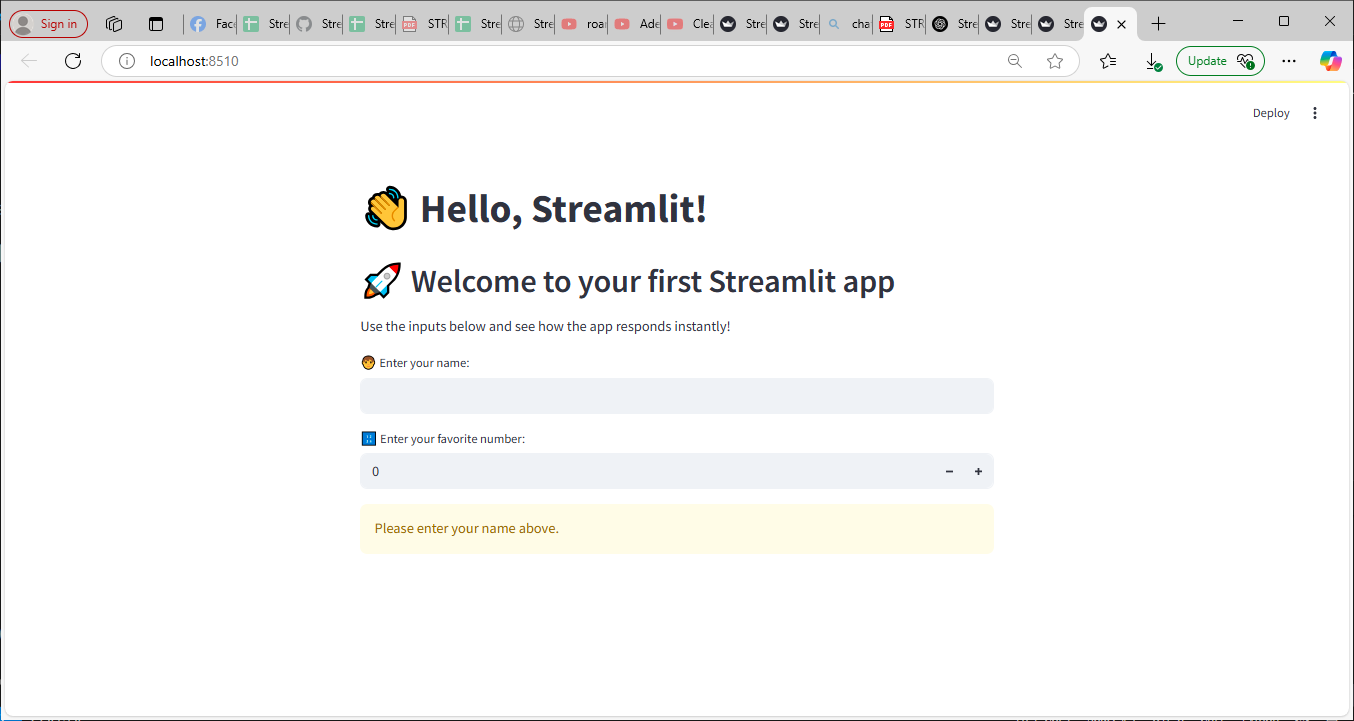
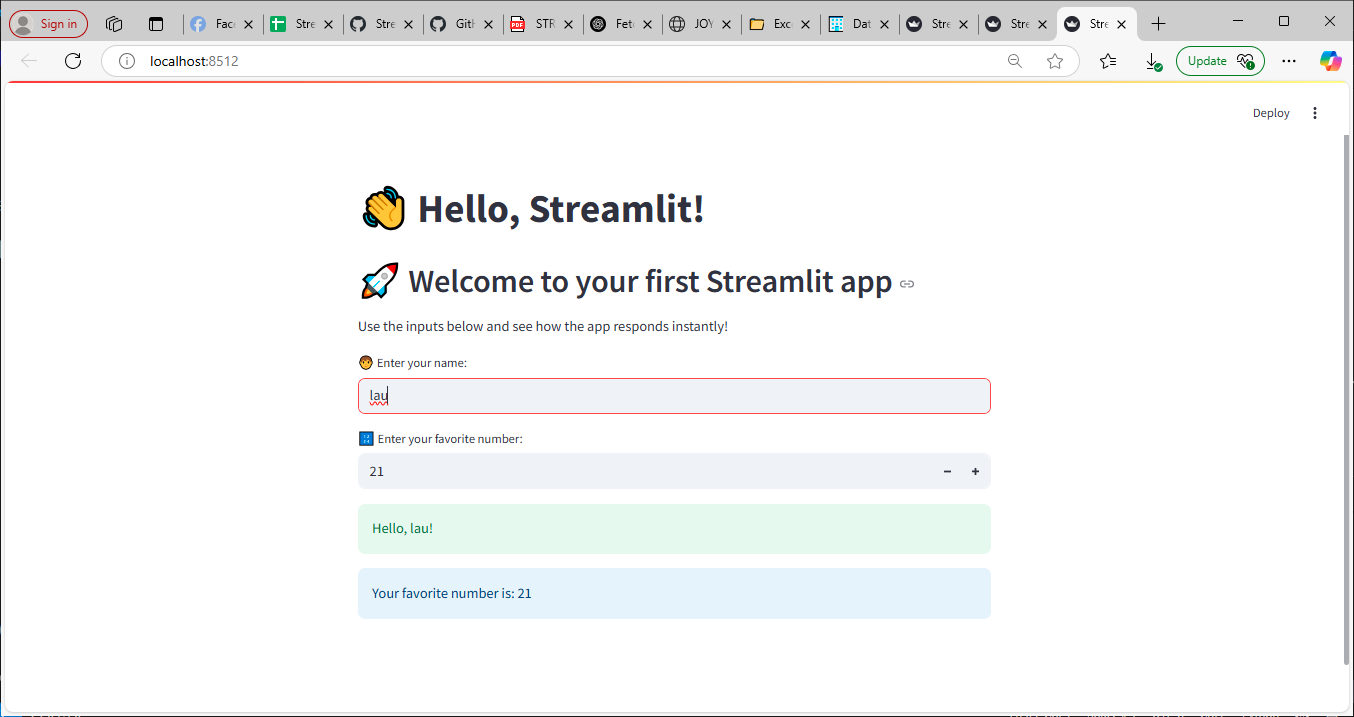
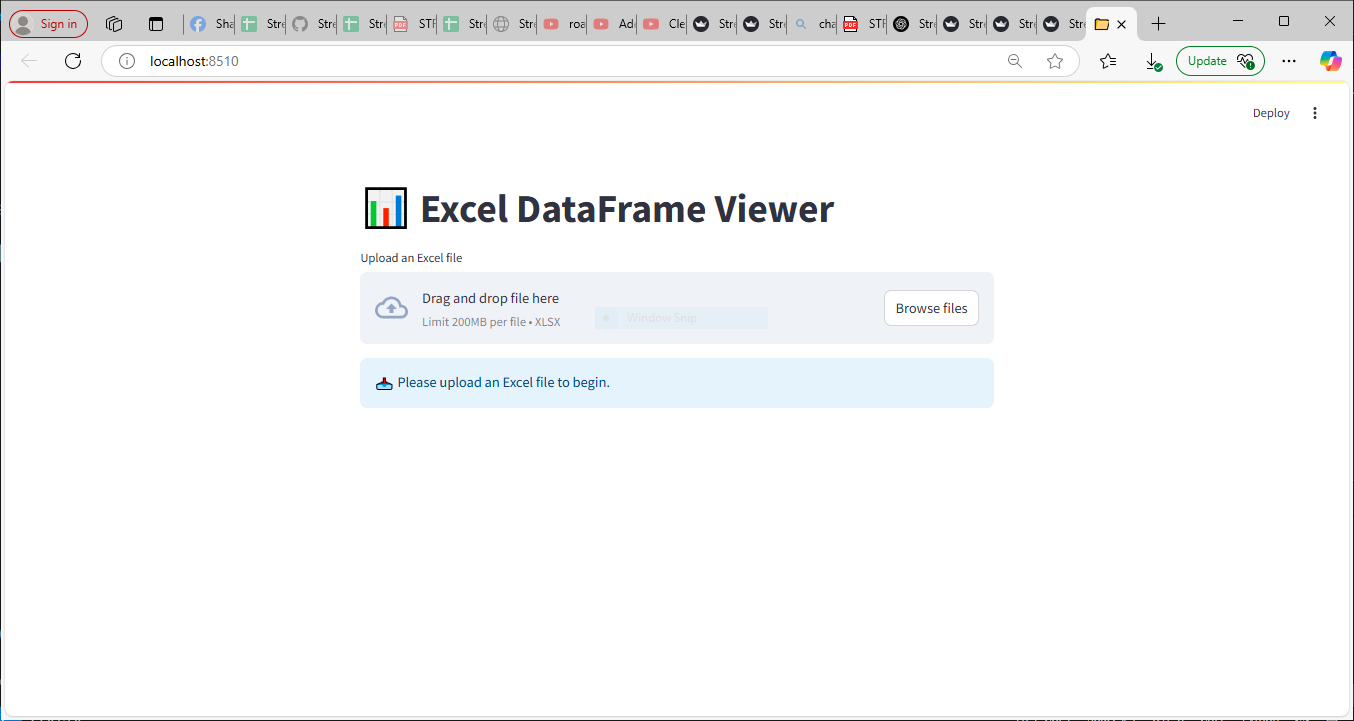
Act1

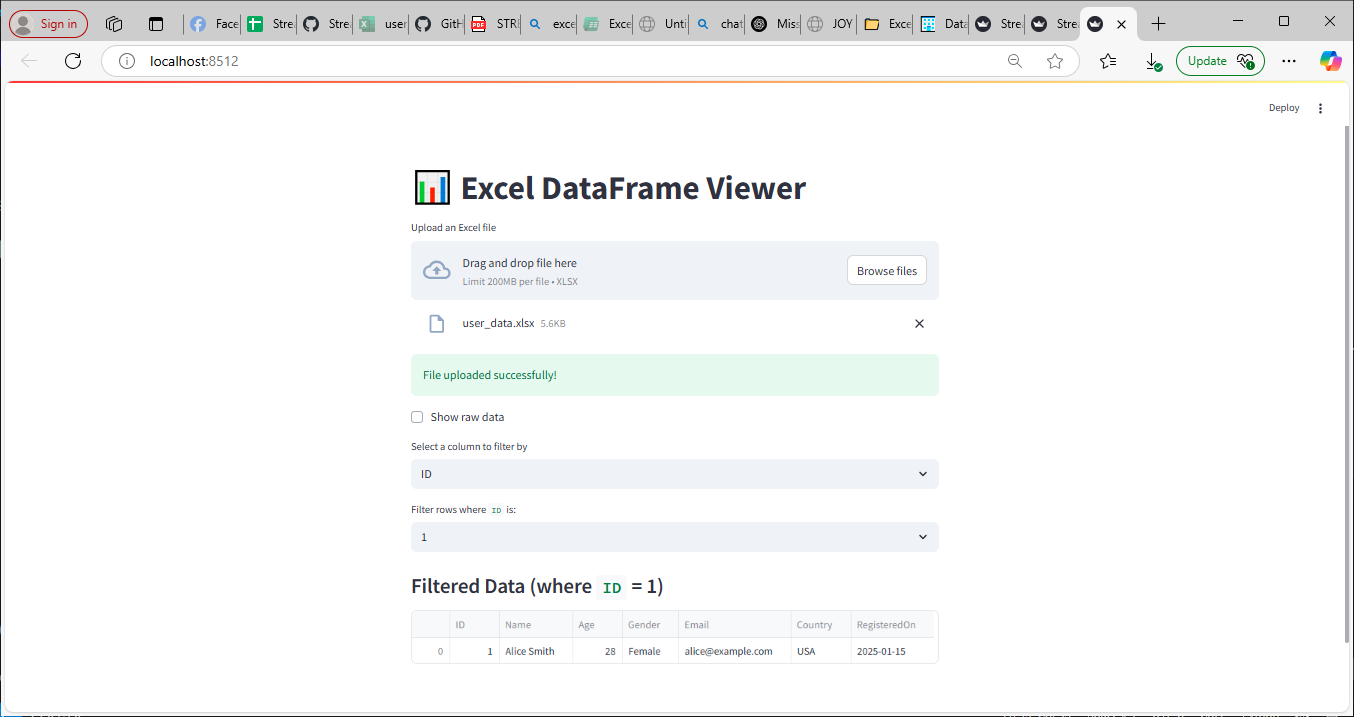




