Interview question

Question You are working as a Data Engineer for a company. The sales team has provided you with a dataset containing sales information. However, the data has some missing values that need to be addressed before processing. You are required to perform the following tasks:

1. Load the following sample dataset into a PySpark DataFrame:
2. Perform the following operations:

a. Replace all NULL values in the Quantity column with 0.

b. Replace all NULL values in the Price column with the average price of the existing data.

c. Drop rows where the Product column is NULL.

d. Fill missing Sales_Date with a default value of '2025-01-01'.

e. Drop rows where all columns are NULL.

schema data = [(1, "Laptop", 10, 50000, "North", "2025-01-01"), (2, "Mobile", None, 15000, "South", None), (3, "Tablet", 20, None, "West", "2025-01-03"), (4, "Desktop", 15, 30000, None, "2025-01-04"), (5, None, None, None, "East", "2025-01-05")]

columns = ["Sales_ID", "Product", "Quantity", "Price", "Region", "Sales_Date"]

```
data = [ (1, "Laptop", 10, 50000, "North", "2025-01-01"), (2,
"Mobile", None, 15000, "South", None), (3, "Tablet", 20, None, "West",
"2025-01-03"), (4, "Desktop", 15, 30000, None, "2025-01-04"), (5,
None, None, None, "East", "2025-01-05") ]
columns = ["Sales_ID", "Product", "Quantity", "Price", "Region",
"Sales_Date"]


df = spark.createDataFrame(data, columns)

df.show()
```

```
+--------+-------+--------+-----+------+----------+
|Sales_ID|Product|Quantity|Price|Region|Sales_Date|
+--------+-------+--------+-----+------+----------+
|       1| Laptop|      10|50000| North|2025-01-01|
|       2| Mobile|    null|15000| South|      null|
|       3| Tablet|      20| null|  West|2025-01-03|
|       4|Desktop|      15|30000|  null|2025-01-04|
|       5|   null|    null| null|  East|2025-01-05|
+--------+-------+--------+-----+------+----------+
```

```
df.createOrReplaceTempView("sales_tbl")
```

```
# replace null value in qty with 0

df.fillna({"Quantity":0}).show()
```

```
+--------+-------+--------+-----+------+----------+
|Sales_ID|Product|Quantity|Price|Region|Sales_Date|
+--------+-------+--------+-----+------+----------+
|       1| Laptop|      10|50000| North|2025-01-01|
|       2| Mobile|       0|15000| South|      null|
|       3| Tablet|      20| null|  West|2025-01-03|
|       4|Desktop|      15|30000|  null|2025-01-04|
|       5|   null|       0| null|  East|2025-01-05|
+--------+-------+--------+-----+------+----------+
```

```
from pyspark.sql.types import *
from pyspark.sql.functions import *

# replace null Quanity with 0 using when-otherwise
df.withColumn("Quantity",
when(col("Quantity").isNull(),0).otherwise(col("Quantity"))).show()
```

```
+--------+-------+--------+-----+------+----------+
|Sales_ID|Product|Quantity|Price|Region|Sales_Date|
+--------+-------+--------+-----+------+----------+
|       1| Laptop|      10|50000| North|2025-01-01|
|       2| Mobile|       0|15000| South|      null|
|       3| Tablet|      20| null|  West|2025-01-03|
|       4|Desktop|      15|30000|  null|2025-01-04|
|       5|   null|       0| null|  East|2025-01-05|
+--------+-------+--------+-----+------+----------+
```

```
%sql
-- fill na with 0
select *,coalesce(Quantity,0) from sales_tbl;

# replace null values in price with average column
average = df.agg(avg("Price")).collect()[0][0]
print(average)
```

```
31666.666666666668
```

```
df.fillna({"Price":average}).show()
```

```
+--------+-------+--------+-----+------+----------+
|Sales_ID|Product|Quantity|Price|Region|Sales_Date|
+--------+-------+--------+-----+------+----------+
|       1| Laptop|      10|50000| North|2025-01-01|
|       2| Mobile|    null|15000| South|      null|
|       3| Tablet|      20|31666|  West|2025-01-03|
|       4|Desktop|      15|30000|  null|2025-01-04|
```

```
|       5|   null|    null|31666|  East|2025-01-05|
+--------+-------+--------+-----+------+----------+
```

```
df.withColumn("Price", when(col("Price").isNull(),
average).otherwise(col("Price"))).show()
```

```
+--------+-------+--------+------------------+------+----------+
