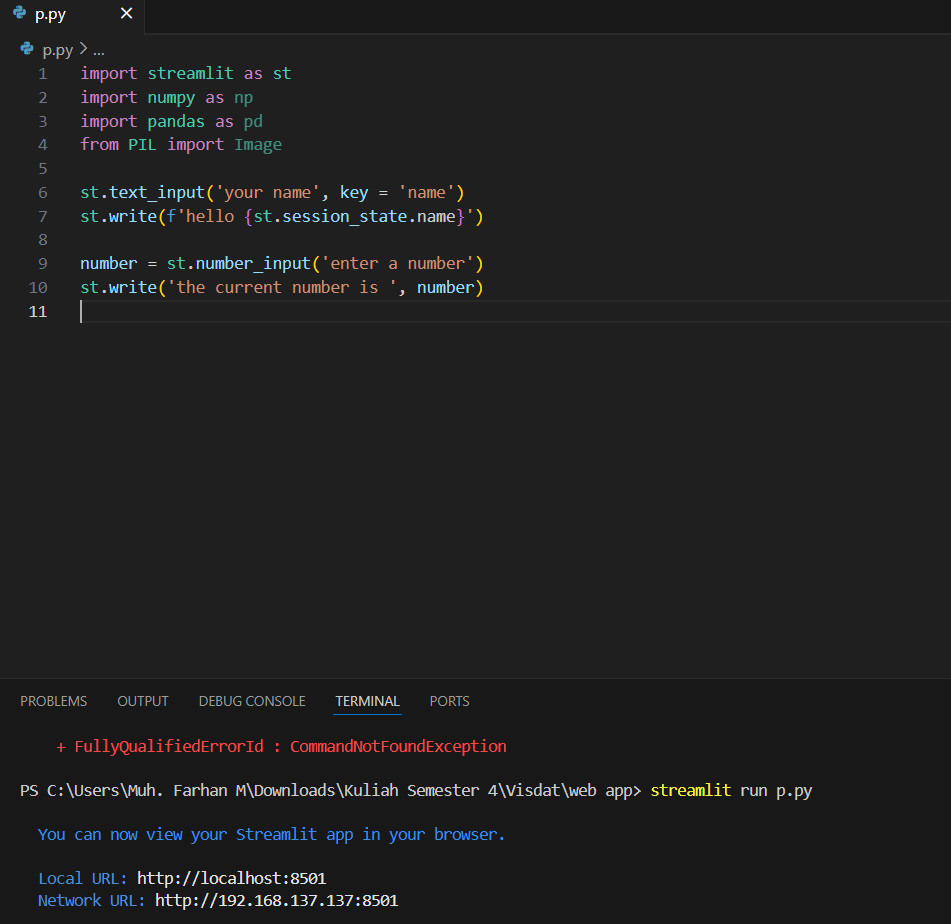
Berikut hasilnya :



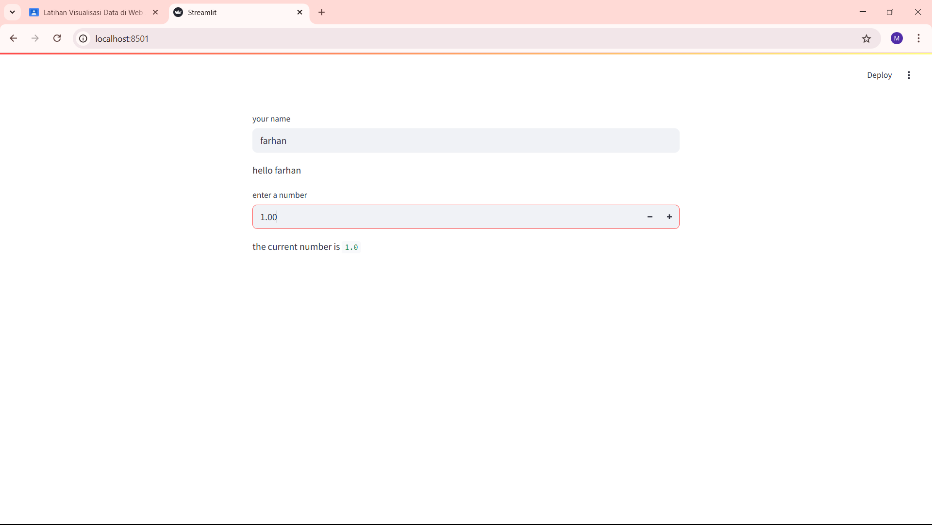
# Data Visualization with Streamlit – Part II

