# Airport Application

ADVANCED PROGRAMMING PROJECT

# Interface design

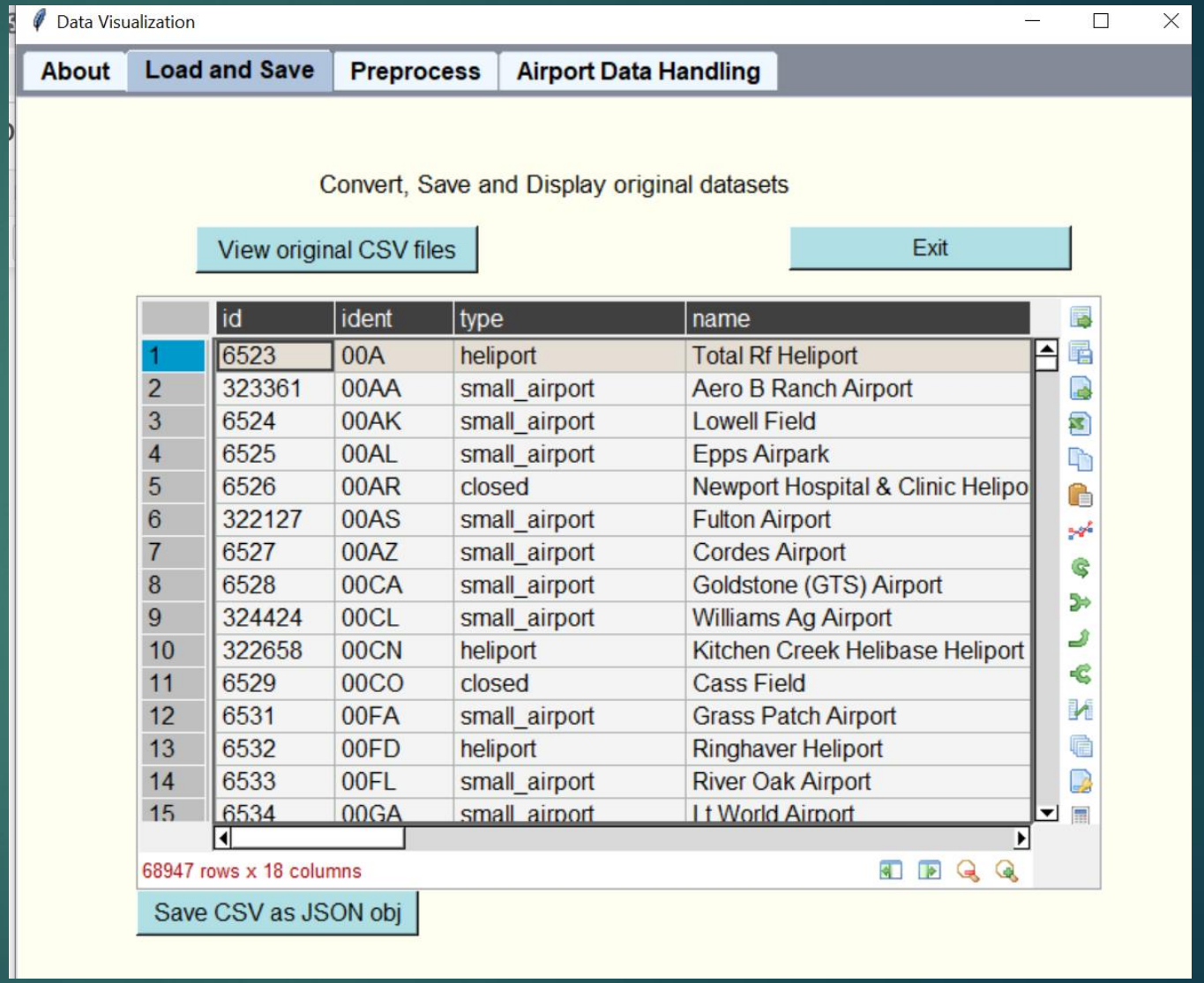
Python data frames, lists, dictionaries, data containers

allow minimum user manual work

Anaconda , Jupyter notebook
feature-rich, robust, and user-friendly

# Load and view data

User can view the files they loaded in data tables.




