Walmart Weekly Sales:  
PostgreSQL ETL Documentation

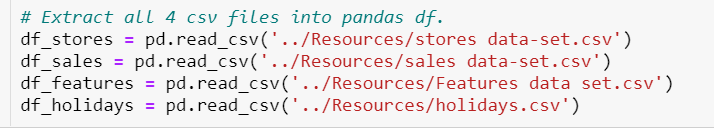
Data Sets: Data is extracted from the Kaggle Walmart Retail Analytics Competition as CSV files. A total of 3 data sets; sales data-set.csv, Features data set.csv, and stores data-set.csv. An additional CSV file, holidays.csv, was created in Excel to help identify which holiday fell on a given date.

ETL Objectives:

* Create new PostgreSQL database in PGAdmin called retail\_db, with Schema for 4 tables developed in Quick Database Diagrams.
* Extract all 4 CSV files to a different Pandas Dataframe in Jupyter Notebooks.
* Transform each Dataframe to match to Schema for PostgreSQL retail\_db database.
* Load transformed Dataframes to PostgreSQL retail\_db, via Pandas to\_sql function and SQLAlchemy in Jupyter Notebook.

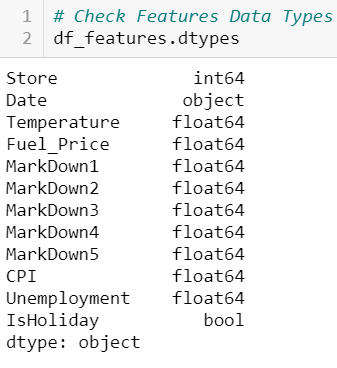
# Data Extraction

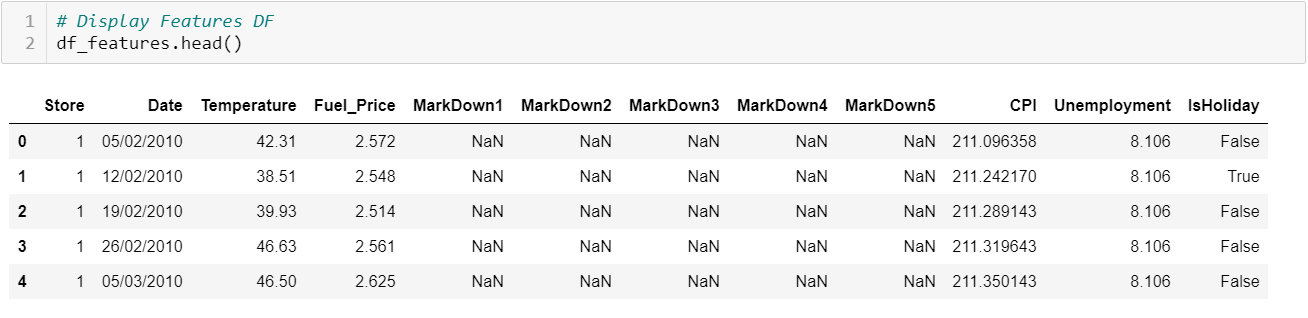
* Jupyter Notebook created for Project.
* Dependencies:
  + Pandas
    - Dataframe Functions
  + Datetime
    - Formatting Dates
  + SQLAlchemy (create\_engine)
    - Load Dataframes into
  + Config (File)
    - Contains PGAdmin Password
* Extracted All 4 CSV files via Pandas read\_csv function



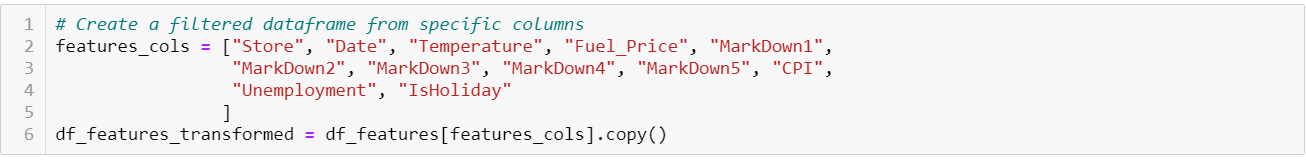
# Features data set.csv Transformation

* Check Data Types & Display Dataframe Pre-Transformation:





