**Secure Check: A Python-SQL Digital Ledger for Police Post Logs**

**Purpose Tools/Techniques**

1   Data storage & schema: PostgreSQL, SQLAlchemy, Render

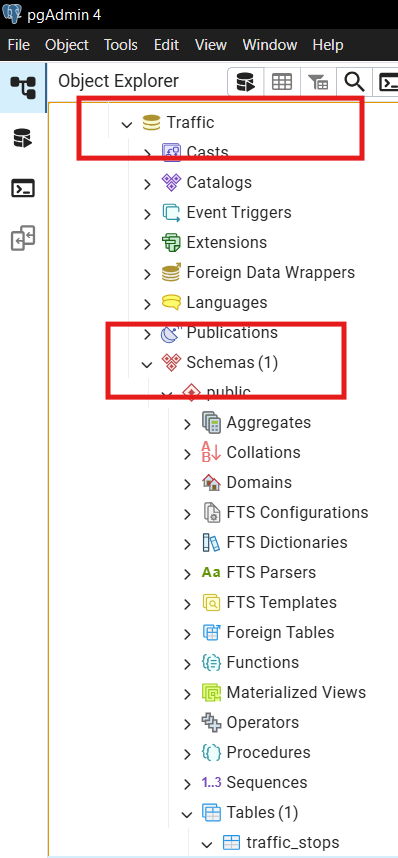
2   Data cleaning: Python, pandas

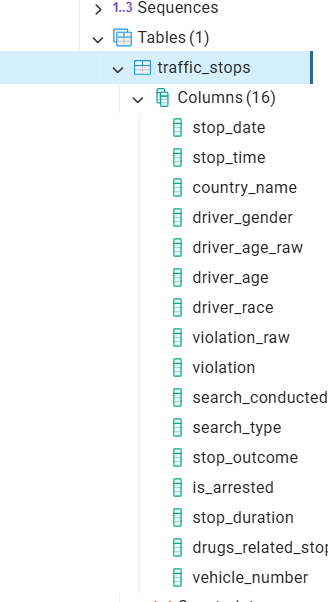
3   Database design: SQL

4   Dashboard & analytics: Streamlit, Plotly, SQLAlchemy

**Define SQL Schema for Police Stop Records**

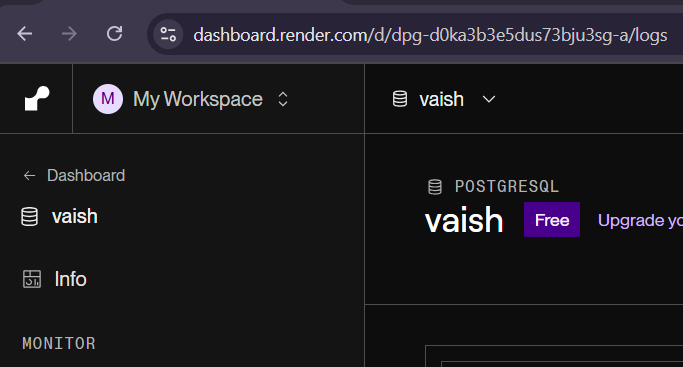
`stop\_id`, `stop\_date`, `stop\_time`, `county\_name`, `driver\_gender`, `driver\_age`, `driver\_race`, `search\_conducted`, `drugs\_related\_stop`, `stop\_duration`, `vehicle\_number`, `violation`, `stop\_outcome`,

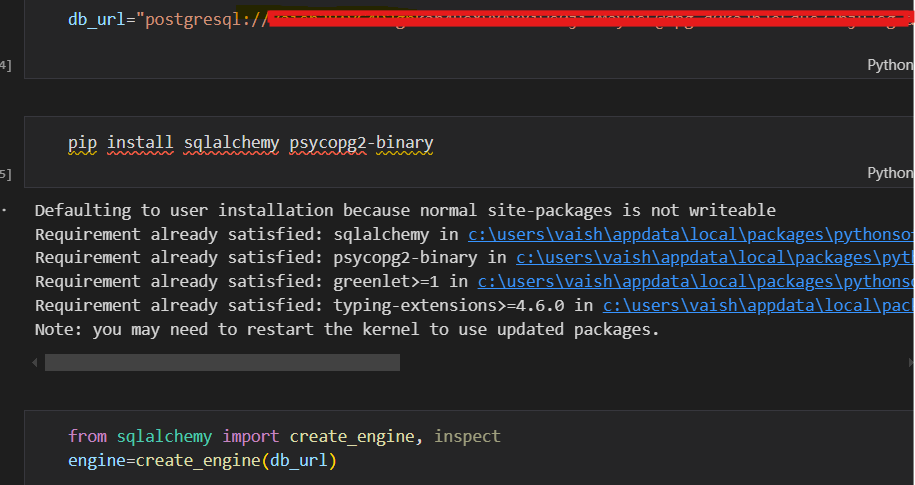




**SQL table creation:**

1. Use Python (pandas, sqlalchemy) to Insert and Query Data
2. Use `pandas` to process data in Python.
3. Use `sqlalchemy` to connect Python to your SQL database.
4. Connecting Render cloud platform with SQL for accessing DB for anywhere.

