Data Engineering Assessment Task

Data Description

Dataset 1: Employee Details

Column Name	Description
EmployeeID	Unique identifier for each employee. This is the primary key for merging datasets.
Name	Full name of the employee.
Department	Department in which the employee works (e.g., Engineering, Marketing, HR).
DateOfJoining	The date when the employee joined the company. Some records may have invalid or missing values.
Email	Employee's email address. Some records may have invalid or missing values.

Dataset 2: Employee Salary Details

Column Name	Description
EmployeeID	Unique identifier for each employee. Matches the EmployeeID in the Employee Details dataset.
Salary	Employee's salary (in USD). Can vary across records for the same employee due to updates.
PerformanceRating	Rating of the employee's performance (on a scale from 1 to 5, where 5 is the highest).
SalaryStartDate	The date when the corresponding salary took effect.

Task Description

1. **Load Both CSV Files:**
   - Load the Employee Details and Employee Salary Details datasets into separate pandas DataFrames.
2. **Merge the Data:**
   - Use the EmployeeID column as the key to merge the two datasets.
3. **Clean the Data:**
   - Identify and resolve issues such as:
     - Missing values in either dataset.
     - Duplicate rows (simulate by duplicating some rows if necessary).
     - Inconsistent or invalid data (e.g., invalid DateOfJoining values).
     - Reformat all date columns to **YYYY-MM-DD** format.
4. **Analyze the Data:**
   - Create a new column for SalaryBand (e.g., Low, Medium, High) based on Salary ranges.
   - Calculate the salary growth percentage for employees compared to their most recent salary only.
   - Filter out employees with a PerformanceRating below 3.
   - Create a new cleaned and filtered dataset containing (EmployeeID, Name,

DateOfJoining, CurrentSalary, PerformanceRating, SalaryBand, SalaryGrowthPercentage).

5. **Load into Database:**
   - Finally, load the created dataset into a postgres database running on Docker.

Deliverables

- Source code.
- README file with:
  - Instructions for running the ETL locally or via Docker.
- Docker images (optional but encouraged).

Bonus:

- Provide a Docker Compose file that includes both the ETL service and the PostgreSQL database. Ensure that the ETL service starts only after confirming the PostgreSQL database is up and running.