This Streamlit code creates a simple interactive web app that greets the user by name and shows their favorite number. It displays a message once the user types their name, otherwise it asks them to enter it.

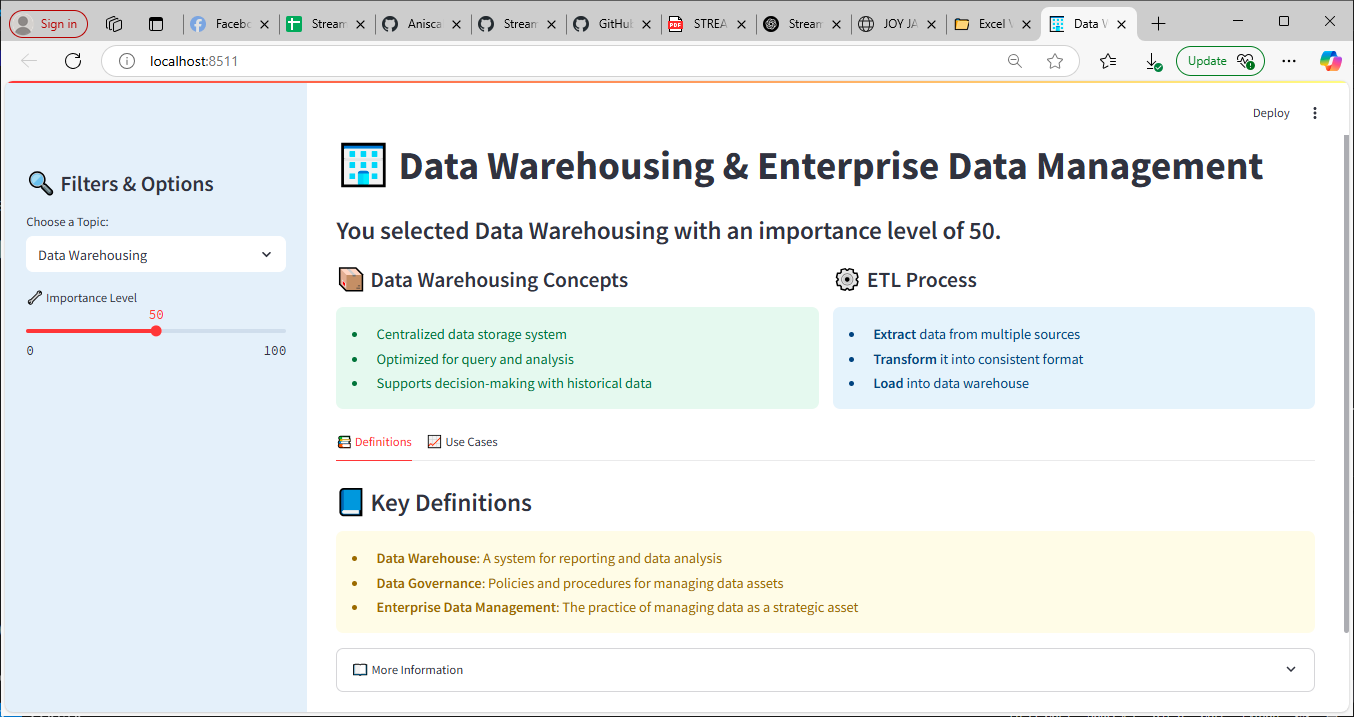
Act2



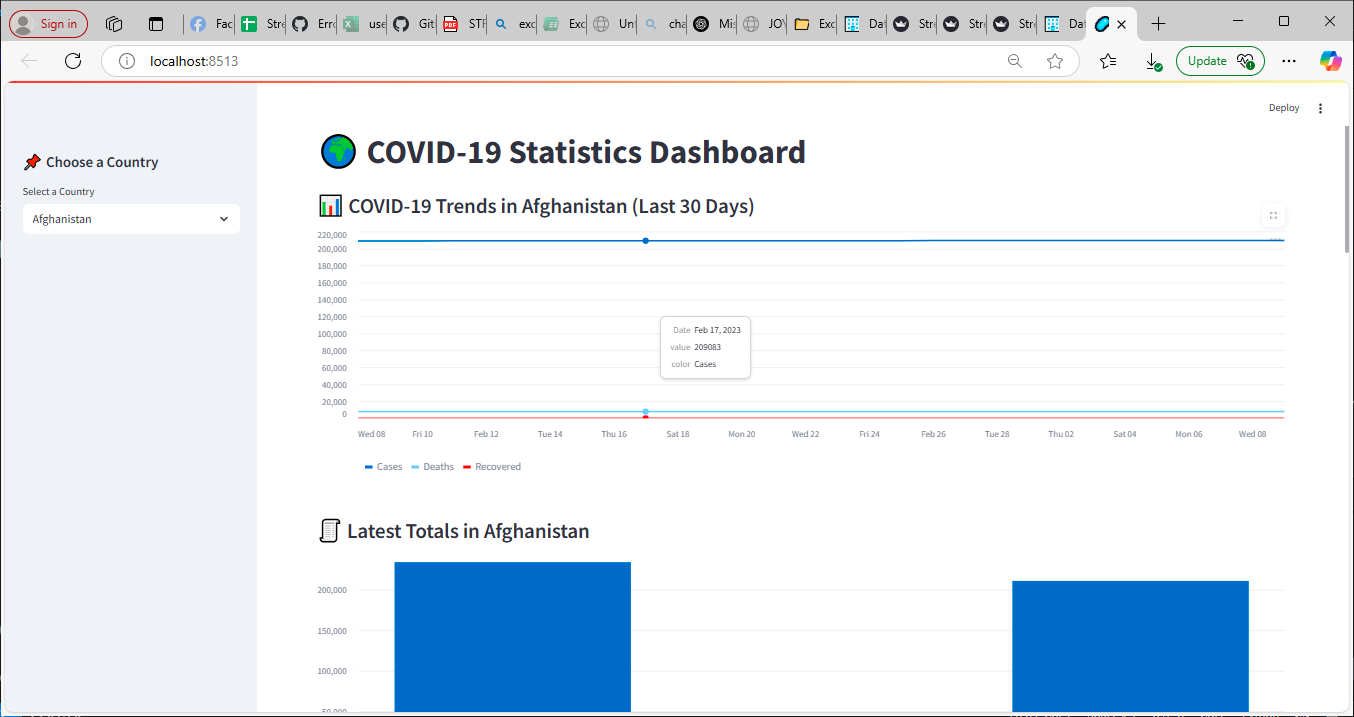
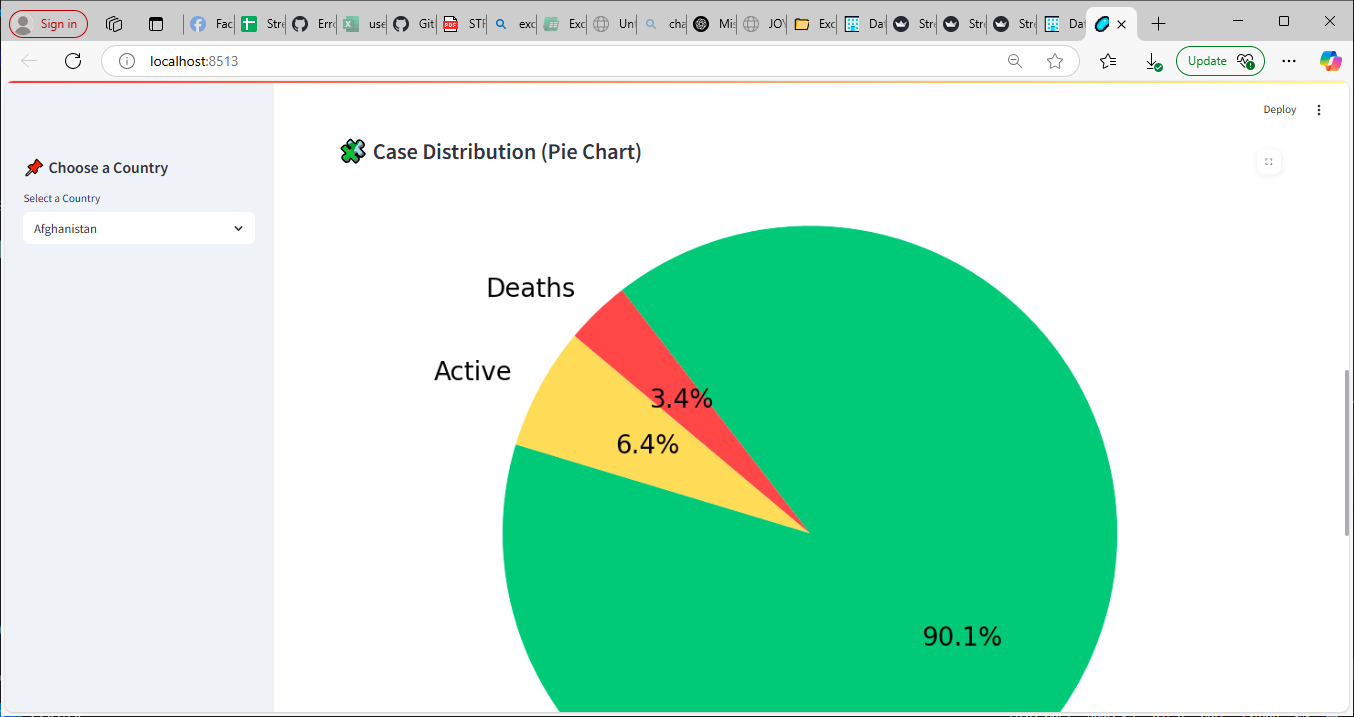
