Task description:
Generate daily demand forecasts for 30 days starts from 09-01-2025

- Objective: build daily forecasts for each SKU using only the provided time series, for the period 2024-01-01 to 2025-08-31 (inclusive).
- Data: mock_test.csv
    - DATE (YYYY-MM-DD), VALUE, ITEM_CODE, NAME, GROUP, NOTES
    - Use only history up to 2025-08-31 for model training/validation.

Evaluation

- Primary metrics: MAE (Mean Absolute Error) and MAPE.
- You may report additional metrics appropriate to your chosen models (e.g., RMSE, pinball loss, coverage).

Required Outputs

- Forecast CSV file
- Columns:
    - DATA: forecast date in YYYY-MM-DD
    - ITEM_CODE: item code
    - FORECAST: point forecast for daily units
    - LOWER_95 / UPPER_95: 95% prediction interval bounds
    - LOWER_97 / UPPER_97: 97% prediction interval bounds
    - LOWER_99 / UPPER_99: 99% prediction interval bounds
- Notes:
    - No missing values.
    - Non-negative forecasts; document any rounding policy if applied.
    - Intervals should be consistent with your modeling approach, briefly state how you computed them.

Workflow (expected steps)

- Exploratory Data Analysis (2–3 core visuals)
    - Examples: daily series plot; seasonality/day-of-week pattern; distribution/zero-share by SKU.
- Data processing
    - Scaling/transformations if used, handling of zeros/outliers etc.
- Feature engineering (as appropriate for your chosen models)
    - Calendar (DOW/MOY/holidays), lags/rolling stats, event windows, etc.
- Model training and validation
    - Use time-aware validation (e.g., rolling or expanding splits). Report MAE and MAPE.
- Forecast generation
    - Produce daily forecasts for 2025-09-01..2025-09-30 with intervals at 95/97/99.

Deliverables

- Code (notebook or script) that fully reproduces your results.
- The per-SKU forecast CSVs.
- A short write-up (or notebook section) summarizing (not required):
    - Modeling choices and rationale.
    - Validation setup and metrics.
    - How prediction intervals were constructed.
    - Any assumptions/limitations.

NOTE: You can use any models/approaches, there are no constraints. If you choose to use AI tools to help you with the test task (if you need to speed up the process), please add this information to notes stating what parts of the process were done with AI.

Test Dataset Description

- File: mock_test.csv
- Granularity: daily unit sales per SKU
- Date coverage: 2024-01-01 to 2025-08-31 (some SKUs may have shorter history)
- Source: synthetic dataset emulating retail demand patterns

Schema (columns)

- DATE (YYYY-MM-DD): calendar date (no calendar gaps)
- VALUE (int ≥ 0): units sold per day
- ITEM_CODE (str): SKU code
- NAME (str): SKU name
- GROUP (str): product group (APPLIANCES, HEATING, THERMOSTATS)
- NOTES (str, optional): service note (may be empty)

Column interpretation and expectations

- VALUE: daily sales (not cumulative). Zeros are valid and indicate no sales that day.
- GROUP (str): product type category (e.g., APPLIANCES, HEATING, THERMOSTATS), used purely for item type grouping, not indicative of demand behavior.
- NOTES: may flag SKU-specific aspects (e.g., limited history). The field may be empty.

Technical details

- Data types: dates as YYYY-MM-DD, VALUE is integer, other fields are strings.
- Sorting: rows sorted by DATE, then by ITEM_CODE.
- Missing values: none, zero values are valid data.

What to consider for analysis/modeling

- Different history lengths: some SKUs have shorter history—affects features and validation strategy.
- Intermittency: for SKUs with many zeros, prefer methods/metrics robust to sparse demand.
- Seasonal/calendar effects: calendar features (DOW, MOY, proximity to events/seasons) are useful to test.
- Structural changes: consider markups or features for level shifts/change-points where applicable.
- Outliers: one-off spikes can affect training; consider robust losses or event features.