Summary

1. Request hourly and daily historical weather in 2016 from World Weather Online API by using Python
2. Extract hourly and daily temperature and date from the raw data and formulate date and time
3. Write formulated data into CSV file as dailyweather.csv and hourlyweather.csv
4. Import sales and weather file into database

Report 1

1. Create a temporary table CTE1: Left join morse and hourlyweather table on record date and time, then each record in morse table will have corresponding temperature. Group this table by hourly temp and item, so each temperature will have sum of sales for each item
2. Create a temporary table CTE2: based on CTE1, use Dense\_rank() to obtain the sales rank of each item under each temperature group.
3. Based on CTE2, select temperature, items, sales from CTE2 with the top sales rank only. Then, for each temperature, it will show the item with top sales and corresponding sales

Report2

1. Create a temporary table CTE1: self join morse table to make each date have all items, then left join dailyweather table on date, so each date will display its temperature and all items
2. Create a temporary table CTE2: Group morse table by record date and item, so each date will have sum of sales for each item
3. Create a temporary table CTE3: Based on CTE1 and CTE2, left join CTE1 and CTE2 on record date and item, so for each record will have date, item, sum of sales, and daily temperature.
4. Create a temporary table CTE4: Based on CTE3, self join CTE3 on item and date difference of record time at 1 to obtain now sales and next sales. Then, for each record, use case statement to determine if next day temperature is higher or lower than **2 degrees Fahrenheit or more** than today temperature. Also, for each record, take the differences of sales sum between today and next day. Thus, For each record, it will have date, item, warmer or colder,

**Assumption NOTE**: 1. when take differences, if a sales record of a specific day does not have previous day sales record, this sales record will be considered as invalid. I do not assume this day has 0 sales.

2. For report requirements, it states warmer or colder by 2 degrees. To clarify, I assume it includes 2 degrees and more than 2 degrees since exactly 2 degrees would be less sense for temperature changes.

1. Based on CTE4: Select Item from CTE4, then use case function to find sales differences with warmer and colder. Group the table by item to obtain the average of the sales change when warmer and colder.