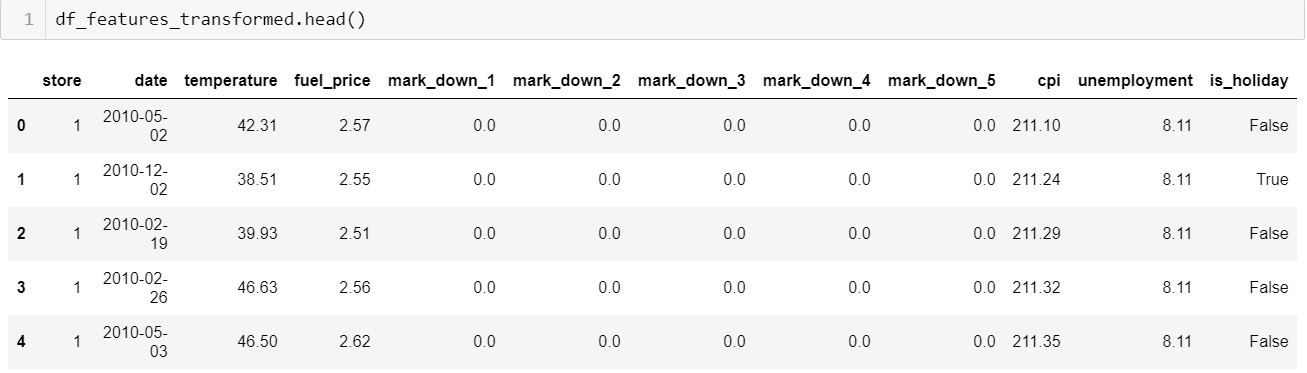
* Select all columns and copy Dataframe.

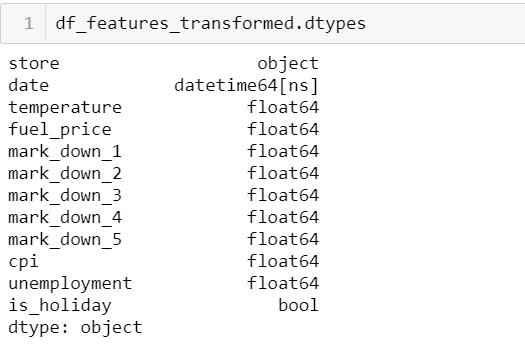


* Convert Date to Datetime: yyyy/mm/dd
* Convert Column “Store” to string
* Round all Float Data Types to 2 decimal places
* Replace all NaN values to 0.0
* Rename columns to include underscores for spacing.
* Map and lowercase all column headers to ensure column headers match PostgreSQL Schema



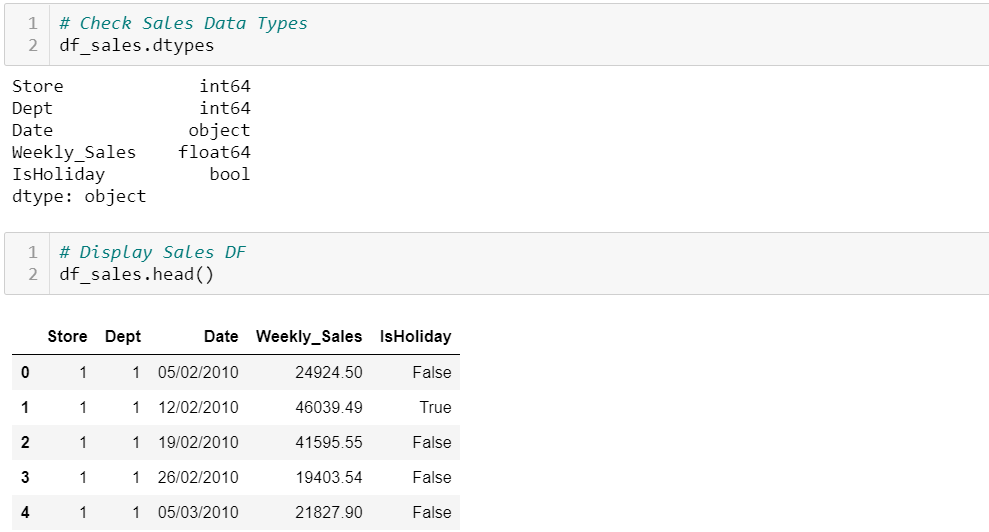
* Display Transformed Dataframe and Data Types



