WGU.edu

D211 Advanced Data Acquisition

Task 1 Video Submission

Mike Mattinson

April 22, 2022

Review Research Question

- Show Customer Density (Loyal Customers) by Region in the United States per 1,000 total population
  - Break down regions by states
  - Calculate density based on total state or region population
  - Show density as “0.043/1000”, meaning 0.043 customers per 1,000 total population.
  - Allow dashboard user to filter map and details by Region

Demonstration/Video

1. Describe the technical environment used to create the dashboards.
2. Demonstrate the functionality of the dashboards.
3. Explain the SQL scripts used to support the creation of the dashboards.
4. Explain how the data streams were prepared to support the analysis.
5. Describe how data were aligned with other data points.
6. Demonstrate how the databases were created.
7. Explain how referential integrity was enforced in the database.

Describe Technical Environment

Primary Components

- PostgreSQL
- pgAdmin 4
- Tableau 2021.4 (student)
- Notepad ++

Secondary Components

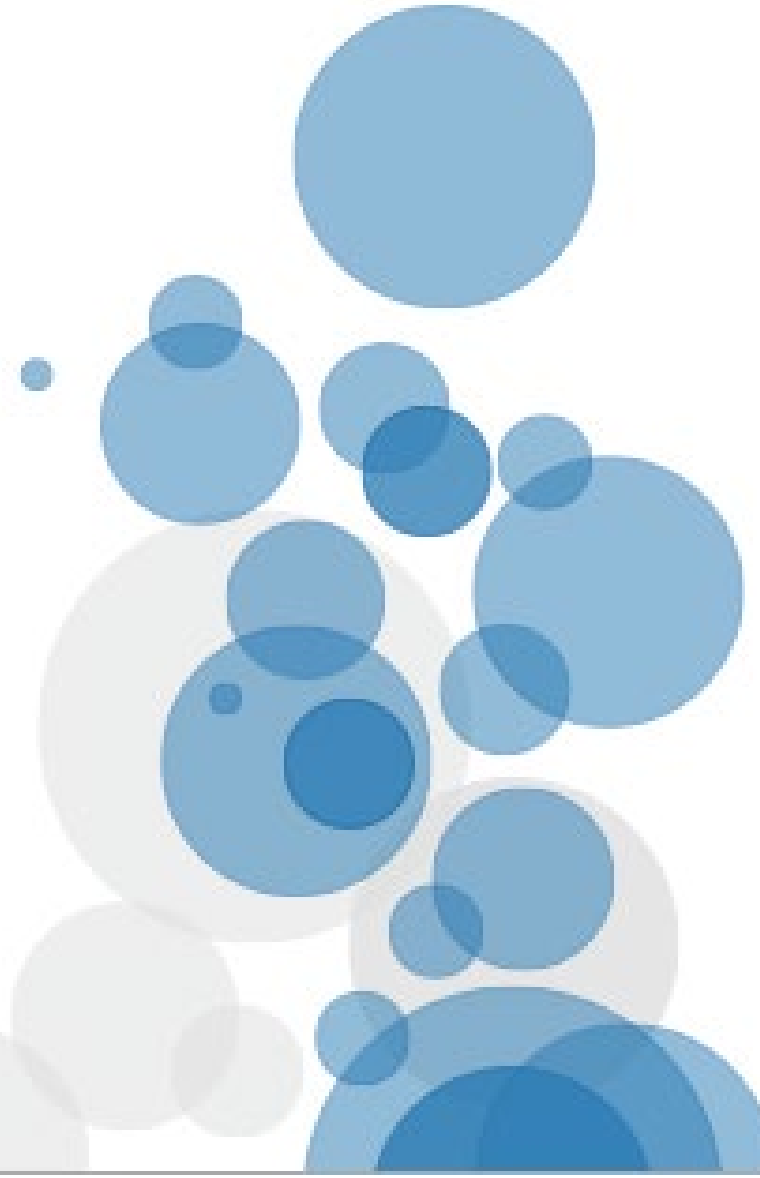
- VS Code
- Python
- Jupyter Notebook

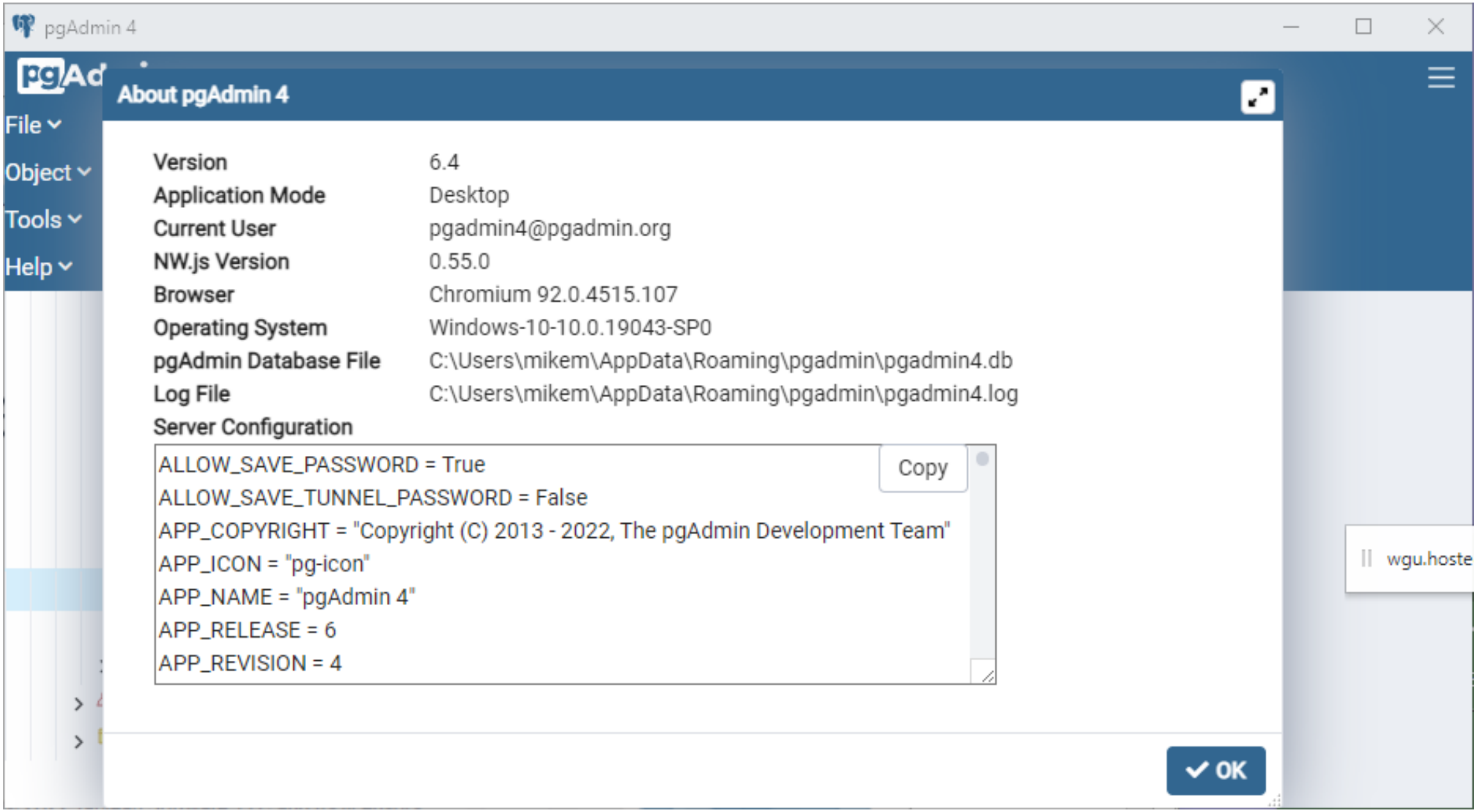
About Tableau

2021.4.4 (20214.22.0213.1102) 64-bit

Tableau Desktop

Professional Edition




pgAdmin 4
File
Object
Tools
Help
About pgAdmin 4
Version 6.4
Application Mode Desktop
Current User pgadmin4@pgadmin.org
