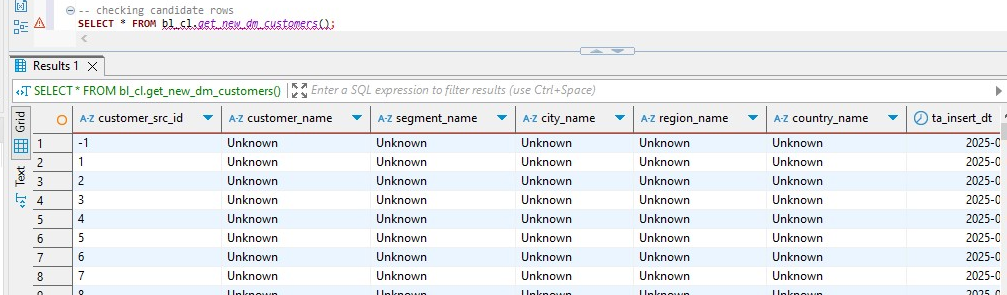
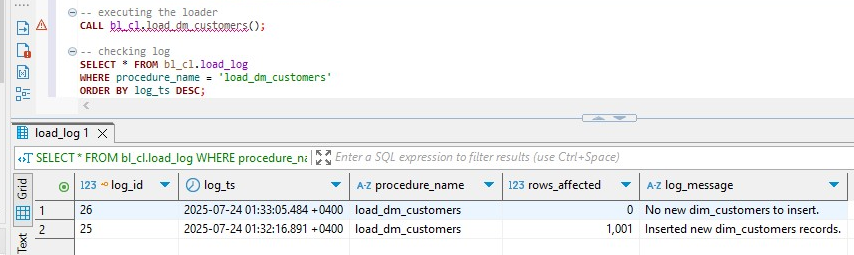
# DM\_CUSTOMERS

Checking candidate rows:

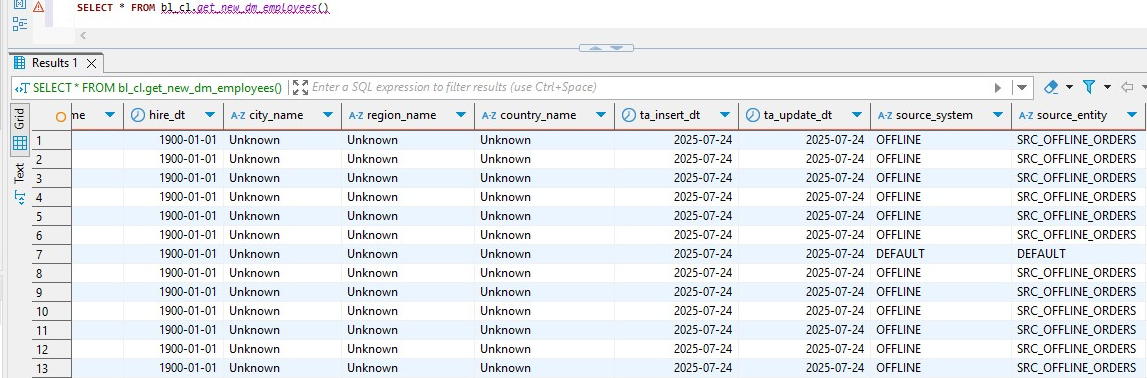


The source files do not include actual customer-level attributes such as customer\_name or segment\_name, so the ETL process populates these fields in the CE\_CUSTOMERS and DIM\_CUSTOMERS tables with the default value 'Unknown'. This approach in data warehousing handles missing data while preserving referential integrity and enabling consistent reporting while avoiding nulls in key descriptive fields. After executing the loader twice, we get that at first it inserted all 1001 candidate rows and at the second time it didn’t do anything because there were no rows left.

