

Province Data Comparison App

This Streamlit app allows users to explore and compare population, administrative units, and housing data for different provinces in Pakistan, specifically for the year 2017. The app supports two modes: **Display Mode** and **Comparison Mode**.

- **Display Mode** shows data and graphs for a single province.
- **Comparison Mode** allows users to compare the data between two selected provinces.

Features

- **Display Mode:** View detailed population, housing, and other data for one province at a time.
- **Comparison Mode:** Compare population or housing data between two provinces with side-by-side bar charts.
- **Interactive Graphs:** Use Plotly for interactive bar graphs to visualize the data comparison.

Requirements

To run the app, ensure you have the following dependencies installed:

- Python 3.7+
- Streamlit
- Pandas
- SQLite3 (for database connection)
- Plotly

You can install the required dependencies using `pip`:

bash

Copy code

```
pip install streamlit pandas plotly sqlite3
```

Setup

1. **Database:** The app fetches data from an SQLite database (`population_data.db`). Ensure this file is available in the same directory as the app or modify the connection string to the correct path.

2. **Database Structure:** The database should have the following tables:
 - **Population:** Contains columns `province`, `indicator`, `value`, and `percentage`.
 - **AdministrativeUnits:** Contains columns `province`, `division`, `population`, and `percentage`.
 - **Housing:** Contains columns `province`, `category`, `units`, and `percentage`.

Example schema:

sql

Copy code

```
CREATE TABLE Population (  
    province TEXT,  
    indicator TEXT,  
    value INTEGER,  
    percentage REAL  
);  
  
CREATE TABLE AdministrativeUnits (  
    province TEXT,  
    division TEXT,  
    population INTEGER,  
    percentage REAL  
);  
  
CREATE TABLE Housing (  
    province TEXT,  
    category TEXT,  
    units INTEGER,  
    percentage REAL  
);
```

3.

Running the App

1. Save the script as `app.py` (or any preferred name).
2. Open your terminal/command prompt and navigate to the directory where `app.py` is located.
3. Run the following command to start the app:

bash

Copy code

```
streamlit run app.py
```

This will open the app in your default web browser.

Modes

1. Display Mode

- **Purpose:** View detailed data and graphs for a single selected province.
- **Available Data:** Population, Housing
- **Display:** Interactive graphs for each dataset (Population and Housing) for the selected province.

2. Comparison Mode

- **Purpose:** Compare data between two selected provinces.
- **Available Data:** Population and Housing
- **Display:** Side-by-side bar charts comparing the data for the two selected provinces.

App Layout

- **Sidebar:**
 - Select the mode: Display Mode or Comparison Mode.
 - Select a province (for Display Mode) or two provinces (for Comparison Mode).
 - Select the type of data to compare in Comparison Mode (Population or Housing).
- **Main Area:**
 - Displays data in tables and interactive graphs based on the selected mode.

Example Usage

1. **In Display Mode:**
 - Select a province from the sidebar.
 - View the population and housing data for that province as tables and bar graphs.
2. **In Comparison Mode:**
 - Select two provinces from the sidebar.
 - Choose either Population or Housing as the comparison type.
 - View a side-by-side comparison of the two provinces' data.

Contributing

Feel free to fork this repository and contribute by submitting pull requests. Contributions can include:

- Adding new data columns
- Improving data visualization
- Enhancing the user interface

License

This project is open-source and available under the MIT License.

Troubleshooting

1. **Error: Missing Columns:**
 - Ensure that the database is correctly structured and all required columns (e.g., `province`, `indicator`, `value`, `units`, etc.) exist.
2. **Database Connection Issues:**
 - Make sure the `population_data.db` file is located in the same directory as the app or update the path to the database in the code.