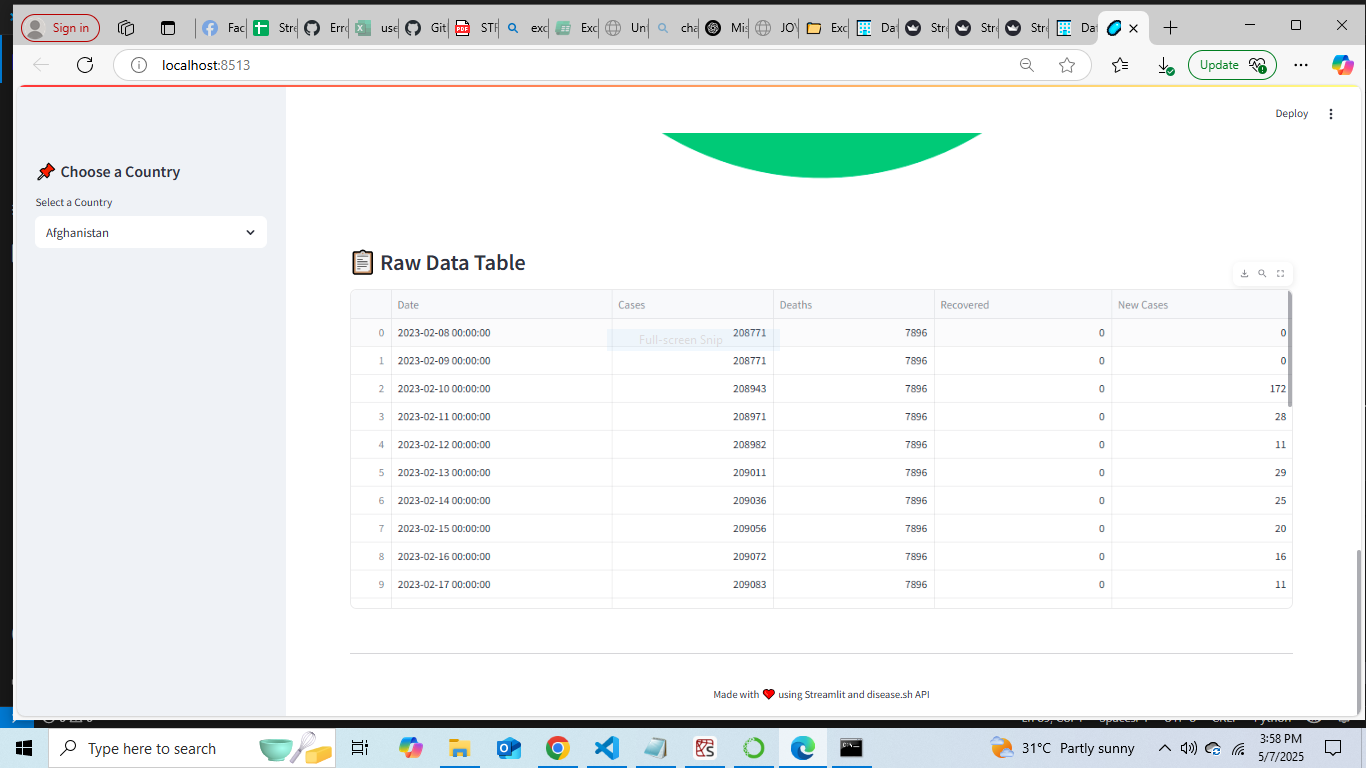


This Streamlit app lets users upload an Excel file, checks if it has at least 5 columns, and displays the data with options to view raw content and filter it by a selected column and value. If there's an error during upload or reading, it shows an error message to guide the user.