## WIDGETS

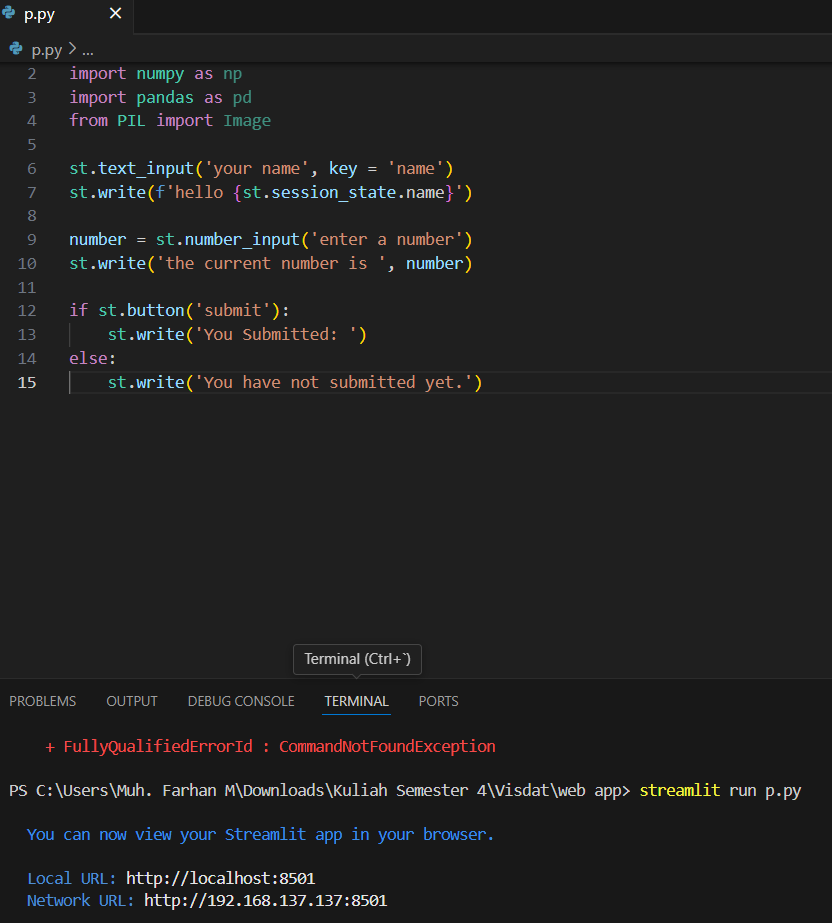
### Streamlit.Text\_Input/Streamlit.Number\_Input

Metode **streamlit.text\_input** digunakan untuk menampilkan kotak input teks yang hanya satu baris. Metode ini akan mengembalikan isi dari kotak input teks yang dimasukkan pengguna. Berikut contoh kode nya :

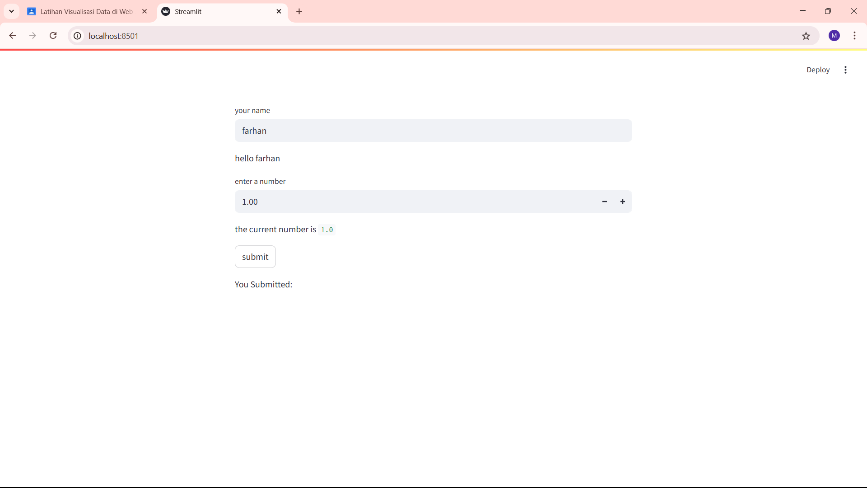
Berikut hasilnya :