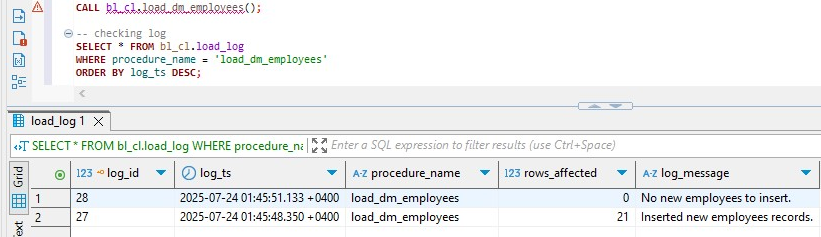


# DM\_EMPLOYEES

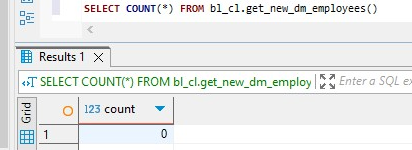
Checking candidate rows:



The same thing applies to this table as it did to DM\_CUSTOMERS. I just wanted to showcase that every employee here is inserted from the offline source, because both of the sources refer to the same set of employees so deduplication logic worked correctly and filtered them out. After executing the loader twice, we get that at first all 21 rows were inserted and at the second call there were no rows left to be inserted.

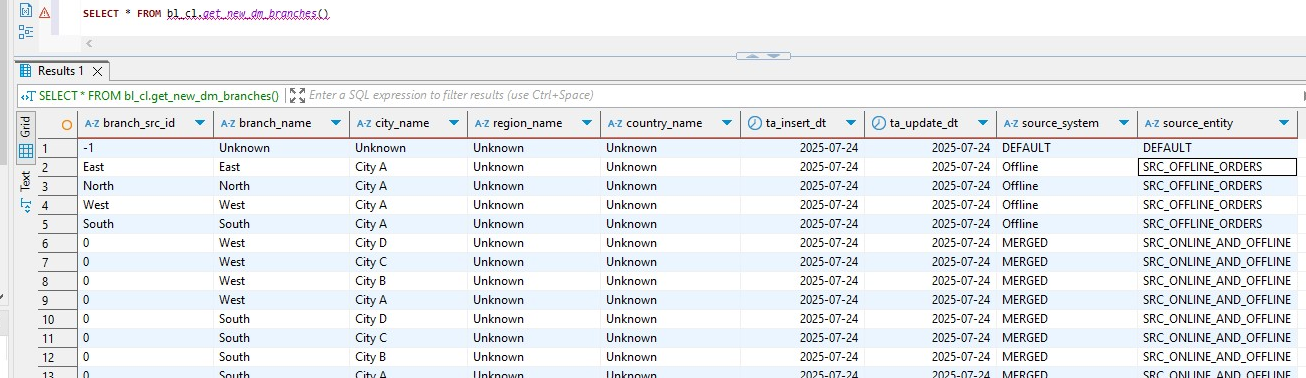


To reckeck this:

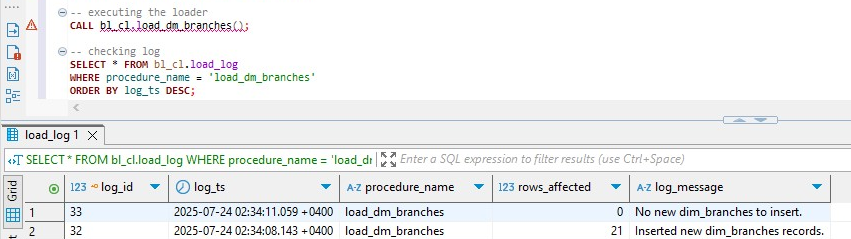


# DIM\_BRANCHES

Checking candidate rows:



The data reflects a mix of default, offline, and merged source records as handled during the 3NF ETL process. The row with branch\_src\_id = -1 represents a default fallback branch used to ensure referential integrity when branch data is missing. Branches such as "East," "North," and "West" originate from the offline source, while rows marked with source\_system = 'MERGED' indicate that the same branch appeared in both online and offline sources and was deduplicated during loading. Some entries with branch\_src\_id = 0 result from missing or malformed source values which were replaced with a placeholder during transformation. Additionally, many rows have 'Unknown' in region\_name and country\_name due to incomplete address data in the associated ce\_addresses table. After executing the loader twice, we get that at first all 21 rows were inserted and at the second call there were no rows left to be inserted.