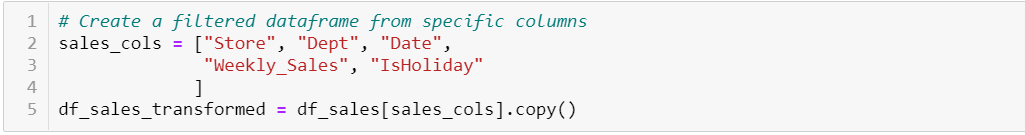


# sales data-set.csv Transformation

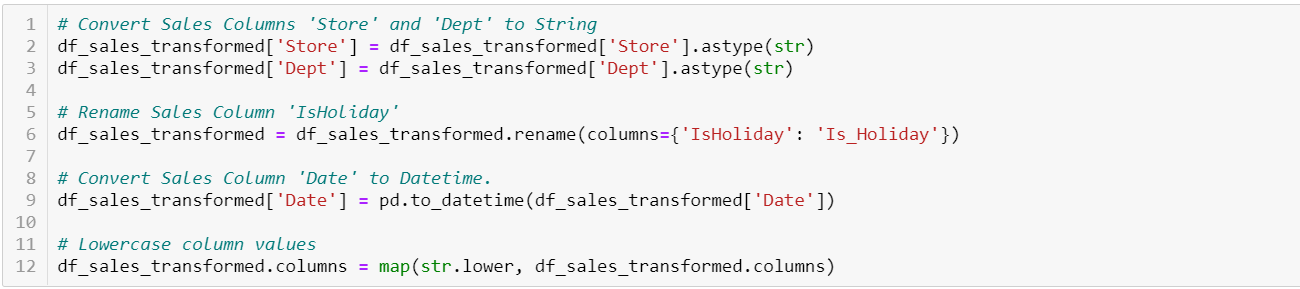
* Check Data Types & Display Dataframe Pre-Transformation:



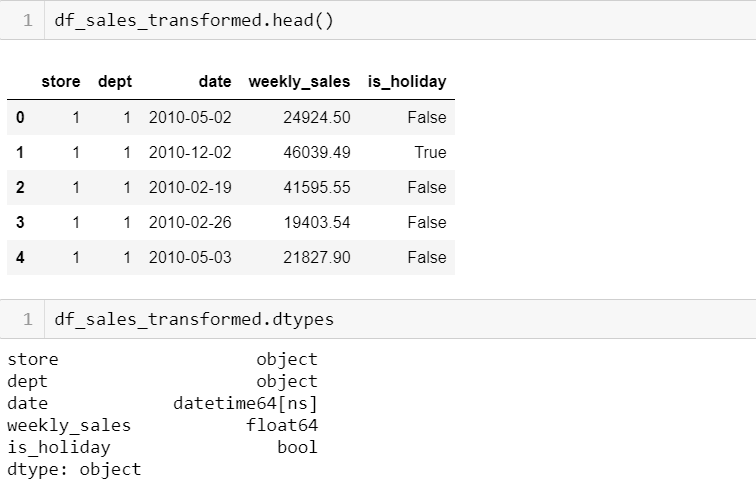
* Select all columns and copy Dataframe.