NW.js Version 0.55.0
Browser Chromium 92.0.4515.107
Operating System Windows-10-10.0.19043-SP0
pgAdmin Database File C:\Users\mikem\AppData\Roaming\pgadmin\pgadmin4.db
Log File C:\Users\mikem\AppData\Roaming\pgadmin\pgadmin4.log
Server Configuration
Copy
ALLOW_SAVE_PASSWORD = True
ALLOW_SAVE_TUNNEL_PASSWORD = False
APP_COPYRIGHT = "Copyright (C) 2013 - 2022, The pgAdmin Development Team"
APP_ICON = "pg-icon"
APP_NAME = "pgAdmin 4"
APP_RELEASE = 6
APP_REVISION = 4
OK
wgu.hoste

Demonstrate Functionality of Dashboards

- Connect to SQL database
- Connect to external .CSV data
- Create necessary joins
- Create Region, State collection
- Create calculated fields
- **Blend data sources using primary and secondary linking fields**
- Setup Default Properties
  - Color
  - Number Format
- Create detail worksheet
- Create summary worksheet
- Create map worksheet
- Create dashboard by combining all the previous worksheets

Short Tableau Demo

Explain SQL used to support Dashboards

- SQL scripts were used to create the customer churn tables
- SQL scripts were occasionally used to query data during the initial design of the dashboards

Explain Primary & External Data Sources

Customer Churn (SQL)

- Customer (24, 10000)
- Location (5, 8583) “Cities”

Source: Wgu.edu Telcom Churn Data

Region, State Population (.CSV)

- States.csv (4, 52)
- Population.csv (2, 52)

