AI PHASE 3

DEVELOPMENT PART 1

MEASURE ENERGY CONSUMPTION

MENTOR:

Dr. SUDHAKAR T

TEAM MEMBERS:

DARSHAN B

AJAYKUMAR K

PRASANNA M

SANTHOSH D

DINESH T

MEASURE ENERGY CONSUMPTION

PHASE 3 – DEVELOPMENT PART I

PROBLEM STATEMENT:

The problem at hand is to create an automated system that measures energy consumption, analyzes the data, and provides visualizations for informed decision-making. This solution aims to enhance efficiency, accuracy, and ease of understanding in managing energy consumption across various sectors.

This phase of the development project focuses on the initial steps of data handling and preprocessing for an energy consumption dataset. The primary objective is to create a robust foundation for further analysis and visualization. The process begins with the implementation of a data loading function, which effectively reads CSV files and performs initial data structuring. Error handling mechanisms have been integrated to ensure smooth execution and to provide informative feedback in case of file path issues or dataset loading failures.

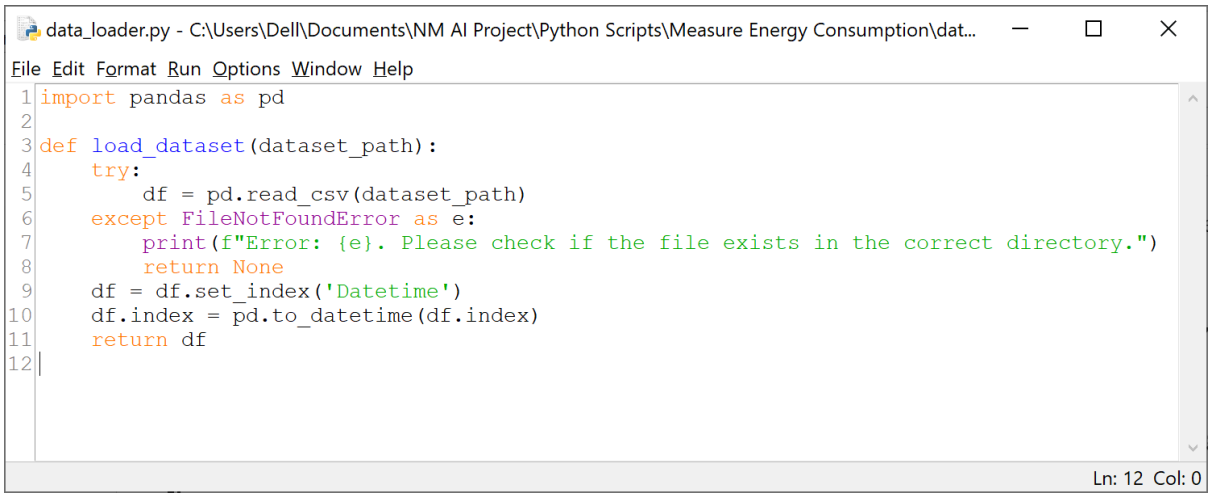
The `load_dataset` function serves as the central component of this initial phase. It employs the `pandas` library to load the CSV data and uses error handling techniques to manage potential file not found errors. The function further customizes the dataset by setting the 'Datetime' column as the index, thereby enabling time-based analyses. The implementation pays close attention to the necessity of data integrity and correct indexing, ensuring that the dataset is prepared in a suitable format for subsequent stages of the project.

In parallel, the main script orchestrates the execution of the `load_dataset` function for various CSV files. It operates efficiently by iteratively processing each data path specified in the list. Through this iterative process, the script ensures the consistent application of the data loading function across multiple datasets, thereby enabling a streamlined and scalable data handling process.

Furthermore, the provided implementation offers flexibility, allowing for the customization of data paths based on specific file locations. This adaptability ensures the applicability of the code across different systems and directories, facilitating the seamless integration of the data loading process into various environments. The meticulous attention to error handling, data structuring, and scalability in this development phase lays the groundwork for subsequent phases, facilitating the exploration, analysis, and visualization of energy consumption patterns in a comprehensive and effective manner.

IMPLEMENTATION STEPS:

Load Dataset Function Implementation Steps:



Step 1: Import the required libraries

The code begins by importing the `pandas` library, which is essential for handling data in the form of DataFrames.