* Convert Column “Store” and “Dept” to String
* Rename Column “IsHoliday” with underscore
* Convert Column “Date” to Datetime: yyyy/mm/dd
* Map and lowercase all column headers to ensure column headers match PostgreSQL Schema

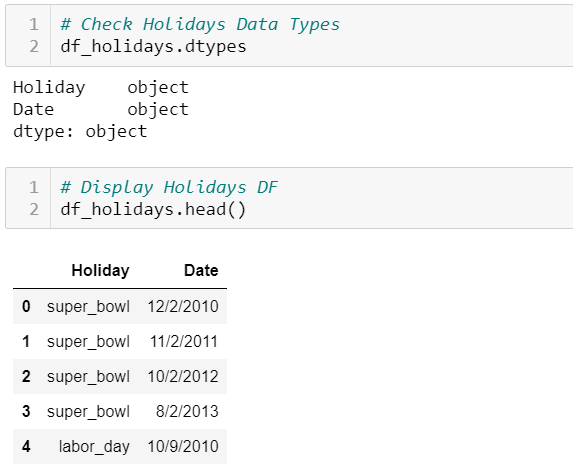


* Display Transformed Dataframe and Data Types

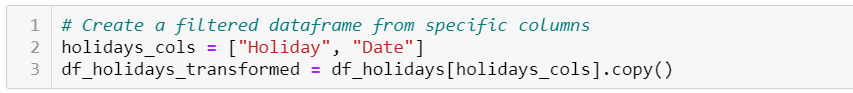


# holidays.csv Transformation

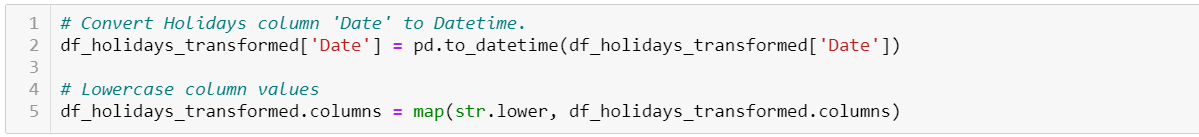
* Check Data Types & Display Dataframe Pre-Transformation:



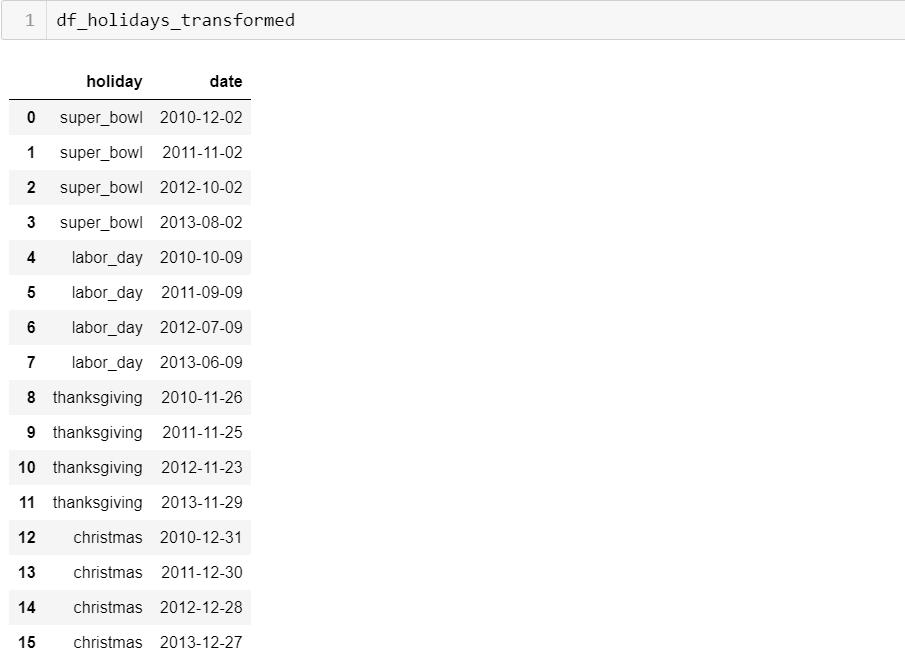
* Select all columns and copy Dataframe.