Source: https://www2.census.gov/

Data Demo

Explain How Data Streams Prepared

- The customer churn data contained location data for city, but the state used was just a 2-letter state code
- Needed to find external data of states where the states were broken down by major regions AND where the states were referenced by name and by the 2-letter state code
- In order to calculate customer density, I needed the total population by state and region. There was a population attribute in the original data but was not the total population of the state. I will use the external data to calculate density

Describe Data Alignment

Customer Churn data

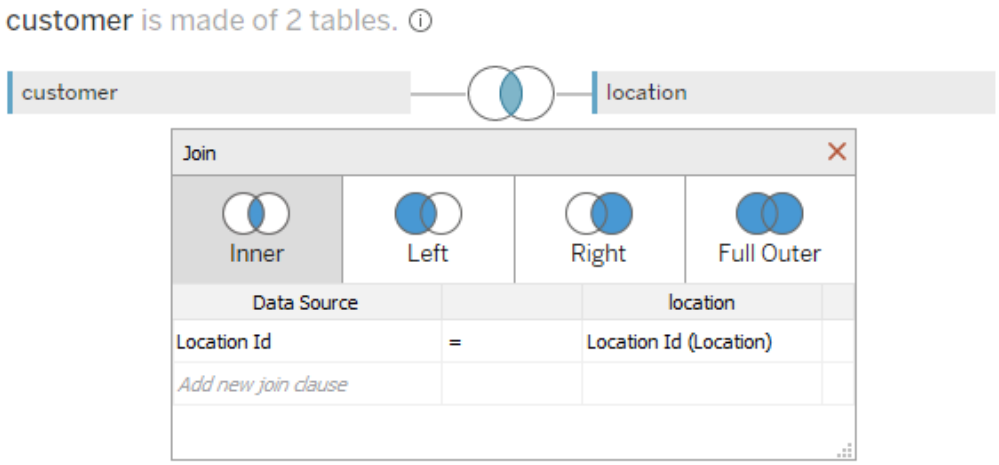
- The customer table is inner joined (or left joined) by a 1:many relationship with the location table.
- The relationship uses an attribute named “**location_id**” which is in both tables (see next slide)

States & Population data

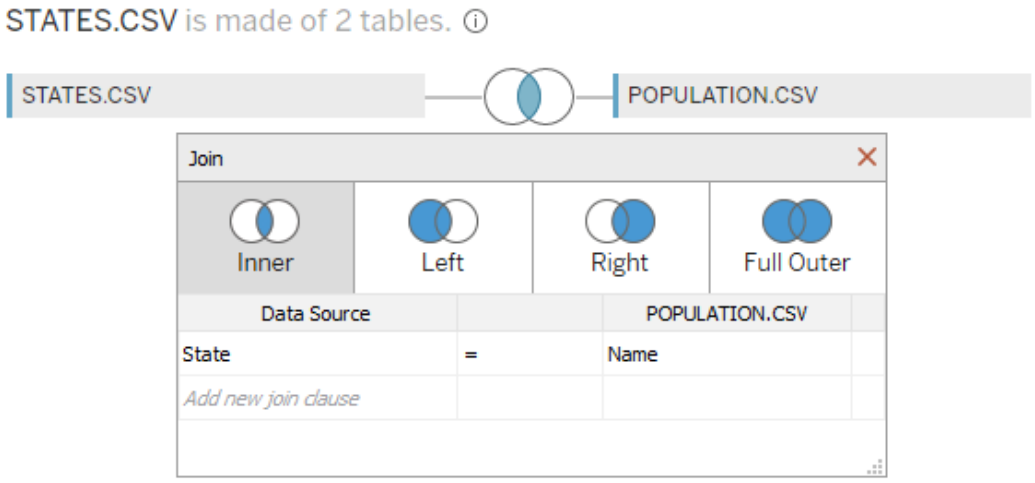
- The states table is inner joined (or left joined) by a 1:1 relationship with the population table.
- The relationship uses attribute “**State**” in the states table and attribute “**Name**” in the population table (see next slide)

Joining Tables

Customer:location join



States:Population join



Explain How Databases Were Created

- Using pgAdmin
- Create database
- Run SQL scripts to create tables
- Import data into tables
- Referential Integrity

pgAdmin Demo

Explain Referential Integrity

- REFERENTIAL INTEGRITY is a database concept that is used to build and maintain logical relationships between tables to avoid logical corruption of data. It is a very useful and important part in RDBMS.
- Usually, referential integrity is made up of the combination of a primary key and a foreign key.
- The main concept of REFERENTIAL INTEGRITY is that it does not allow to add any record in a table that contains the foreign key unless the reference table containing a corresponding primary key.
- Reference:
  - https://www.w3resource.com/sql/joins/joining-tables-through-referential-integrity.php#:~:text=A%20REFERENTIAL%20INTEGRITY%20is%20a,key%20and%20a%20foreign%20key.