sonnen Data Engineering Coding Challenge:

You have been provided with a CSV file containing battery time series data. The data has common data quality issues such as missing values, incorrect data types, and inconsistent formatting. Your task is to develop an application that will transform and cleanse the data, and then write the cleaned data to an output file. Feel free to make use of common libraries for data engineering tasks. We don't expect the perfect solution, but rather want to create a starting point for a technical discussion. You don't have to spend more than 4 – 6 hours on the task. If any questions arise, feel free to ask. Good luck with the challenge ☺.

Instructions:

1. Write a data pipeline in Python that performs the following tasks:
    - Load the CSV file.
    - Perform data transformation and cleansing operations to address the following issues:
        i. Missing values: Replace missing values with appropriate default values (e. g. Null values)
        ii. Incorrect data types: Convert columns to their appropriate data types (e. g. integers).
        iii. Remove duplicates and corrupt values from the dataset.
    - After cleansing, calculate the total grid_purchase and grid_feedin over all batteries for each hour of the day.
    - Add a column to your dataframe that indicates the hour with the highest grid_feedin of the day (e. g. a Boolean value)
    - Write the transformed data to an output file in CSV format.

2. Containerize your application using Docker. Write a Dockerfile that:
    - Uses an appropriate base image.
    - Copies your application code into the Docker image.
    - Sets the entry point command to execute your application.

3. Build a Docker image from your Dockerfile and tag it appropriately.

4. Run your Docker image locally to perform the transformation.

5. Write a documentation on how to build and run the image locally.

6. Upload all resources to your public GitHub repository and send us the link.

Deliverables:

- A Python application that performs data transformation and cleansing.
- A Dockerfile for containerizing your application.
- The transformed data written to an output file.
- A README file as a documentation and instruction manual.