|Sales_ID|Product|Quantity|             Price|Region|Sales_Date|
+--------+-------+--------+------------------+------+----------+
|       1| Laptop|      10|           50000.0| North|2025-01-01|
|       2| Mobile|    null|           15000.0| South|      null|
|       3| Tablet|      20|31666.666666666668|  West|2025-01-03|
|       4|Desktop|      15|           30000.0|  null|2025-01-04|
|       5|   null|    null|31666.666666666668|  East|2025-01-05|
+--------+-------+--------+------------------+------+----------+
```

```
%sql
select Price from sales_tbl;

# drop rows where product column is null

df.show()
```

```
+--------+-------+--------+-----+------+----------+
|Sales_ID|Product|Quantity|Price|Region|Sales_Date|
+--------+-------+--------+-----+------+----------+
|       1| Laptop|      10|50000| North|2025-01-01|
|       2| Mobile|    null|15000| South|      null|
|       3| Tablet|      20| null|  West|2025-01-03|
|       4|Desktop|      15|30000|  null|2025-01-04|
|       5|   null|    null| null|  East|2025-01-05|
+--------+-------+--------+-----+------+----------+
```

```
# drop rows where product is null
df.filter(col("Product").isNotNull()).show()
```

```
+--------+-------+--------+-----+------+----------+
|Sales_ID|Product|Quantity|Price|Region|Sales_Date|
+--------+-------+--------+-----+------+----------+
|       1| Laptop|      10|50000| North|2025-01-01|
|       2| Mobile|    null|15000| South|      null|
|       3| Tablet|      20| null|  West|2025-01-03|
|       4|Desktop|      15|30000|  null|2025-01-04|
+--------+-------+--------+-----+------+----------+
```

```
%sql
-- drop rows where product is null
```

```
select * from sales_tbl
where Product is not null;
```

```
# drop rows where all columns are null
df.dropna("all").show()
```

```
+--------+-------+--------+-----+------+----------+
|Sales_ID|Product|Quantity|Price|Region|Sales_Date|
+--------+-------+--------+-----+------+----------+
|       1| Laptop|      10|50000| North|2025-01-01|
|       2| Mobile|    null|15000| South|      null|
|       3| Tablet|      20| null|  West|2025-01-03|
|       4|Desktop|      15|30000|  null|2025-01-04|
|       5|   null|    null| null|  East|2025-01-05|
+--------+-------+--------+-----+------+----------+
```

```
# Fill missing Sales_Date with a default value of '2025-01-01'.
df.withColumn("Sales_Date",when(col("Sales_Date").isNull(),'2025-01-
01').otherwise(col("Sales_Date"))).show()
```

```
+--------+-------+--------+-----+------+----------+
|Sales_ID|Product|Quantity|Price|Region|Sales_Date|
+--------+-------+--------+-----+------+----------+
|       1| Laptop|      10|50000| North|2025-01-01|
|       2| Mobile|    null|15000| South|2025-01-01|
|       3| Tablet|      20| null|  West|2025-01-03|
|       4|Desktop|      15|30000|  null|2025-01-04|
|       5|   null|    null| null|  East|2025-01-05|
+--------+-------+--------+-----+------+----------+
```

```
df.fillna({"Sales_Date":'2025-01-01'}).show()
```

```
+--------+-------+--------+-----+------+----------+
|Sales_ID|Product|Quantity|Price|Region|Sales_Date|
+--------+-------+--------+-----+------+----------+
|       1| Laptop|      10|50000| North|2025-01-01|
|       2| Mobile|    null|15000| South|2025-01-01|
|       3| Tablet|      20| null|  West|2025-01-03|
|       4|Desktop|      15|30000|  null|2025-01-04|
|       5|   null|    null| null|  East|2025-01-05|
+--------+-------+--------+-----+------+----------+
```

```
pdf = df.toPandas()
```

```
pdf
```

```
# fill null Quantity with 0 in pandas
pdf["Quantity"].fillna(0)
```

```
Out[44]: 0    10.0
1     0.0
2    20.0
3    15.0
4     0.0
Name: Quantity, dtype: float64
```

```
# fill null price with average in Pandas
pdf["Price"].fillna(pdf["Price"].mean())
```

```
Out[46]: 0    50000.000000
1    15000.000000
2    31666.666667
3    30000.000000
4    31666.666667
Name: Price, dtype: float64
```

```
# drop row where Product column is null
pdf.dropna(subset=["Product"])
```

```
# drop rows where all columns are null in Pandas
pdf.dropna(how="all")
```