* Convert Column “Date” to Datetime: yyyy/mm/dd
* Map and lowercase all column headers to ensure column headers match PostgreSQL Schema



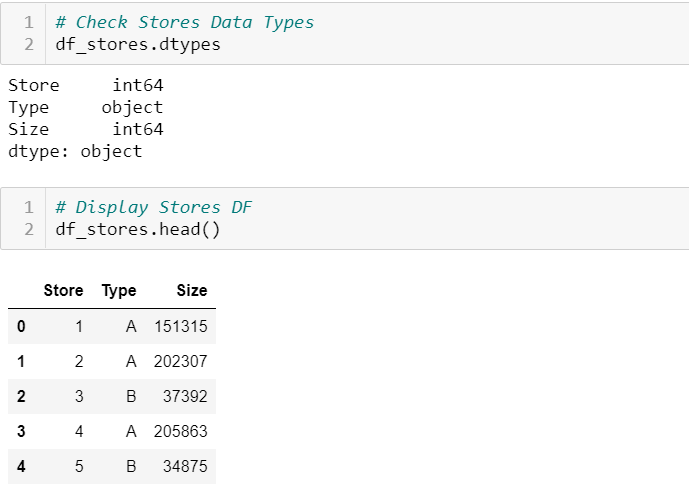
* Display Transformed Dataframe and Data Types



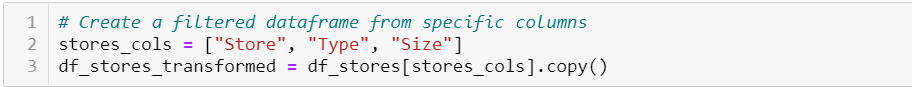


# Stores data-set.csv Transformation

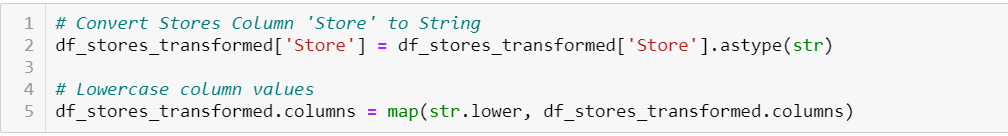
* Check Data Types & Display Dataframe Pre-Transformation:



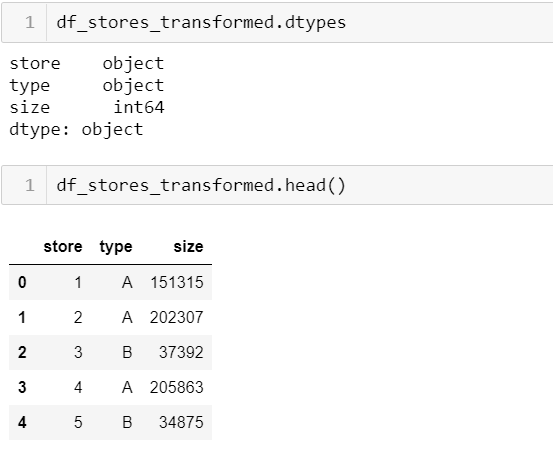
* Select all columns and copy Dataframe.