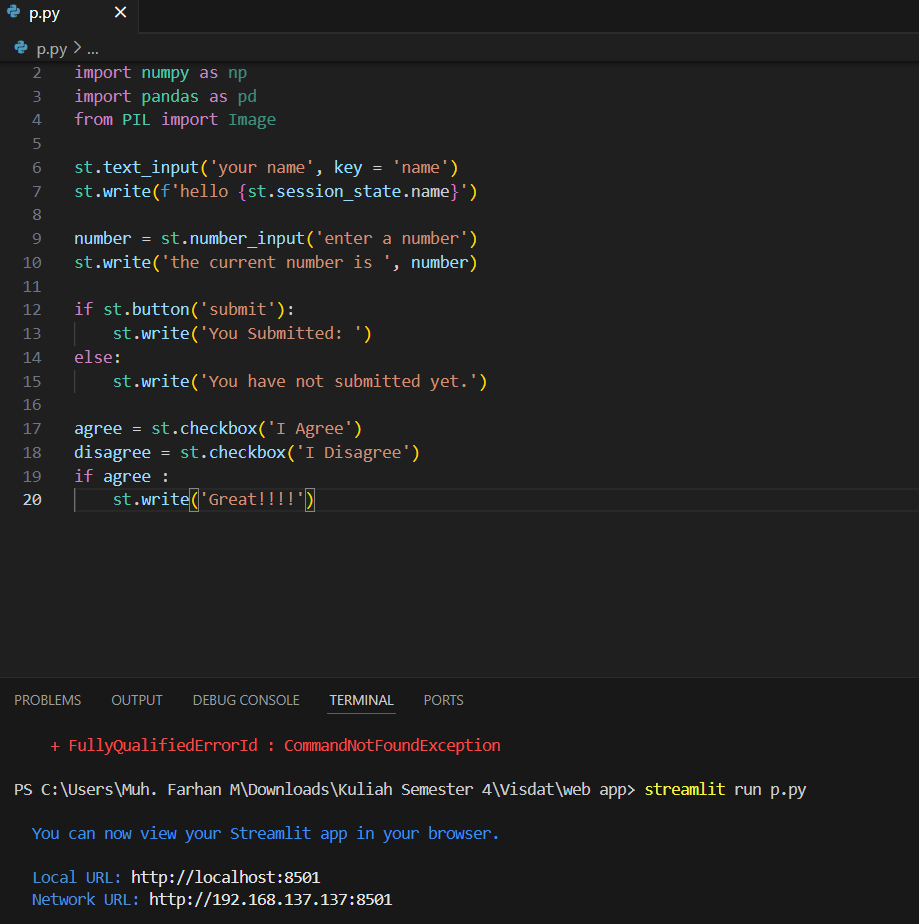
### Streamlit.Button

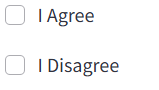
Metode **streamlit.button** mengembalikan nilai True atau False, yang menunjukkan apakah tombol telah diklik pada siklus aplikasi terakhir. Berikut contoh kode nya :

Berikut hasilnya setelah di klik tombol Submit :

