Working Principle of the Emissions Analysis Code

1. Data Loading and Preparation

- Load the dataset from the CSV file (Emission.csv) into a Pandas DataFrame.
- Clean column names by removing leading and trailing spaces for consistency.
- Convert specific columns to numeric data types, coercing invalid entries to NaN for better error handling.
- Add new calculated columns:
    - **Speed_mps**: Convert speed from km/h to m/s.
    - **Power**: Multiply Force by Speed_mps.
    - **NMHC**: Calculate Non-Methane Hydrocarbons by subtracting CH4.Dilute.mass from THC.TailPipe.mass.

2. Phase-wise Summation

- Define a function calculate_sum(start, end) to sum the specified columns over a given row range.
- Split the data into three phases:
    - Phase 1: Rows 0 to 600
    - Phase 2: Rows 601 to 1200
    - Phase 3: Rows 1201 to 1800
- Compute and print the summed values for each phase.

3. Speed and Distance Analysis

- Calculate mean speeds for each phase.
- Compute distances for each phase using the formula:
  distance=mean speed×time (in hours). Here, the time is 600 seconds, converted to hours (600 / 3600).

4. Phase-wise Distance-Weighted Multiplications

- Multiply the distance of each phase by its summed values to generate distance-weighted results.

5. Emissions per Kilometer

- Calculate the emissions per kilometre for each phase as:

$$\text{emissions per km} = \frac{\text{sum of emissions in the phase}}{\text{distance of the phase}}$$

- Weight the emissions per kilometre for each phase by their respective weights (25%, 50%, 25%).
- Sum these weighted values to obtain the overall weighted emissions per kilometer.

6. Efficiency Calculations

- Convert columns related to catalytic efficiency to numeric types.
- Define a function calculate efficiency(start, end) to compute the average efficiency for a given row range.
- Calculate phase-wise efficiencies and total efficiency across all phases.
- Calculate phase-wise efficiencies and total efficiency across all phase

$$\text{Efficiency NMHC} = \left(\frac{\text{baseline} - \text{NMHC}}{\text{baseline}}\right) \times 100$$

7. Statistical Analysis

- Compute the mean, standard deviation, minimum, maximum, and variance for:
  - Cell temperature (CellTemp column).
  - Relative humidity (HumsRel column).

8. Plotting

- Generate multiple line plots for various emissions parameters, including:
  - THC, CO, NOX mass (Pre-PrimCat and TailPipe) vs. time.
  - Catalytic efficiency vs. time for THC, CO, NOX.
  - Speed, force, power, and acceleration vs. time.
  - Angular acceleration vs. time using assumed vehicle mass and radius.

9. Exporting Results to PDF

- Use the reportlab library to format results into a structured PDF:
  - Organize the results by section (e.g., "Sum values for phases", "Efficiency values").
  - Add headers and format tables with styles.
- Use matplotlib.backends.backend_pdf.PdfPages to save all plots into a single PDF.

10. Summary Results

- Organize results into a dictionary (results_dict) for easier formatting and future modifications.
- Save the summary and detailed analysis to a PDF file (3_2emissions.pdf) using the previously defined save_results_to_pdf function.

11. Conclusion

The program combines emission data analysis, visualization, and result reporting:

- It calculates metrics for speed, distance, emissions, and efficiency across different driving phases.
- Results are saved as structured PDFs for both numerical analysis and graphical trends, allowing insights into vehicle emissions and performance.