Individual Activity Report

1st Alan Babu Manuel
x23157143@student.ncirl.ie
2nd Melin Mary Lalu
x23185104@student.ncirl.ie
Vishnunath Nharekatt
x22234217@student.ncirl.ie

We are selected three data sets which are Air pollution ,Sea level rise,CO2 emissions.Each member of this group was work on their individual data set.Here explains the work of each members in their data sets.

I. Air Pollution Data Set – Alan Babu Manuel[Student ID-x23157123]

Summary:

This report outlines the contributions I made to the project of selecting a global air pollution dataset, storing it in MongoDB and performing ETL data pipeline for the air pollution data. The pipeline consists of four main tasks:

1. **Selecting Dataset and Storing in MongoDB**: I have selected Global Air pollution dataset from Kaggle and stored it in MongoDB under the Database name DAP_Project
2. **Extracting data from MongoDB:** I developed a Python script using pymongo to connect to a MongoDB database and extract air pollution data. The script retrieves all documents from a specific collection and converts them into a Pandas DataFrame
3. **Transforming the data**: Another Python script performs data cleaning and transformation steps on the extracted DataFrame. This includes handling missing values, removing duplicates.
4. **Loading data into PostgreSQL:** The final script utilizes psycopg2 to connect to a PostgreSQL database. It reads the transformed data from a CSV file (output of the transform script) and loads it into a designated table within the PostgreSQL database. The script also includes error handling and logging capabilities

Details of My Work:

- **Developed Python scripts for each stage:** I wrote Python code using the luigi library for task orchestration. Each script defines a separate task, making the pipeline modular and reusable.

- **Implemented data extraction logic:** I used pymongo to connect to MongoDB and retrieve air pollution data. The script converts the retrieved data into a Pandas DataFrame for further processing.
- **Performed data cleaning and transformation:** I implemented data cleaning steps like handling missing values and removing duplicates. Additionally, the script includes a commented-out section for potential outlier removal techniques.
- **Loaded data into PostgreSQL:** I utilized psycopg2 to establish a connection with a PostgreSQL database. The script reads the transformed data from a CSV file and inserts it into a designated table. The script also incorporates error handling to manage potential issues during data insertion.
- **Ensured code clarity and maintainability:** I used clear variable names, comments, and proper code structure to enhance code readability and maintainability for future modifications.
- From PostgreSQL loaded the data into a numpy dataframe and performed further data visualization techniques for insights.

Overall, my contributions established a functional data pipeline that extracts air pollution data from MongoDB, transforms it for analysis, and loads it into a PostgreSQL database. This provides a foundation for further data exploration and analysis tasks within the project.

II. Sea level rise Dataset -Melin Mary Lalu[Student ID-x23185104]

This report outlines the contributions I made to the project of selecting a global sea level rise dataset, storing it in MongoDB and performing ETL data pipeline for the sea level rise data. The pipeline consists of four main tasks:

1. **Selecting Dataset and Storing to MongoDB**: I have selected sea level rise dataset from Kaggle and stored it in MongoDB under the Database name DAP_Project
2. **Extracting data from MongoDB:** I created a Python script using pymongo to connect to a MongoDB database and extract climate change data.
3. **Transforming the data**: Build another Python script for handling the outliers.
4. **Loading data into PostgreSQL:** The completed script use psycopg2 to connect to a PostgreSQL database. It gets the transformed data from a CSV file and loads it into a specified table[climate_analysis] in the PostgreSQL database.

Details of My Work:

- **Developed Python scripts in each stage:** I develop Python code using the luigi library for each task. Each script defines a separate task and it is reusable.

- **Implemented data extraction logic:** I used pymongo to connect to MongoDB and retrieve sea level rise data. The script converts the retrieved data into a Pandas DataFrame for further processing.
- **Performed data cleaning and transformation:** Checked all the data and remove the outliers using IQR Method.
- **Loaded data into PostgreSQL:** I used psycopg2 to connect to a PostgreSQL database and the script reads data from a CSV file and inserts it in a table in postgresql.
- **Visualization and Modelling:** Using different methods for visualize the data and build some models using machine learning algorithms.
- **Ensured code clarity:** I used specific variable names, comments, and suitable code structure to improve code readability and will use for future changes.

Overall, my efforts built a feasible data process that extracts Sea level rise data from MongoDB, prepares it for analysis, and puts it into a PostgreSQL database.
This serves as a foundation for future data exploration and analysis tasks within the project.

III. CO2 Emission Dataset- Vishnunath Nharekatt[Student ID- x23157143]

This report outlines the contributions I made to the project of selecting a global CO2 emission dataset, storing it in MongoDB and performing ETL data pipeline for the air pollution data. The pipeline consists of four main tasks:

1. **Selecting Dataset and Storing in MongoDB**: I have selected Global CO2 emission dataset from global CO2 emission website and stored it in MongoDB under the Database name DAP_Project
2. **Extracting data from MongoDB:** I developed a Python script using pymongo to connect to a MongoDB database and extract air pollution data. The script retrieves all documents from a specific collection and converts them into a Pandas DataFrame
3. **Transforming the data** : Another Python script performs data cleaning and transformation steps on the extracted DataFrame. This includes handling missing values, removing duplicates.
4. **Loading data into PostgreSQL:** The final script utilizes psycopg2 to connect to a PostgreSQL database. It reads the transformed data from a CSV file (output of the transform script) and loads it into a designated table within the PostgreSQL database. The script also includes error handling and logging capabilities.

Details of My Work:

- **Developed Python scripts for each stage:** I used Python code using the luigi library for the task. Each script defines a separate task, making the pipeline reusable.

- **Implemented data extraction logic:** I used pymongo to connect to MongoDB and retrieve CO2 emission data. The script converts the retrieved data from the MongoDB into a Pandas DataFrame for further processing.
- **Performed data cleaning and transformation:** I implemented data cleaning steps like handling missing values, renamed some column name for better identification, object values were converted to the proper data type for mathematical operations, converting categorical variables into a numerical format using label encoding suitable for modelling and removing duplicates. Additionally, the script includes some section for potential outlier removal techniques.
- **Loaded data into PostgreSQL:** I utilized psycopg2 to establish a connection with a PostgreSQL database. The script reads the transformed data from a CSV file and inserts it into a designated table. The script also incorporates error handling to manage potential issues during data insertion.
- **Ensured code clarity and maintainability:** I used clear variable names, comments, and proper code structure to enhance code readability and maintainability for future modifications.
- From PostgreSQL loaded the data into a NumPy dataframe and performed further data visualization techniques for insights.
- **Post Processing:** After all the pre-process down by other group members I combined the three data sets that stored in the Postgres by setting 'Country' as the primary key and retrieve the combined data from the Postgres and done some visualizations.

Overall, my efforts built a feasible data process that extracts CO2 emission data from MongoDB, prepares it for analysis, and puts it into a PostgreSQL database in docker. This serves as a foundation for future data exploration and analysis tasks within the project.

Each team member prepares their content for the report and the PPT. Finally, we combined that content. Each member in the group work on all part of their data set.