| | id | ident | type | name |
|---|---|---|---|---|
| 1 | 6523 | 00A | heliport | Total Rf Heliport |
| 2 | 323361 | 00AA | small_airport | Aero B Ranch Airport |
| 3 | 6524 | 00AK | small_airport | Lowell Field |
| 4 | 6525 | 00AL | small_airport | Epps Airpark |
| 5 | 6526 | 00AR | closed | Newport Hospital & Clinic Helipo |
| 6 | 322127 | 00AS | small_airport | Fulton Airport |
| 7 | 6527 | 00AZ | small_airport | Cordes Airport |
| 8 | 6528 | 00CA | small_airport | Goldstone (GTS) Airport |
| 9 | 324424 | 00CL | small_airport | Williams Ag Airport |
| 10 | 322658 | 00CN | heliport | Kitchen Creek Helibase Heliport |
| 11 | 6529 | 00CO | closed | Cass Field |
| 12 | 6531 | 00FA | small_airport | Grass Patch Airport |
| 13 | 6532 | 00FD | heliport | Ringhaver Heliport |
| 14 | 6533 | 00FL | small_airport | River Oak Airport |
| 15 | 6534 | 00GA | small_airport | Lt World Airport |

# Pre-processing the data.

Preparing the initial data set

Managing inconsistences, errors, missing values.

Handle null values

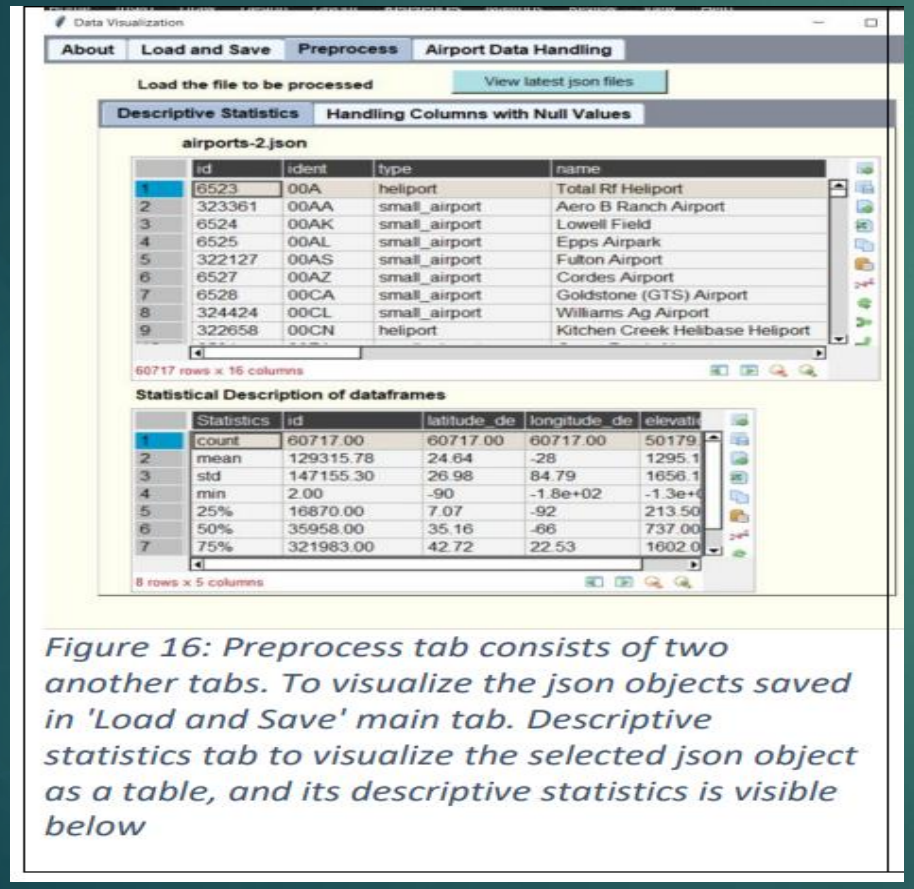



*Figure 16: Preprocess tab consists of two another tabs. To visualize the json objects saved in 'Load and Save' main tab. Descriptive statistics tab to visualize the selected json object as a table, and its descriptive statistics is visible below*