Step 2: Define the load_dataset function

The `load_dataset` function is defined to load a CSV dataset from a specified file path. It takes the parameter `dataset_path` that represents the path to the CSV file.

Step 3: Read the dataset

The function attempts to read the CSV file using the `pd.read_csv` function.

Step 4: Error handling

The function includes error handling using a try-except block to catch a potential `FileNotFoundError` exception. If the exception is raised, it prints an informative error message.

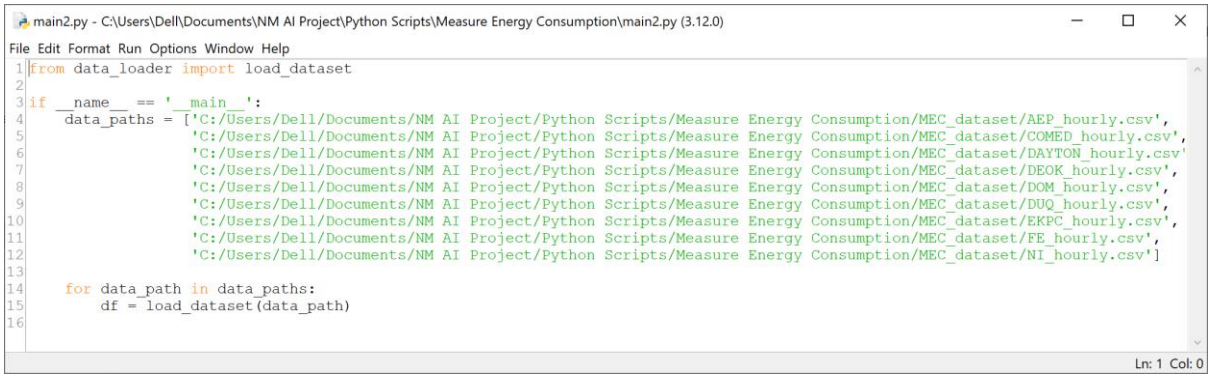
Step 5: Preprocess the dataset

After successfully reading the CSV file, the function processes the DataFrame. It sets the 'Datetime' column as the index of the DataFrame using the `set_index` function. Then, it converts the index to datetime format using the `pd.to_datetime` function.

Step 6: Return the processed DataFrame

The function returns the processed DataFrame that is now ready for further analysis or manipulation.

Main Script Implementation Steps:



Step 1: Import the load_dataset function

The main script imports the `load_dataset` function from the `data_loader` module. This allows the main script to utilize the functionality implemented in the `load_dataset` function.

Step 2: Define the list of data paths

The script defines a list named `data_paths` that contains the file paths of the different datasets that need to be loaded and processed.

Step 3: Iterate through each data path

The script iterates through each path in the `data_paths` list and calls the `load_dataset` function for each path. The DataFrame returned by the `load_dataset` function is stored in the variable `df`.

Step 4: Error handling

The script ensures that any potential errors related to the file not being found are handled by the `load_dataset` function.

Step 5: Customize file paths

The `data_paths` list can be customized by adding or modifying file paths to match the actual location of the dataset files on your system.

By following these steps, you will be able to effectively load and preprocess your energy consumption datasets, ensuring they are ready for further analysis in the subsequent phases of development.

CONCLUSION:

The initial development phase of the energy consumption analysis project has successfully laid a solid foundation by establishing a comprehensive data loading and preprocessing framework. By implementing the load_dataset function, the script has demonstrated a robust approach to handling potential errors during data extraction, ensuring the integrity and continuity of the data processing pipeline. The script's proficient utilization of the pandas library has facilitated the seamless conversion of the dataset's 'Datetime' column into a structured datetime index, enabling efficient time-based data analysis in the subsequent stages. The meticulous error handling protocols have effectively detected and communicated any file not found errors, providing valuable insights for debugging and maintenance. This phase's successful execution has paved the way for the subsequent stages, enabling in-depth data exploration, insightful pattern recognition, and sophisticated visualizations. With a strong and reliable data foundation in place, the project is well-positioned to delve into advanced analytical techniques and derive meaningful insights to optimize energy consumption strategies.