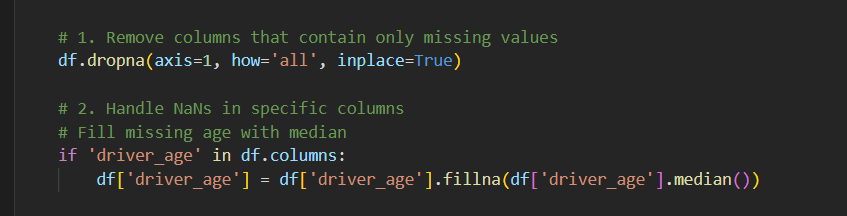




**Python for Data Processing**

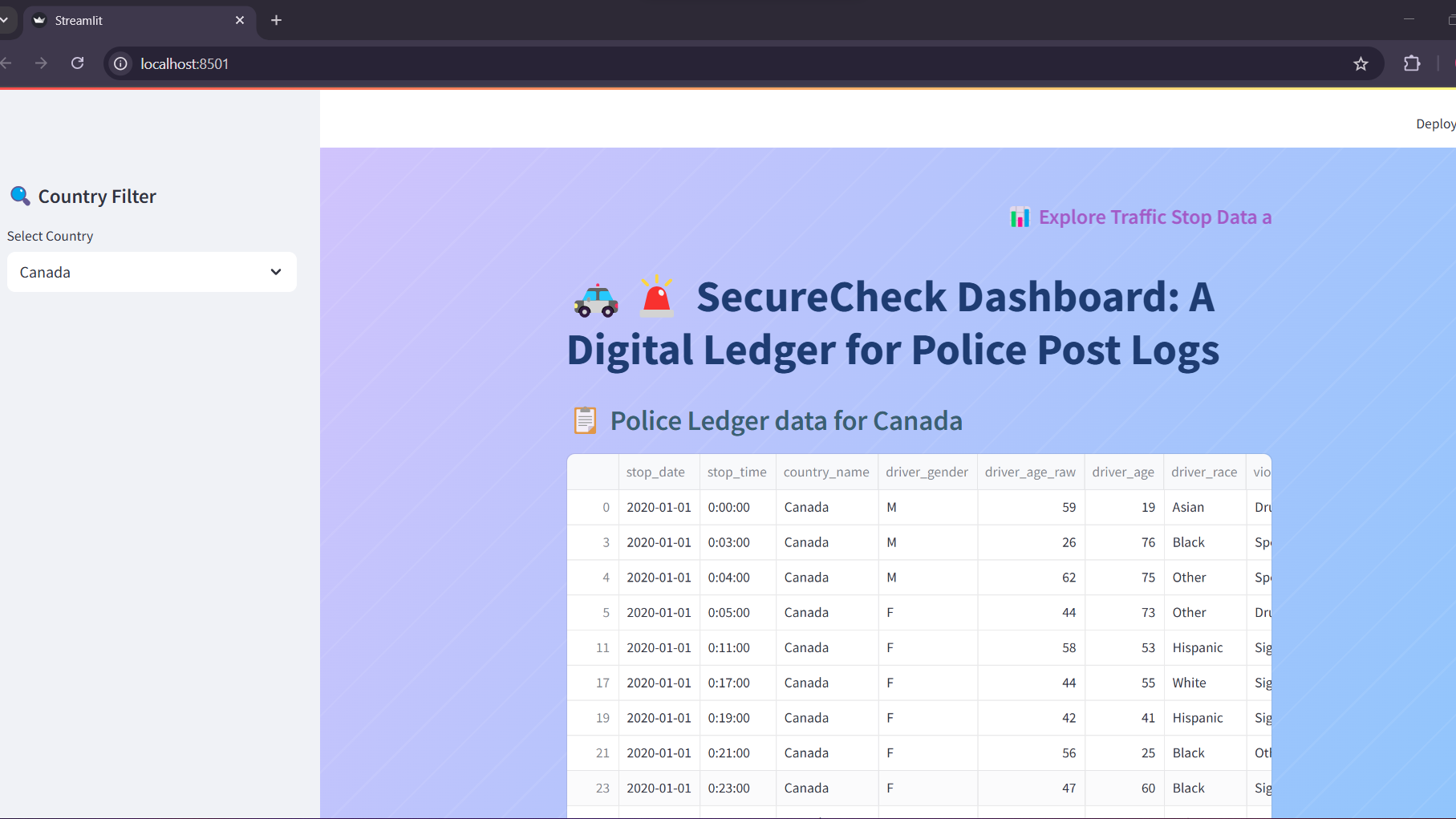
1.Remove Columns That Only Contain Missing Values