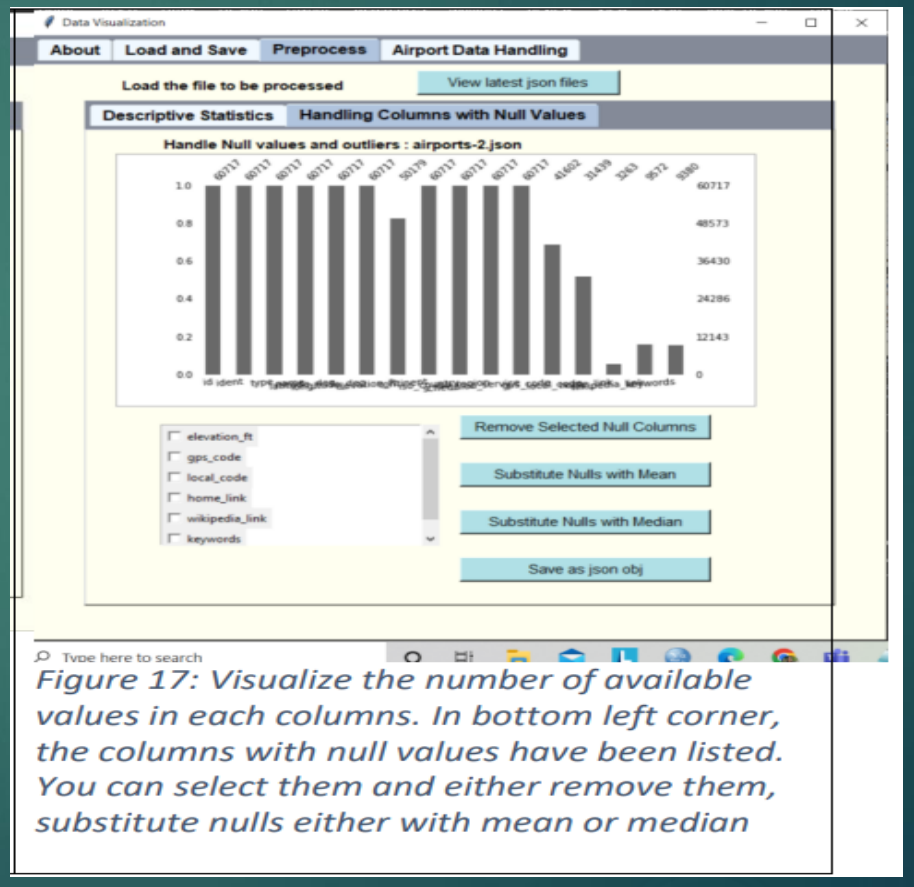



*Figure 17: Visualize the number of available values in each columns. In bottom left corner, the columns with null values have been listed. You can select them and either remove them, substitute nulls either with mean or median*

# Data handling

User requirement to delete closed airport types.

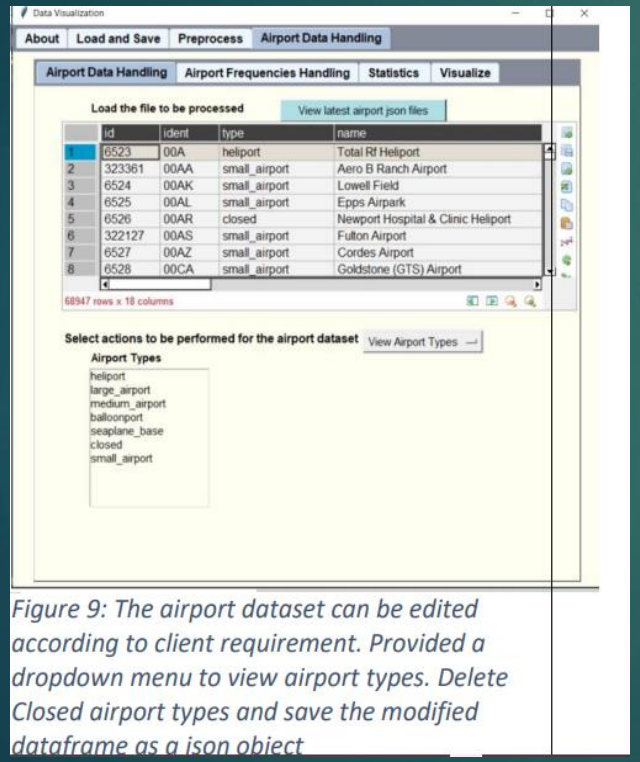



*Figure 9: The airport dataset can be edited according to client requirement. Provided a dropdown menu to view airport types. Delete Closed airport types and save the modified dataframe as a json object*