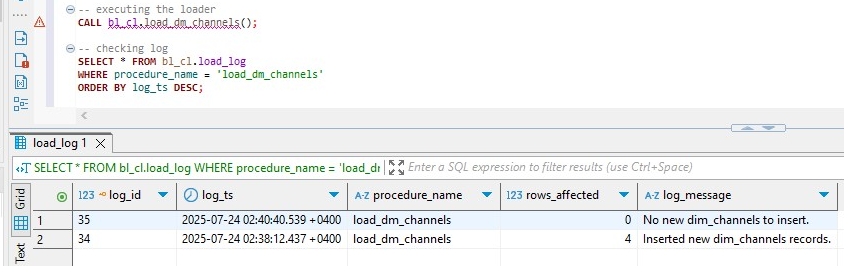


To recheck it

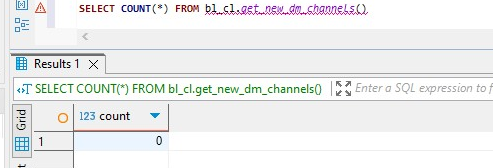


# DIM\_CHANNELS

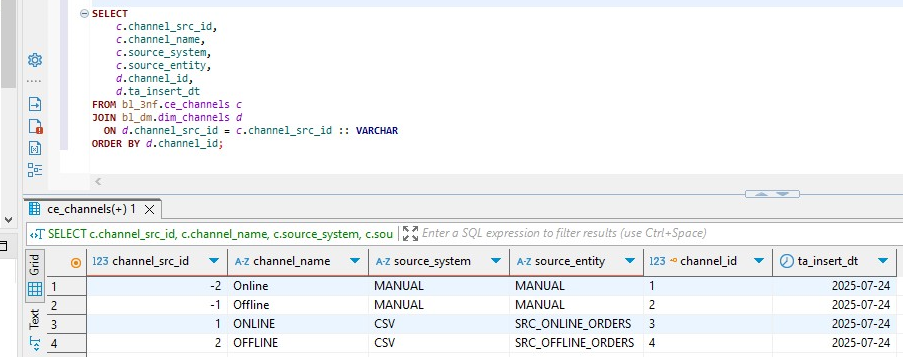
I accidentally executed loader before checking candidate rows, so after executing it two times we get that 4 channels were inserted at first time and none at the second time.



So the new count of candidate rows is 0



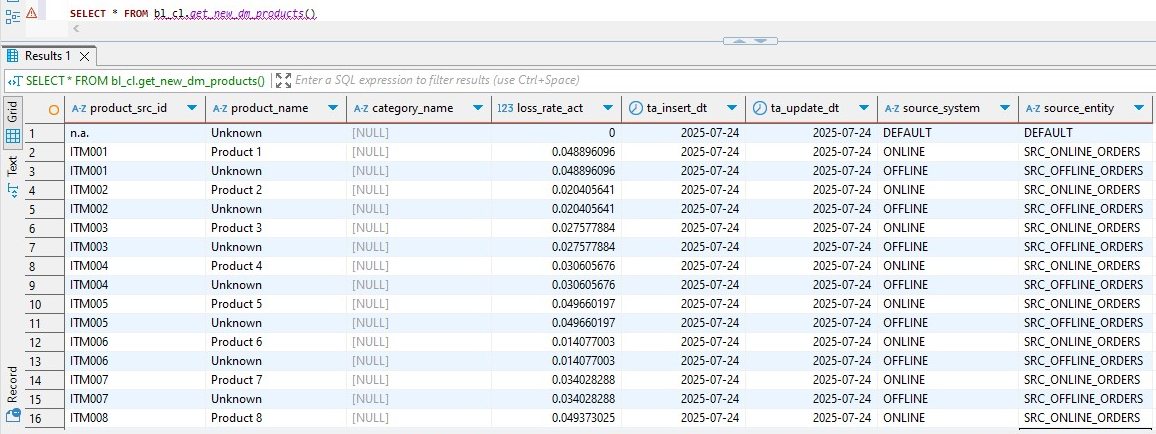
To see which records were inserted



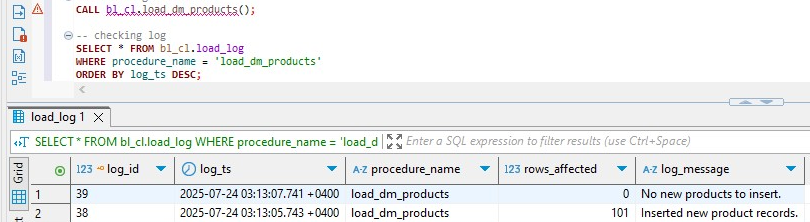
The reason you see two different entries for both Online and Offline channels — one with source\_system = MANUAL and another with source\_system = CSV is because we have fallback rows that are **manually inserted default records** used when the source data is missing. On the other hand, the rows with channel\_src\_id = ONLINE and OFFLINE and source\_system = CSV come directly from the actual raw data.