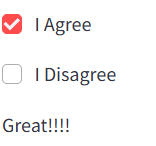


### Streamlit.Checkbox

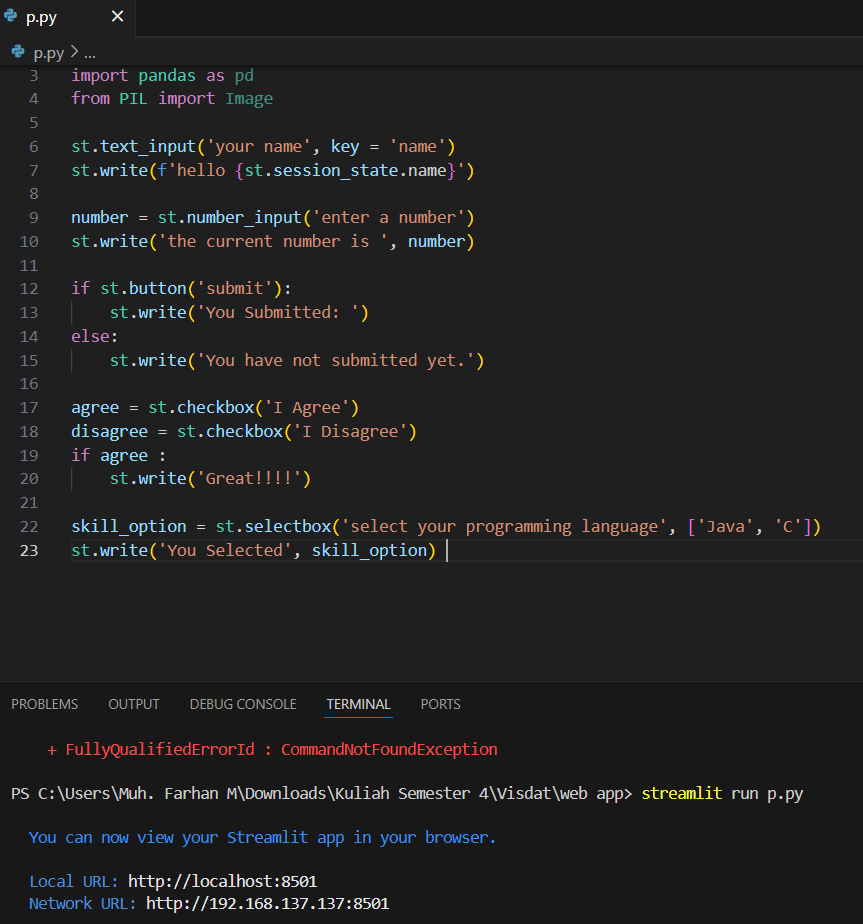
Metode kotak centang mengembalikan nilai True atau False, yang menunjukkan apakah kotak centang telah dipilih. Berikut contoh kode nya :

Berikut hasilnya sebelum di centang :

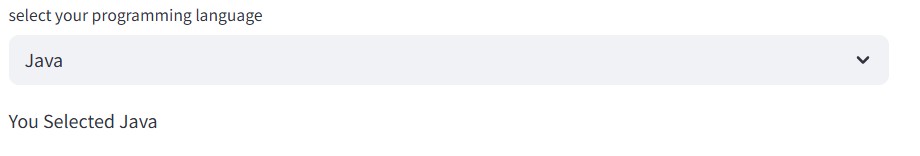
Hasil setelah di centang :



### Streamlit.Selectbox

Metode kotak pilih mengembalikan opsi yang telah dipilih oleh pengguna. Berikut contoh kode nya :

Berikut hasilnya :



### Streamlit.Slider

Metode penggeser mengembalikan nilai saat ini dari widget penggeser. Berikut adalah contoh kode nya :

Berikut hasilnya :



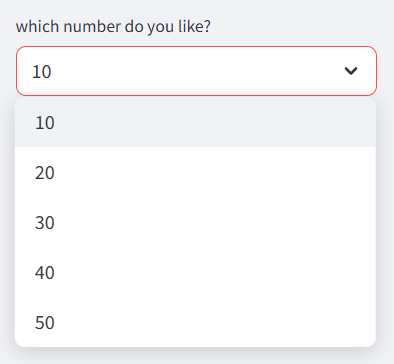
## ADD WIDGETS TO SIDEBAR

Kita juga bisa menggunakan **st.sidebar.[nama\_elemen]** untuk menambahkan elemen ke sidebar. Setiap elemen yang ditambahkan ke **st.sidebar** akan muncul di sebelah kiri, membuat tampilan halaman menjadi lebih rapi.