Filter records on Airport type and country

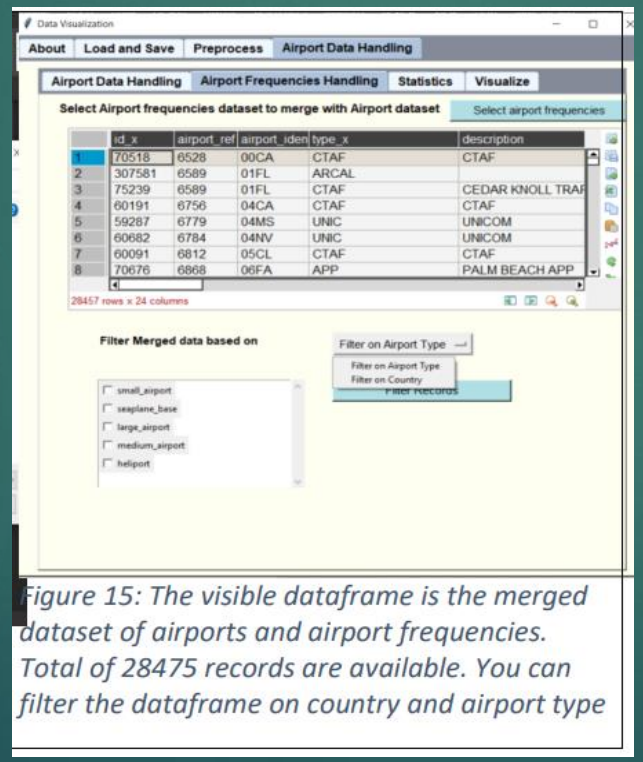



*Figure 15: The visible dataframe is the merged dataset of airports and airport frequencies. Total of 28475 records are available. You can filter the dataframe on country and airport type*

Handle statistical records

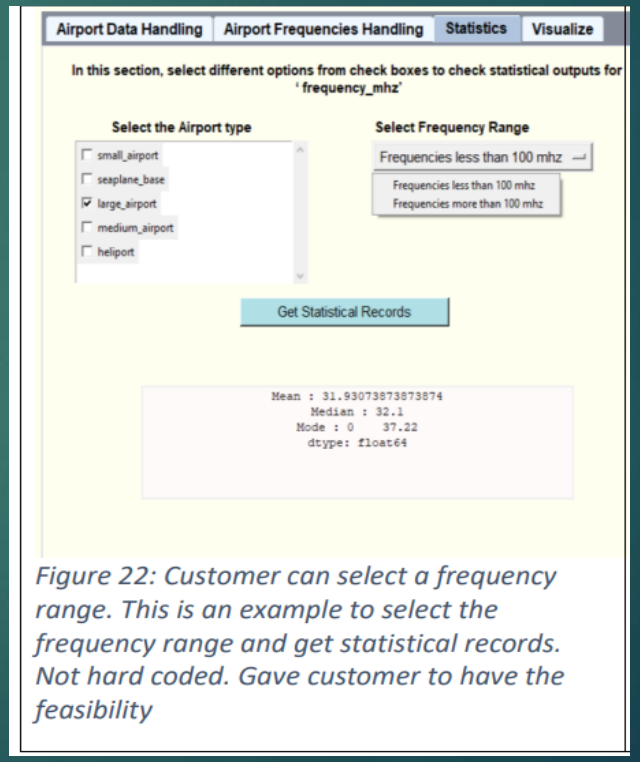



*Figure 22: Customer can select a frequency range. This is an example to select the frequency range and get statistical records. Not hard coded. Gave customer to have the feasibility*

# Data visualization

Seaborn is a high-level interface for drawing statistical graphics with Matplotlib

Bar graphs as exploratory data visualization tool for comparing subgroups in the data to display the results.