2.Handle the NaN Values

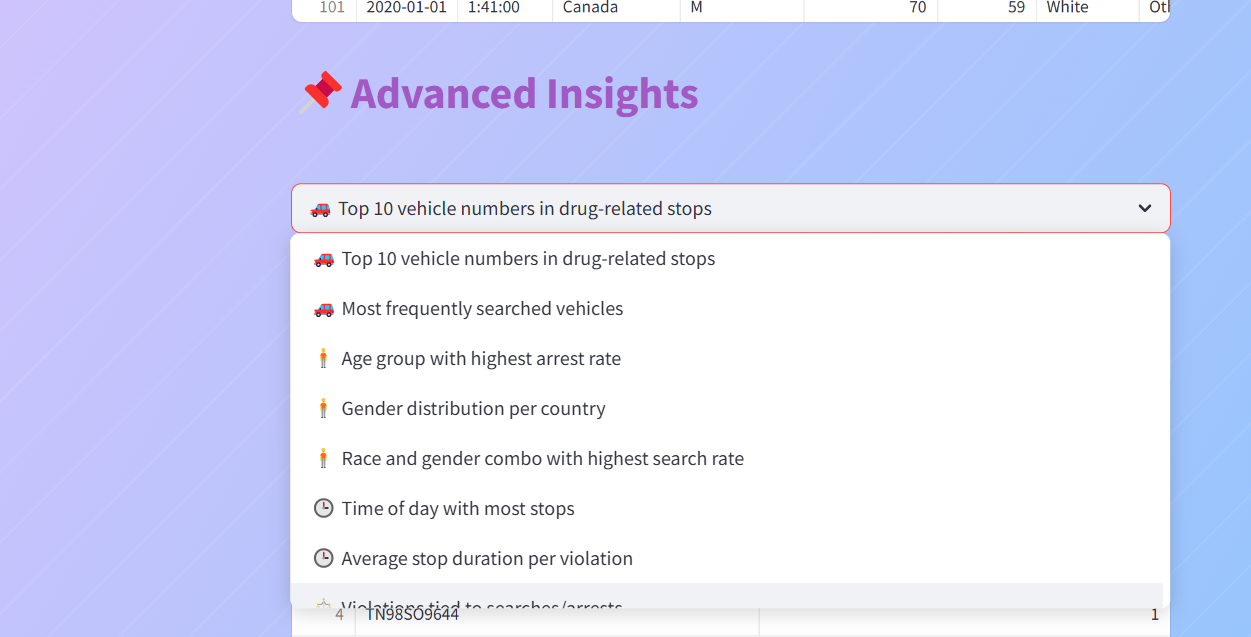


**Streamlit Dashboard**

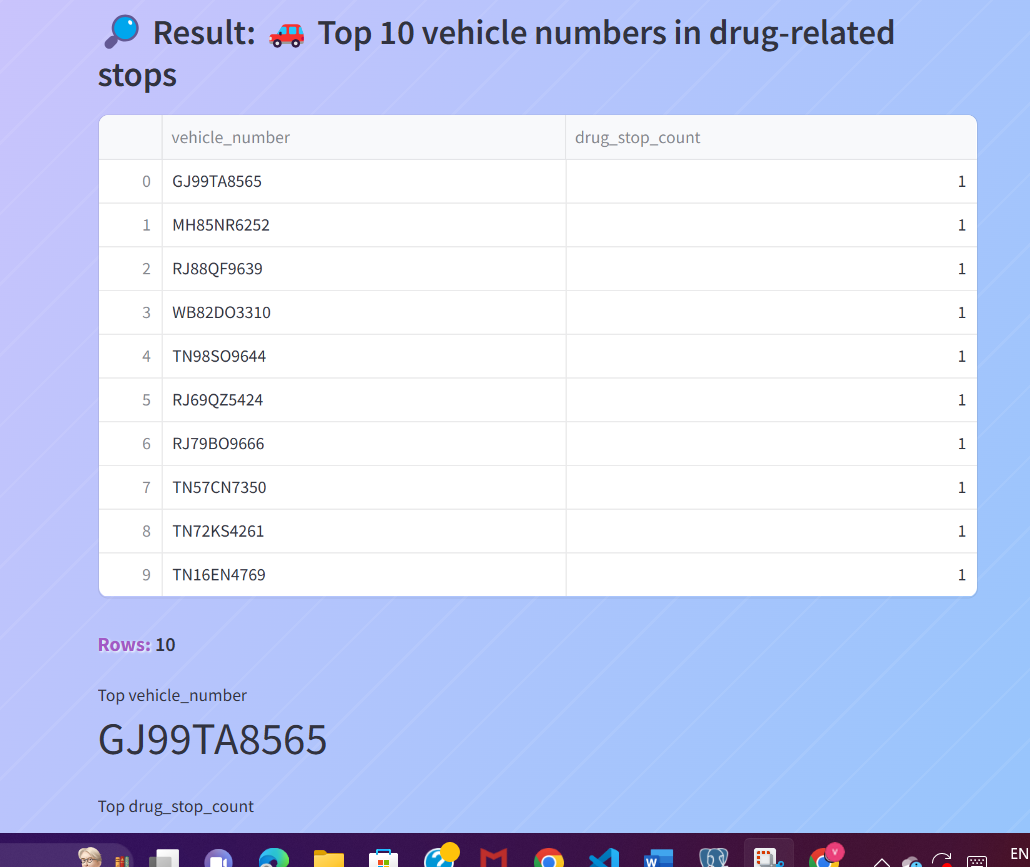
**a. Display Vehicle Logs, Violations, and Officer Reports**



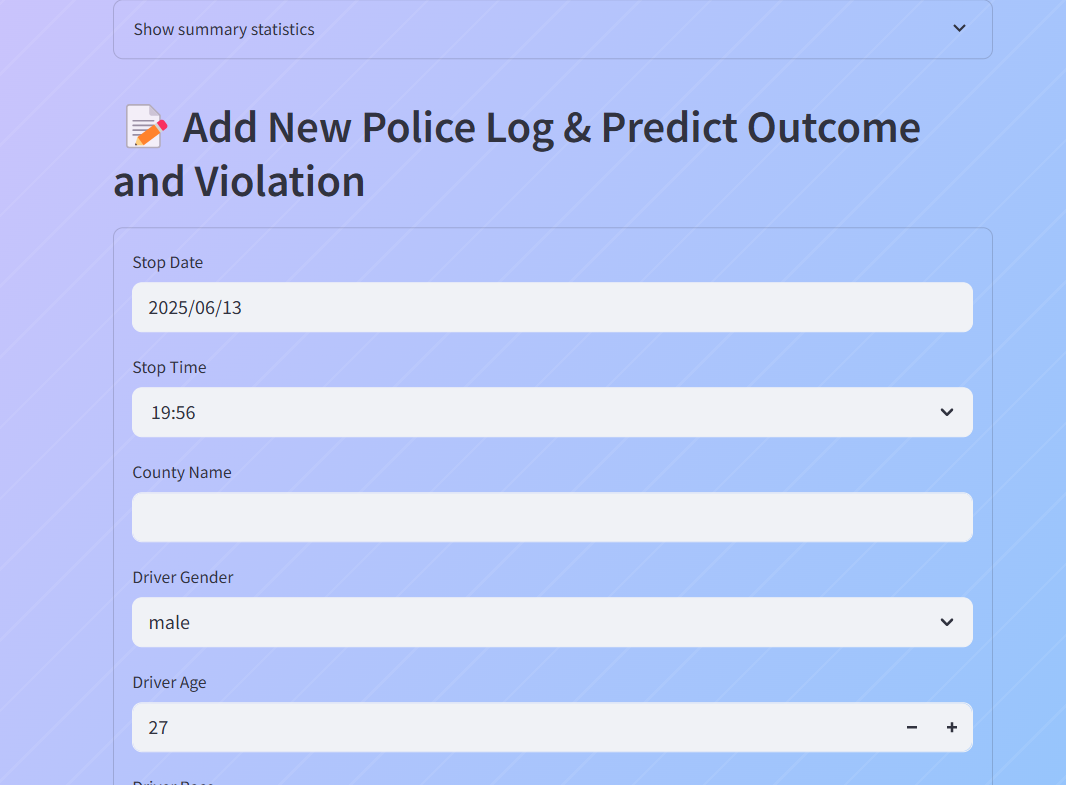
**b. Implement SQL-Based Search Filters for Quick Lookups**



**c. Generate Analytics and Trends (e.g., High-Risk Vehicles)**



**Prediction Violation Output:**



**Prediction outcome snapshot:**