Berikut adalah kode nya :

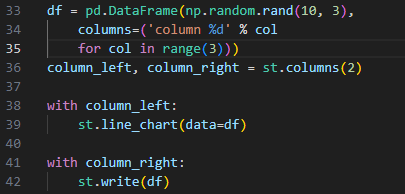


Berikut hasilnya :

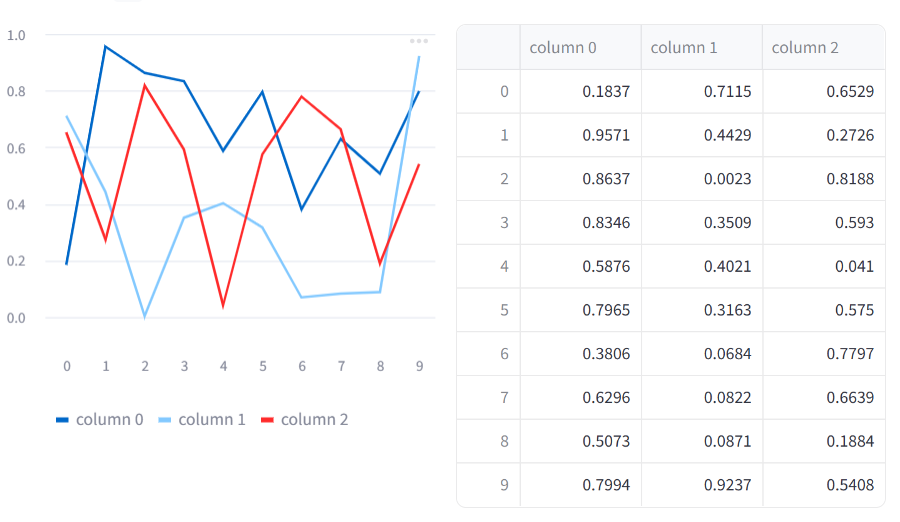


**Streamlit.columns()** memungkinkan kita untuk mengatur widget agar tampil berdampingan. **Streamlit.form** digunakan untuk membuat formulir yang mengelompokkan elemen-elemen bersama dengan tombol "Kirim". Sedangkan **Streamlit.expander** dapat digunakan untuk menyembunyikan konten besar agar lebih rapi.

Berikut adalah contohnya :



Berikut hasilnya :



