Data Engineer Coding Test

This is a coding test for Data Engineer role.

Source Data to process

```
1. order_detail.csv
```

Name	Type	Note
order_created_timestamp	timestamp	format YYYY-MM-DD HH:MM:SS
status	string	
price	integer	
discount	float	
id	string	
driver_id	string	
user_id	string	
restaurant_id	string	

```
2. restaurant_detail.csv
```

Name	Type	Note
id	string	
restaurant_name	string	
category	string	
esimated_cooking_time	float	
latitude	float	
longitude	float	

Business requirements

- Create two tables in postgre database with the above given column types.
    - order_detail table using **order_detail.csv**
    - restaurant_detail table using **restaurant_detail.csv**
- Once we have these two tables in postgre DB, ETL the same tables to Hive with the same names and corresponding Hive data type using the below guidelines
    - Both the tables should be **external table**.
    - Both the tables should have **parquet file format**.
    - restaurant_detail table should be partitioned by a column name **dt** (type string) with a static value **latest**.
    - order_detail table should be partitioned by column named **dt** (type string) extracted from **order_created_timestamp** in the format **YYYYMMDD**.

```
Example of dt column

order_created_timestamp: "2019-06-08 17:31:57"
dt: "20190608"
```

- After creating the above tables in Hive, create two new tables **order_detail_new** and **restaurant_detail_new** with their respective columns and partitions and add one new column for each table as explained below.

Table Name	New Column Name	Logic
order_detail	discount_no_null	replace all the NULL values of**discount** column with 0
restaurant_detail	cooking_bin	using **esimated_cooking_time** column and the below logic

esimated_cooking_time	cooking_bin
10-40	1
41-80	2
81-120	3
greater than 120	4

```
Final column count of each table (including partition column):
1. order_detail = 9
2. restaurant_detail = 7
3. order_detail_new = 10
4. restaurant_detail_new = 8
```

SQL requirements

1. Get the average **discount** for each **category**
2. Row count per each **cooking_bin**

CSV output requirements

Save the above query output to CSV files name **discount.csv** and **cooking.csv**.

Technical Requirements

- Use any of the **big data** / other frameworks (Use Dockers if needed).
- Include a README file that explains how we can deploy your code.