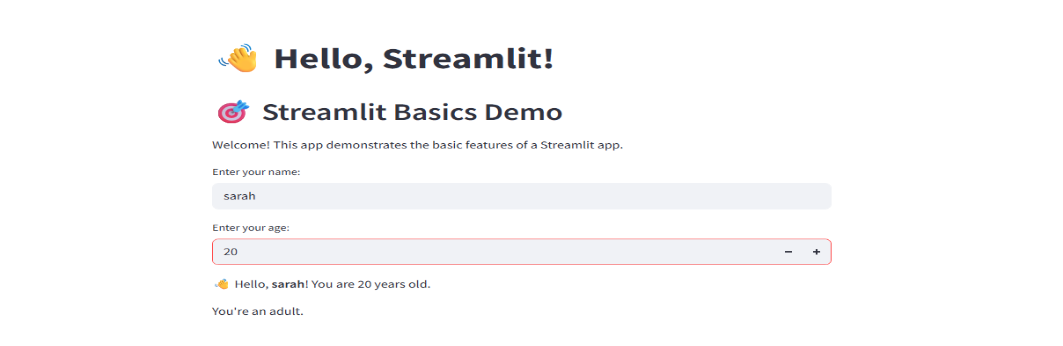
FUNDAMENTALS OF ENTERPRISE DATA MANAGEMENT STREAMLIT WORKBOOK

I. Hello, Streamlit!

Objective: Understand basic components of a Streamlit app.

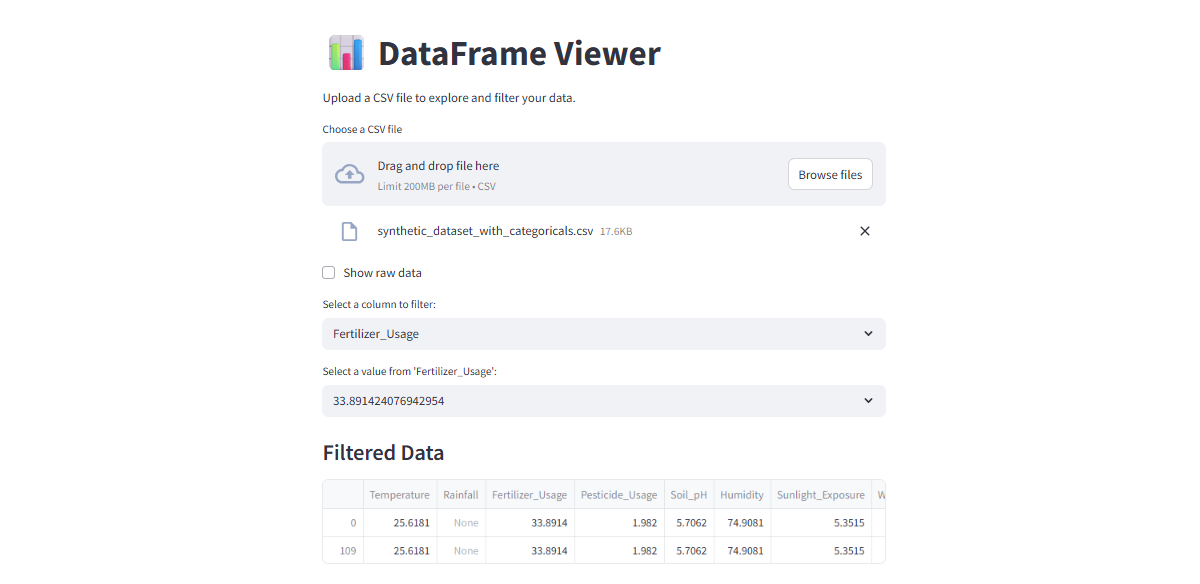


Explanation:

This Streamlit web app demonstrates the basic structure and components of a Streamlit application. It uses st.title, st.header, and st.write to display text, and allows user interaction through st.text\_input and st.number\_input. Based on the user's name and age input, it dynamically displays a personalized message and a category based on age.

II. DataFrame Viewer

Objective: Load and display data.

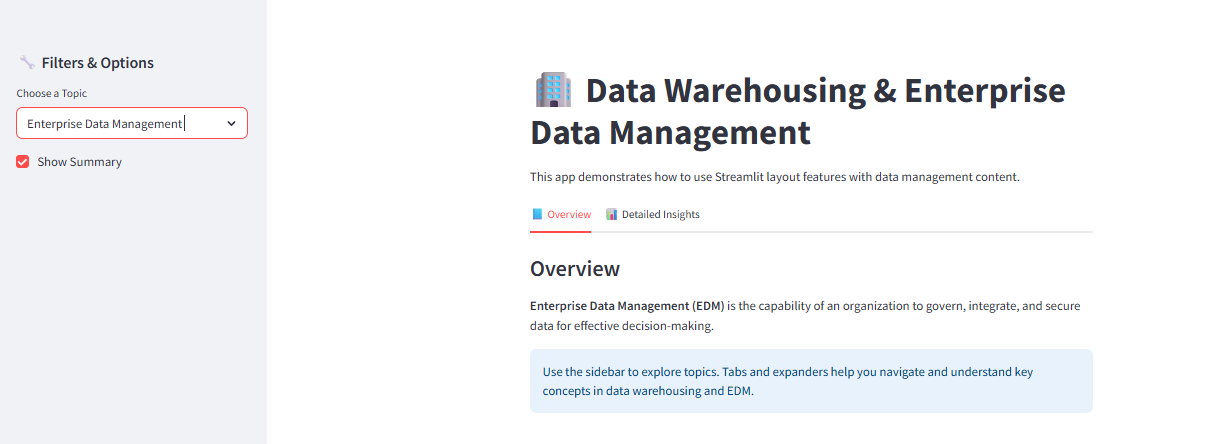


Explanation:

This app allows users to upload a CSV file, then displays the raw data using a checkbox toggle. It uses st.selectbox to let users choose a column and filter the displayed data based on selected values in that column.

III. Sidebar and Layout

Objective: Organize content with Streamlit layout components.



A close-up of a data storage

AI-generated content may be incorrect.

Explanation:

This app uses st.sidebar for topic selection and options, st.tabs and st.columns to organize content into readable sections, and st.expander to toggle in-depth explanations. It presents educational content on data warehousing, ETL, and enterprise data management in a structured and interactive format.

IV. Fetch and Display API Data

Objective: Connect to an external API and display results. A graph of orange bars

AI-generated content may be incorrect.A graph of a heatmap

AI-generated content may be incorrect.

A graph with blue triangles

AI-generated content may be incorrect.A pie chart with numbers and a number of different colored circles

AI-generated content may be incorrect.A graph with a line

AI-generated content may be incorrect.

Explanation:

This app fetches real-time COVID-19 statistics from a public API, parses the JSON into a DataFrame, and uses five different chart types—bar, pie, line, area, and heatmap—to visualize data trends. It demonstrates how to integrate external data sources and create interactive visual analytics using Streamlit.

V. Data Pipeline with Database

Objective: Integrate a Streamlit app with a relational database (MySQL)

A screenshot of a computer

AI-generated content may be incorrect.

Explanation:

This Streamlit app connects to a MySQL database using SQLAlchemy, allowing users to fetch and filter data, display results, and insert new records via a form. It also includes a basic login system for user authentication, simulating a secure access feature for database operations.

VI. Real-Time Video Stream with CV

Objective: Use OpenCV to capture video and process it in real-time. 

Explanation: This app uses OpenCV (cv2.VideoCapture) to stream live video from the webcam and displays each frame using st.image() in real time. It demonstrates real-time image processing integration between OpenCV and Streamlit.