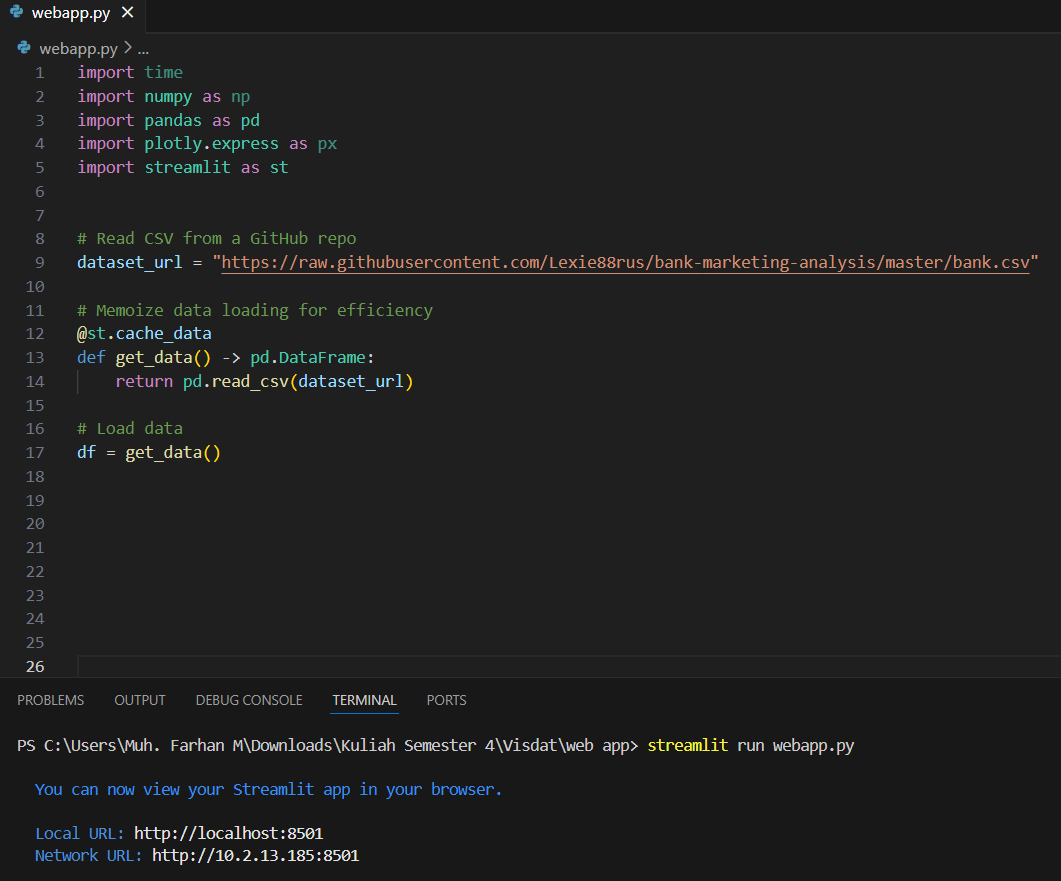
## CACHING

Caching adalah fitur yang disediakan oleh Streamlit untuk mempercepat kinerja aplikasi. Fitur ini membuat aplikasi berjalan lebih cepat, terutama saat memuat atau memproses data dalam jumlah besar. Dengan caching, aplikasi dapat menghindari pengulangan operasi berat dengan menyimpan hasil sebelumnya. Anda hanya perlu menambahkan dekorator **@st.cache** dalam kode untuk mengaktifkan caching.

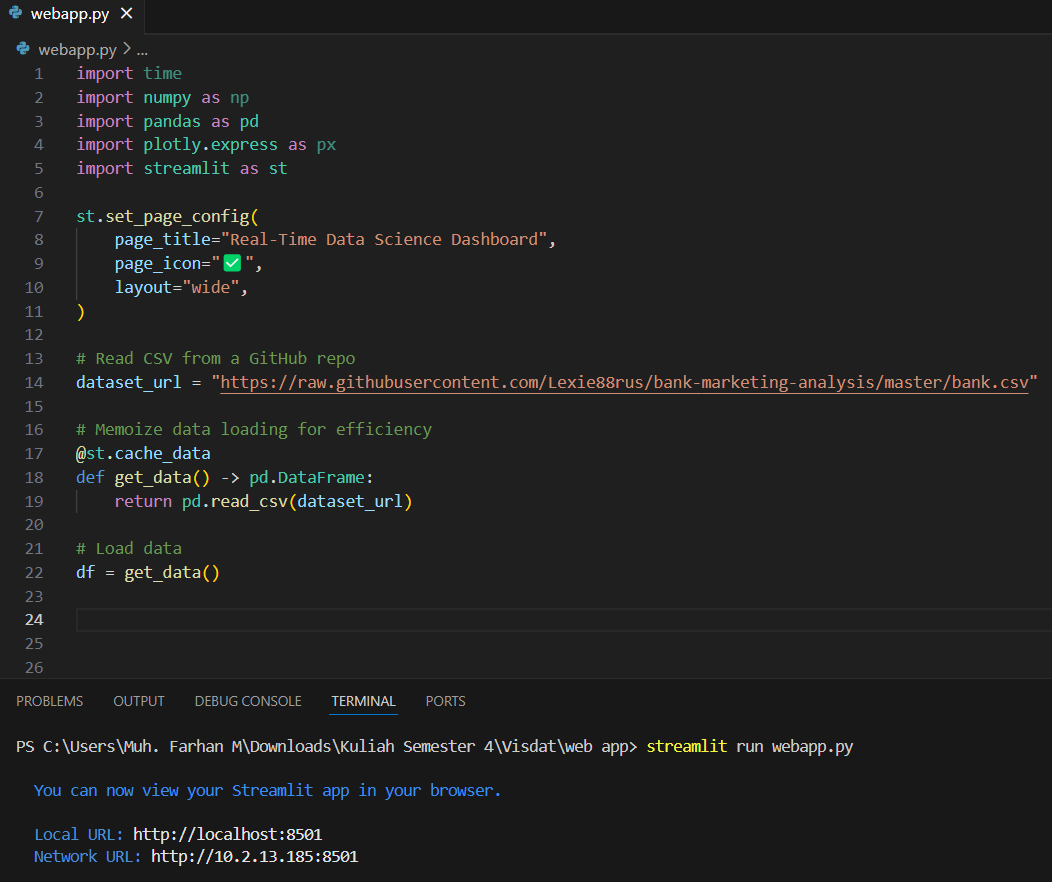
# Build A Real-Time Live Dashboard With Streamlit