# DIM\_PRODUCTS

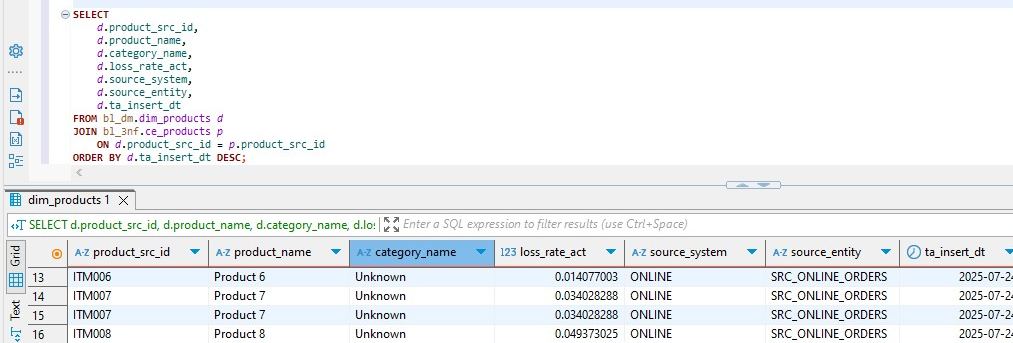
Checking candidate rows:



The column category\_name wasn’t provided in the source so its null everywhere. I couldn’t execute loader because of the NOT NULL CONSTRAINT, so I used COALESCE and changed nulls to uknown everywhere. I couldn’t take screenshot of it so I will just show rows that were inserted after it. After executing the loader twice, we get that at first all 101 rows were inserted and at the second call there were no rows left to be inserted.

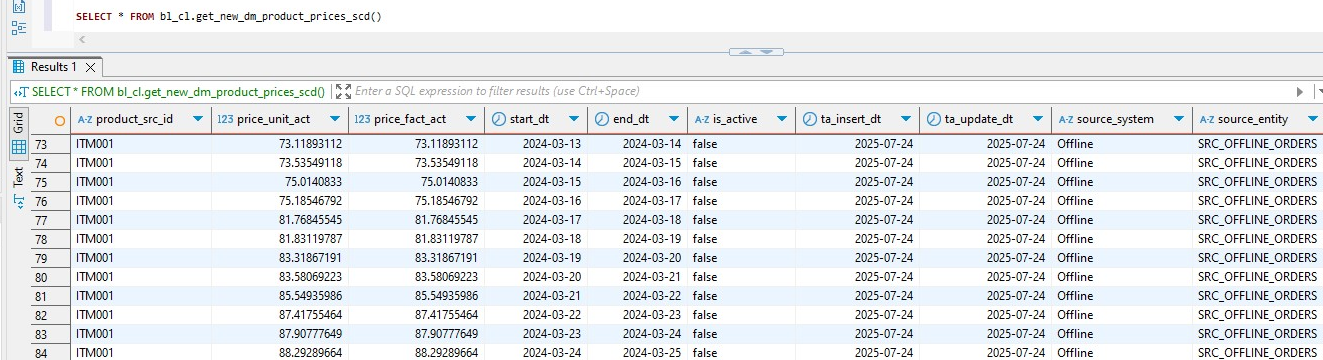


And these are the rows that were inserted



# PRODUCT\_PRICES\_SCD

Checking candidate rows:



These result clearly demonstrates that SCD Type 2 is applied to it correctly, each price for its source entity and item code has its own starting date and ending date. The ending date of each each price is the beginning of the new one. Is active column is false for each price except the last one for each item. After executing the loader twice, we get that at first all 10000 rows were inserted and at the second call there were no rows left to be inserted.