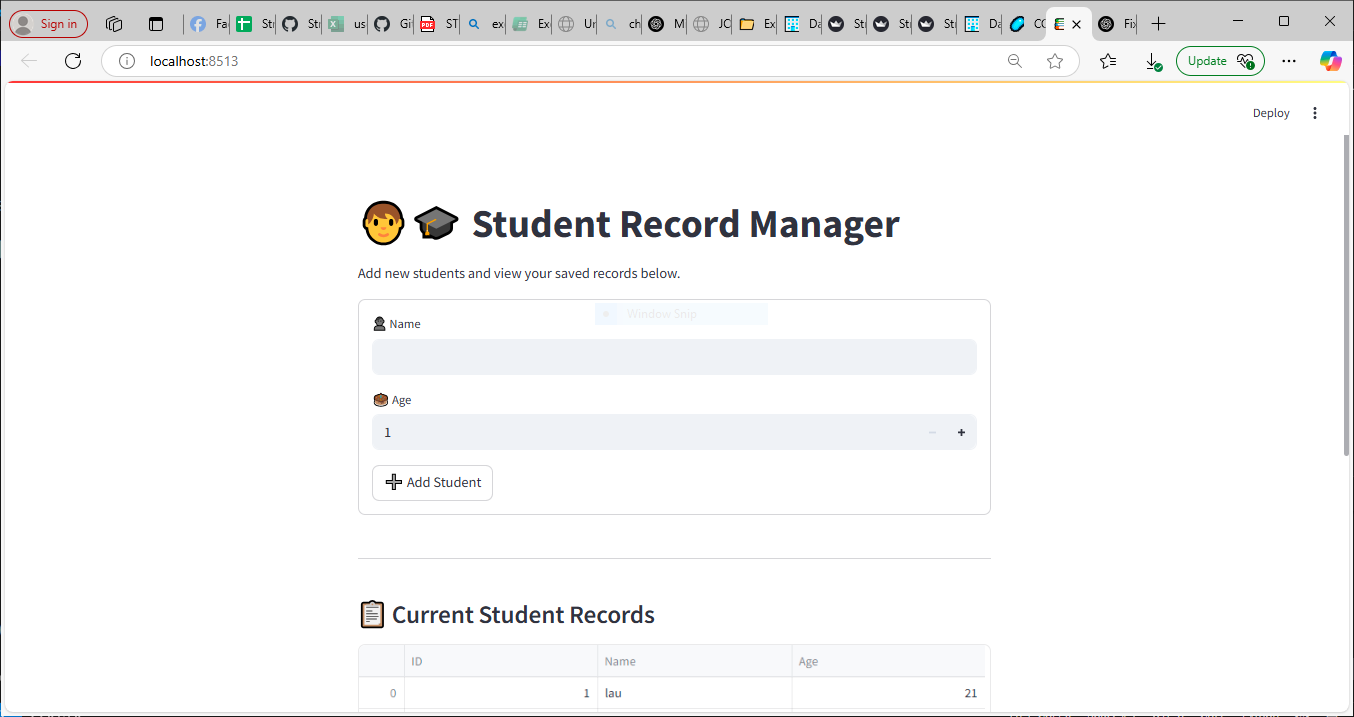
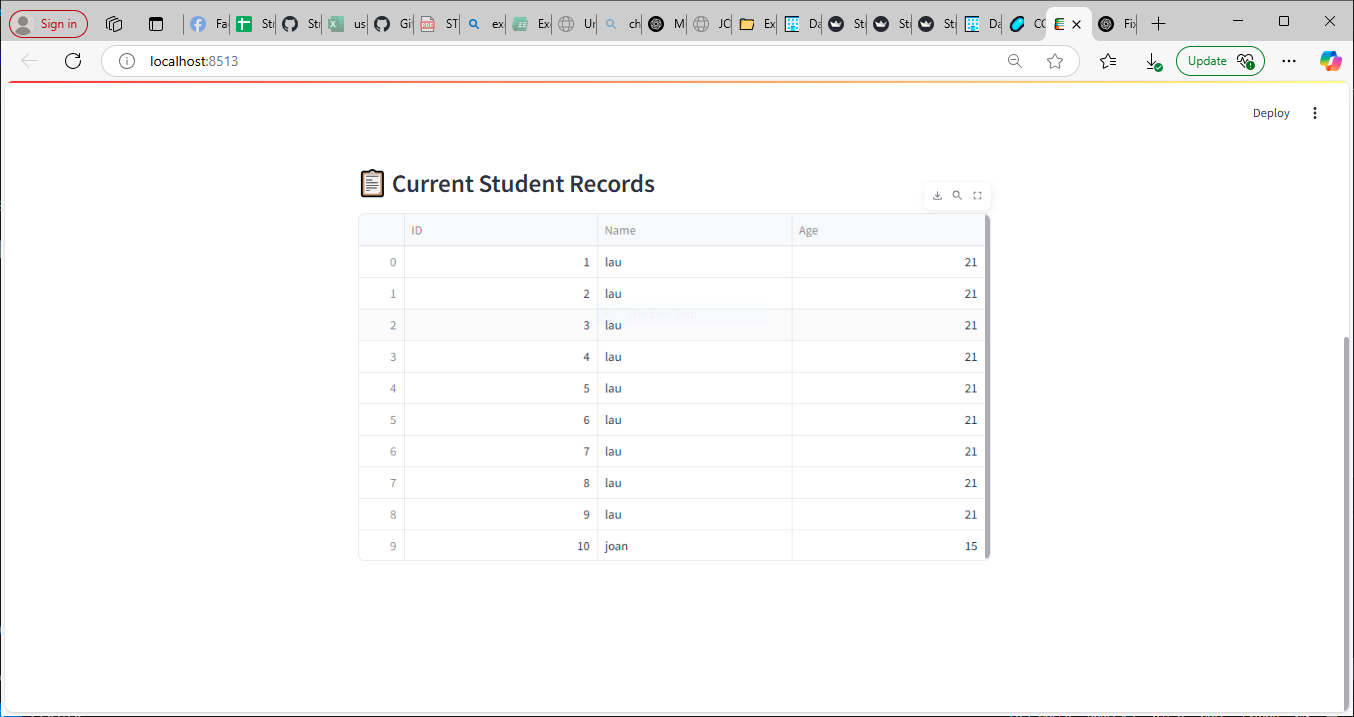
Act3



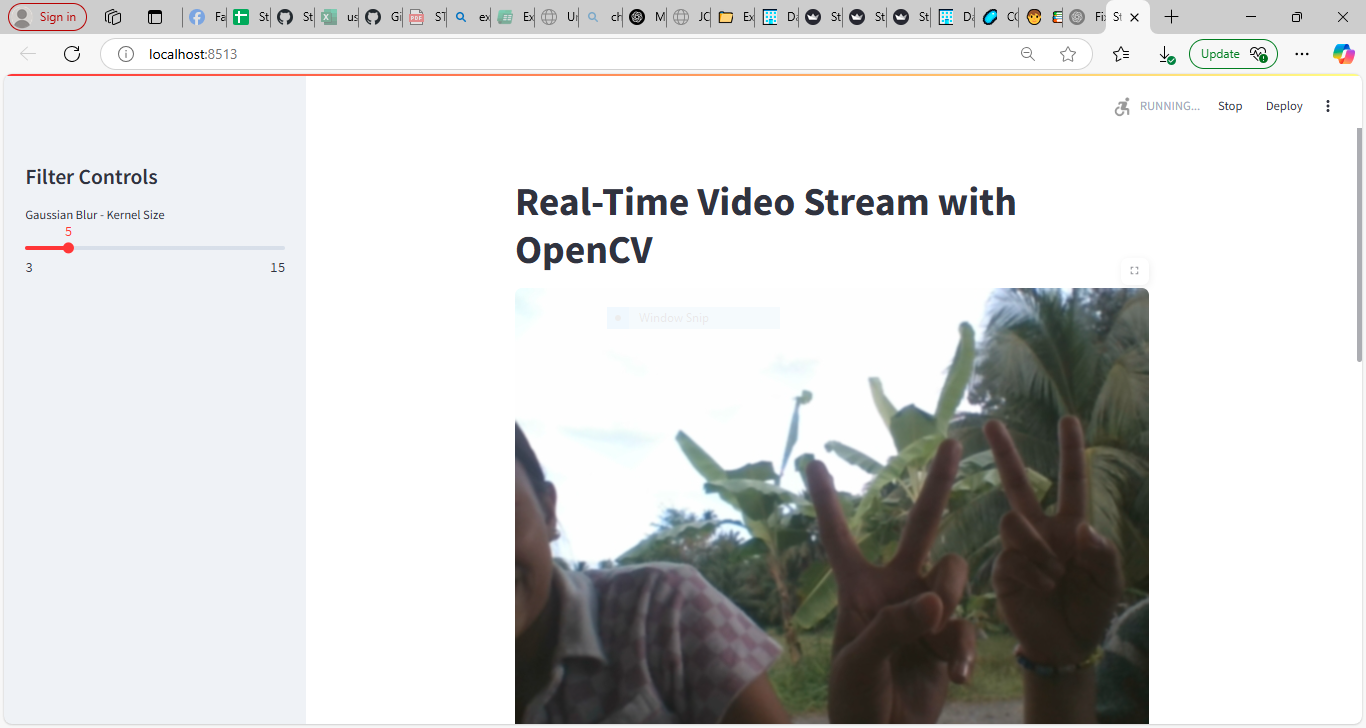
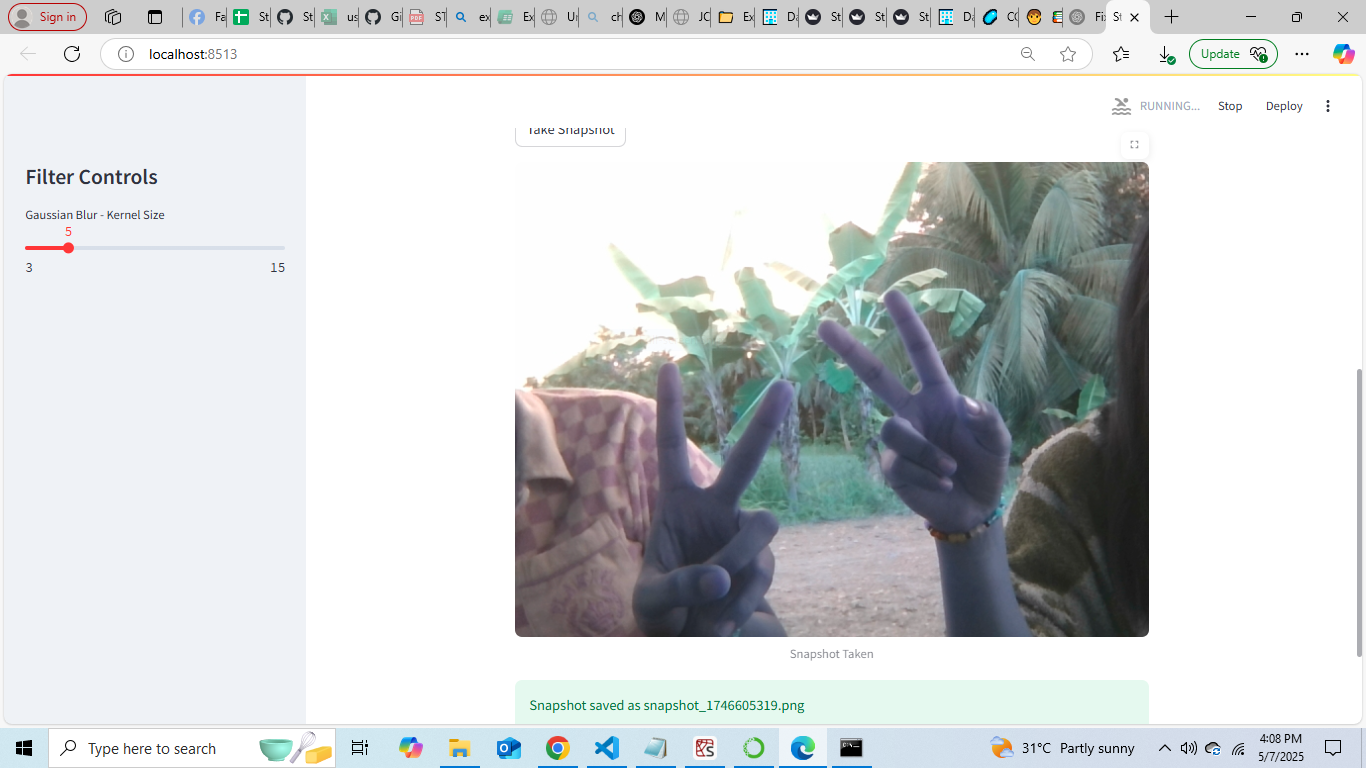
This Streamlit code creates an interactive web page for exploring concepts in data warehousing and enterprise data management, including styled layouts, sidebar filters, tabs, and informative content. It lets users choose a topic and importance level, then displays related definitions, use cases, and additional explanations in a visually organized format.

Act4

This Streamlit app displays COVID-19 statistics by fetching live data from the disease.sh API and visualizing it through charts and tables for any selected country. It includes trends over the past 30 days, current totals, new cases, and a pie chart of case distribution using Matplotlib.

Act5

This Streamlit app lets users manage a simple student database by adding names and ages into a local SQLite database and displaying them in a table. It includes a form for input, validation for empty names, and dynamic feedback after each submission.

Act6

This Streamlit app uses OpenCV to display a real-time webcam video stream with an adjustable Gaussian blur filter, controlled via a sidebar slider. It also allows users to take and save snapshots from the live video feed.