Membuat sebuah dasbor (dashboard) yang dapat menampilkan data secara langsung dan terus diperbarui (real-time) menggunakan Streamlit. Berikut langkah-langkahnya :

1. Mengimpor Libraries Yang Diperlukan Dan Membaca Data Input



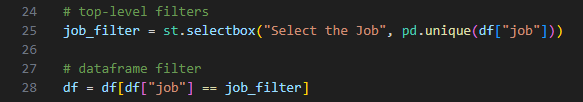
1. Pengaturan Dasbor Dasar



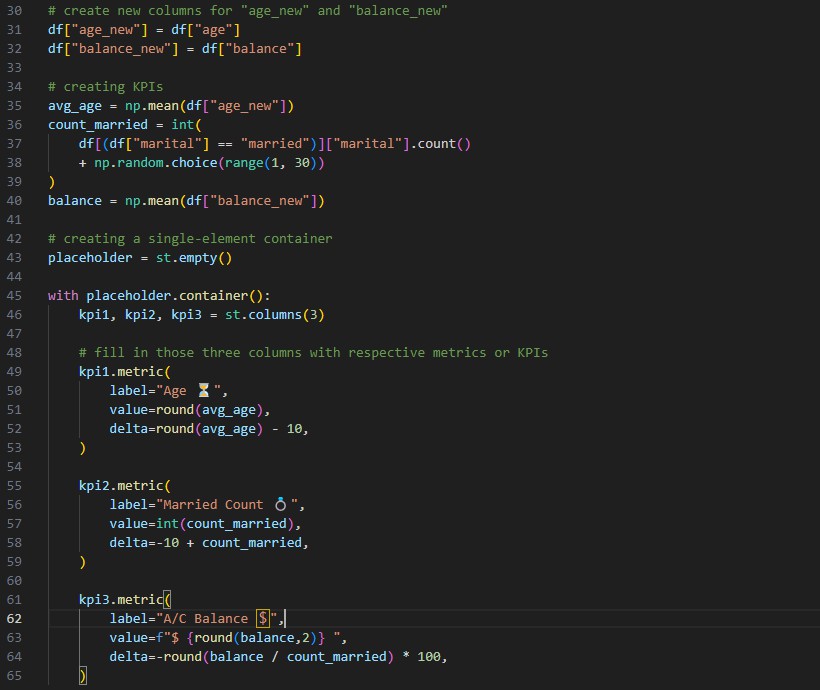
1. Merancang Antarmuka Pengguna (UI)
   1. Judul Halaman



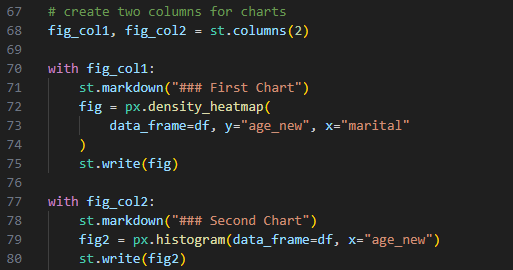
* 1. Filter Tingkat Atas



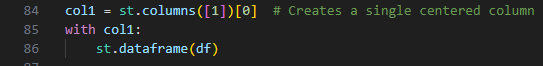
* 1. Kartu KPI/Ringkasan



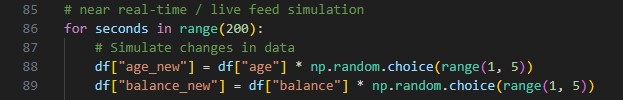
* 1. Grafik Interaktif



* 1. Tabel Data



1. Refresh Dasbor untuk Menerima Data Langsung atau Waktu Real



1. Komponen Pembaruan Otomatis



1. Hasilnya :

