NYC Taxi Trip Data Analysis using PySpark

Objective:

Evaluate the candidate's ability to work with large datasets using PySpark by assessing skills in data ingestion, cleaning, transformation, aggregation, and performance optimisation.

Dataset:

- **Primary Dataset:** Use the NYC Taxi Trip Dataset yellow taxi trip records
- **Timeframe:** Only use the data from December 2024

Task Requirements:

1. **Data Ingestion:**
   - Ingest the NYC Taxi Trip dataset into a Spark DataFrame using PySpark.
   - Either infer the schema automatically or define it explicitly (e.g., specifying data types for timestamps, numerical values, etc.).
2. **Data Cleaning:**
   - Identify and filter out rows with missing or obviously invalid data (e.g., negative fares, impossible timestamps, or distances).
   - Convert pickup and drop-off time columns into proper timestamp data types.
   - Ensure that numerical fields (e.g., fare amounts, trip distances) are in the expected format.
3. **Data Transformation & Feature Engineering:**
   - Calculate a new column representing the trip duration (e.g., difference between drop-off and pickup times).
   - Extract additional features such as:
     - Hour of the day.
     - Day of the week.
   - Create a categorical column that classifies trips as ‘short’, ‘medium’, or ‘long’ based on distance thresholds.
4. **Aggregation and Analysis:**
   - Compute key statistics (mean, median, min, max) for trip duration, fare amount, and distance.
   - Group the data by hour/day and compute:
     - Total number of trips.
     - Average fare and trip duration per time interval.
   - Identify peak usage hours and any observable trends within December 2024.
5. **Performance Optimization:**
   - Demonstrate the use of caching, partitioning, and/or bucketing strategies to optimise query performance.
   - Explain or show how Spark configurations or data format choices can improve processing times.
   - Discuss how you would further optimise or scale this solution in a production environment which would include all taxi data starting from 2009.
6. **Reporting & Visualisation:**
   - Create a well-documented notebook with markdown cells that clearly explain each step of your process.
   - Include charts or graphs (using built-in visualisation tools or libraries like matplotlib) to present:
     - Distribution of trip durations.
     - Trip counts over different hours/days within December 2024.

- Additional visual insights that support your analysis.

Deliverables:

1. A well-documented notebook containing:
   - The complete PySpark code.
   - Explanatory markdown text outlining your approach and decisions.
   - Visual outputs (charts/graphs) that support your analysis.
2. A brief markdown section discussing the optimisations applied and potential further improvements.

Evaluation Criteria:

- Accuracy and efficiency in processing the dataset.
- Clarity, modularity, and documentation of the code.
- Ability to handle data quality issues and apply appropriate transformations.
- Use of Spark optimisation techniques.
- Quality of the analysis and the insights drawn from the data.