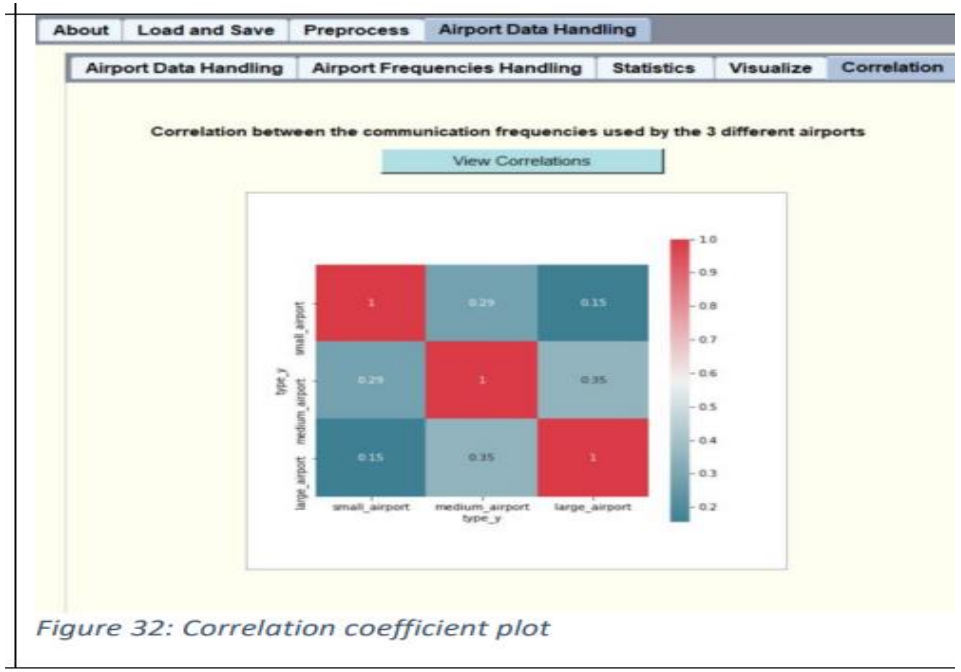



Figure 32: Correlation coefficient plot

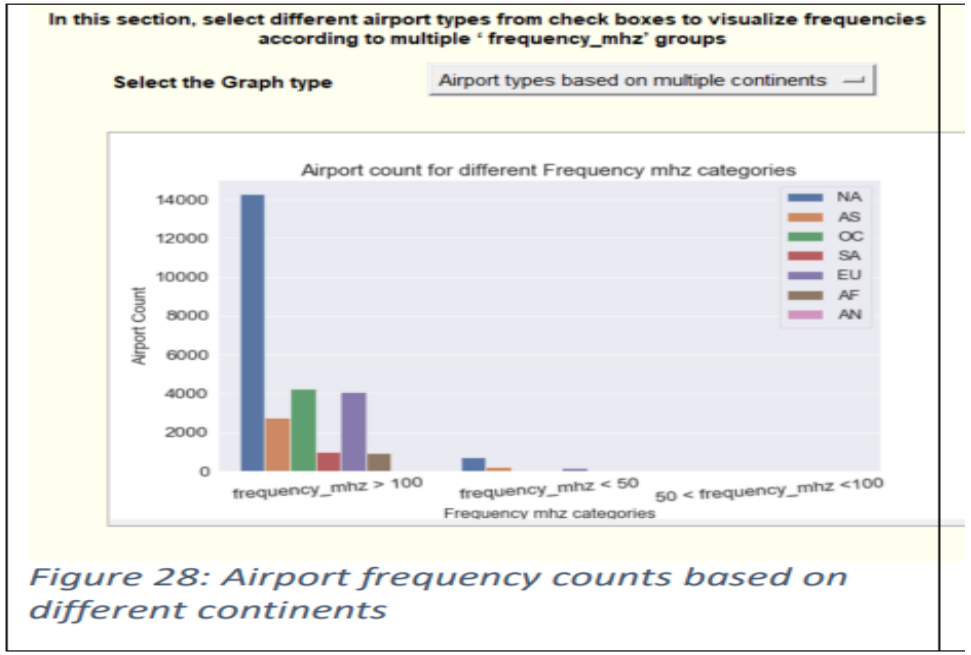



Figure 28: Airport frequency counts based on different continents

# Correlation between each airport

Thank you!!!!!

Q&A