Snowflake Query Tracker Result

Demand Forecasting & Anomaly Detection

Rossmann Store Sales Data

Generated: 7/29/2025, 5:33:33 PM   Total Queries: 2
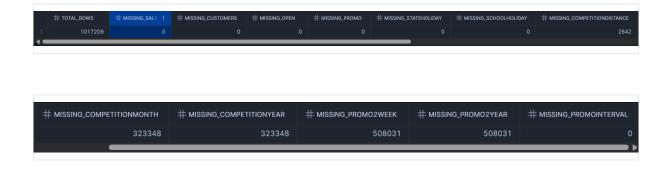
Query #1

Topic: Data Cleaning & Preparation 2.1: Identify NULLs

SQL Query:

```
SELECT
  COUNT(*) AS Total_Rows,
  COUNT(CASE WHEN Sales IS NULL THEN 1 END) AS Missing_Sales,
  COUNT(CASE WHEN Customers IS NULL THEN 1 END) AS
Missing_Customers,
  COUNT(CASE WHEN Open IS NULL THEN 1 END) AS Missing_Open,
  COUNT(CASE WHEN Promo IS NULL THEN 1 END) AS Missing_Promo,
  COUNT(CASE WHEN StateHoliday IS NULL THEN 1 END) AS
Missing_StateHoliday,
  COUNT(CASE WHEN SchoolHoliday IS NULL THEN 1 END) AS
Missing_SchoolHoliday,
  COUNT(CASE WHEN CompetitionDistance IS NULL THEN 1 END) AS
Missing_CompetitionDistance,
  COUNT(CASE WHEN CompetitionOpenSinceMonth IS NULL THEN 1 END) AS
Missing_CompetitionMonth,
  COUNT(CASE WHEN CompetitionOpenSinceYear IS NULL THEN 1 END) AS
Missing_CompetitionYear,
  COUNT(CASE WHEN Promo2SinceWeek IS NULL THEN 1 END) AS
Missing_Promo2Week,
  COUNT(CASE WHEN Promo2SinceYear IS NULL THEN 1 END) AS
Missing_Promo2Year,
  COUNT(CASE WHEN PromoInterval IS NULL THEN 1 END) AS
Missing_PromoInterval
FROM ROSSMANN_FEATURES;
```

Result:

# TOTAL_ROWS	# MISSING_SALE :	# MISSING_CUSTOMERS	# MISSING_OPEN	# MISSING_PROMO	# MISSING_STATEHOLIDAY	# MISSING_SCHOOLHOLIDAY	# MISSING_COMPETITIONDISTANCE
1017209	0	0	0	0	0	0	2642

# MISSING_COMPETITIONMONTH	# MISSING_COMPETITIONYEAR	# MISSING_PROMO2WEEK	# MISSING_PROMO2YEAR	# MISSING_PROMOINTERVAL
323348	323348	508031	508031	0

Query #2

Topic: 2.2: Clean + Convert Categorical Columns

SQL Query:

```
# Step 1: Normalize StateHoliday

UPDATE ROSSMANN_FEATURES
SET StateHoliday = 'None'
WHERE StateHoliday = '0';

# Step 2: Fill PromoInterval NULLs

UPDATE ROSSMANN_FEATURES
SET PromoInterval = 'None'
WHERE PromoInterval IS NULL;

# Step 3: Fill CompetitionDistance NULLs (with median ~2000)

UPDATE ROSSMANN_FEATURES
SET CompetitionDistance = 2000
WHERE CompetitionDistance IS NULL;

#  Step 4: Fill CompetitionOpenSinceMonth and Year

UPDATE ROSSMANN_FEATURES
SET CompetitionOpenSinceMonth = 0
WHERE CompetitionOpenSinceMonth IS NULL;

UPDATE ROSSMANN_FEATURES
SET CompetitionOpenSinceYear = 0
WHERE CompetitionOpenSinceYear IS NULL;


# Step 5: Fill Promo2SinceWeek and Year

UPDATE ROSSMANN_FEATURES
SET Promo2SinceWeek = 0
WHERE Promo2SinceWeek IS NULL;

UPDATE ROSSMANN_FEATURES
SET Promo2SinceYear = 0
```

# number of rows updated	# number of multi-joined rows updated
986159	0

# number of rows updated	# number of multi-joined rows updated
0	0

# number of rows updated	# number of multi-joined rows updated
0	0

# number of rows updated	# number of multi-joined rows updated
323348	0

# number of rows updated	# number of multi-joined rows updated
508031	0