Milestone 1 Completion Report: Data Preparation & Feature Engineering

1. Data Ingestion and Verification

The initial phase involved loading and validating the dataset to ensure its quality.

- **Data Loading**: I have developed Python scripts using the Pandas library to ingest the **train_FD001.txt** file from the NASA CMAPSS dataset. Correct column headers were assigned based on the official **readme.txt** documentation.
- **Integrity Check**: An immediate verification was performed, which confirmed the dataset's integrity and, most importantly, the **absence of missing values**.

2. Preprocessing Techniques and Feature Engineering

Several preprocessing techniques were applied to clean the data and engineer the target variable.

- **RUL Calculation**: The critical target variable, Remaining Useful Life (RUL), was engineered for the training data. This was calculated for each cycle by subtracting the current cycle number from the engine's final cycle at failure. This successfully produced the Computed RUL targets for all engine cycles.
- **Visual Analysis**: I have used **matplotlib** and **seaborn** to visualize sensor data trends over time. This allowed us to confirm that many sensors showed clear upward or downward trends, indicating their value as predictive features.
- **Feature Removal**: I calculated the standard deviation of each sensor column and dropped those with a standard deviation of zero, as they provide no predictive information. This resulted in Cleaned and preprocessed CMAPSS sensor data.

3. Data Transformation for LSTM Model

The final step was to transform the data into a format suitable for a time-series model.

- **Sequence Generation**: I converted the flat time-series data into overlapping sequences or "rolling windows". I used a sequence length of 50, meaning each data sample now consists of 50 consecutive time steps of sensor readings.
- **Final Structure**: This transformation produced a 3D NumPy array for the features (X_train) and a corresponding 1D array for the RUL labels (y_train), which is the required input for an LSTM model. The process ensures the Correctness of rolling window sequence generation.