* Convert Column “Store” to String
* Map and lowercase all column headers to ensure column headers match PostgreSQL Schema

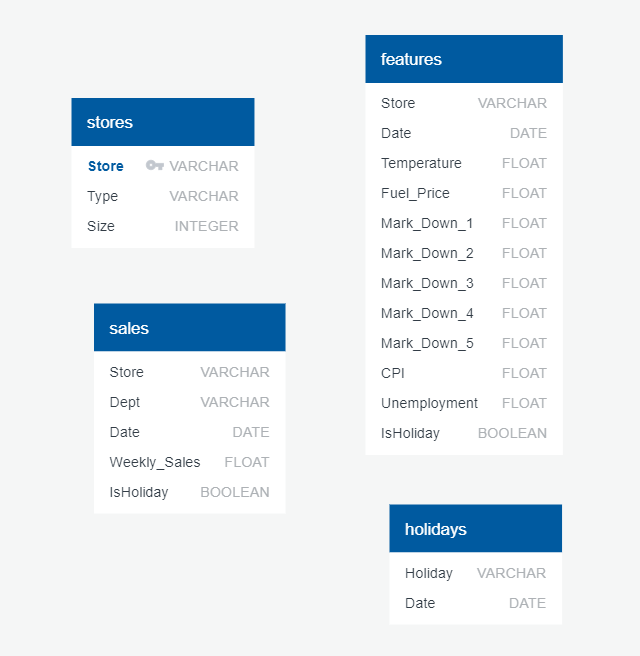


* Display Transformed Dataframe and Data Types

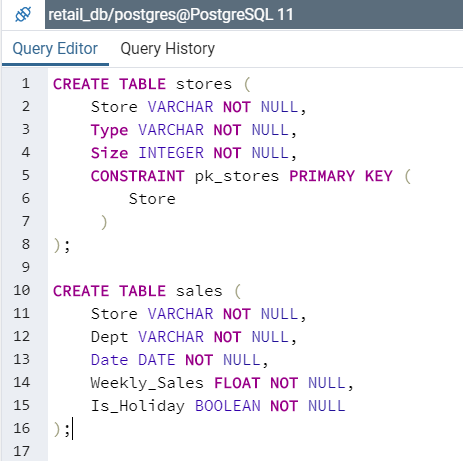
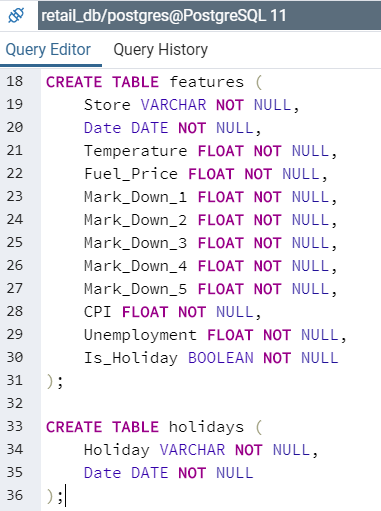


# PostgreSQL Database & Schema

* Quick Database Diagrams was used to create retail\_db Schema.



* Create Table Statements:

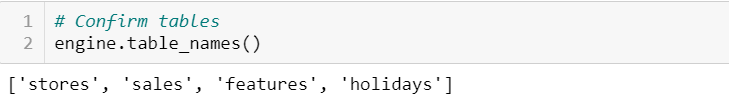
 

Load to PostgreSQL Database

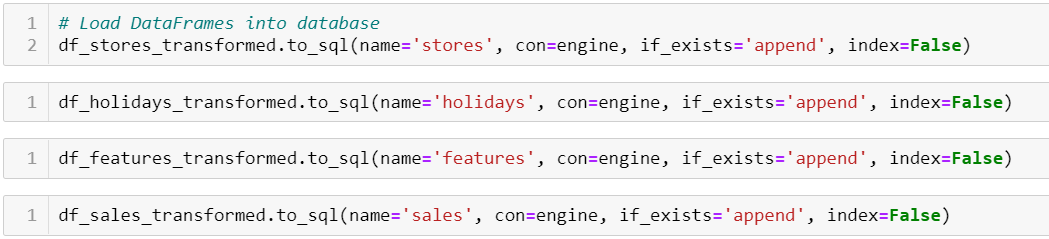
* Create Database Connection to PostgreSQL retail\_db: Config file contains Password



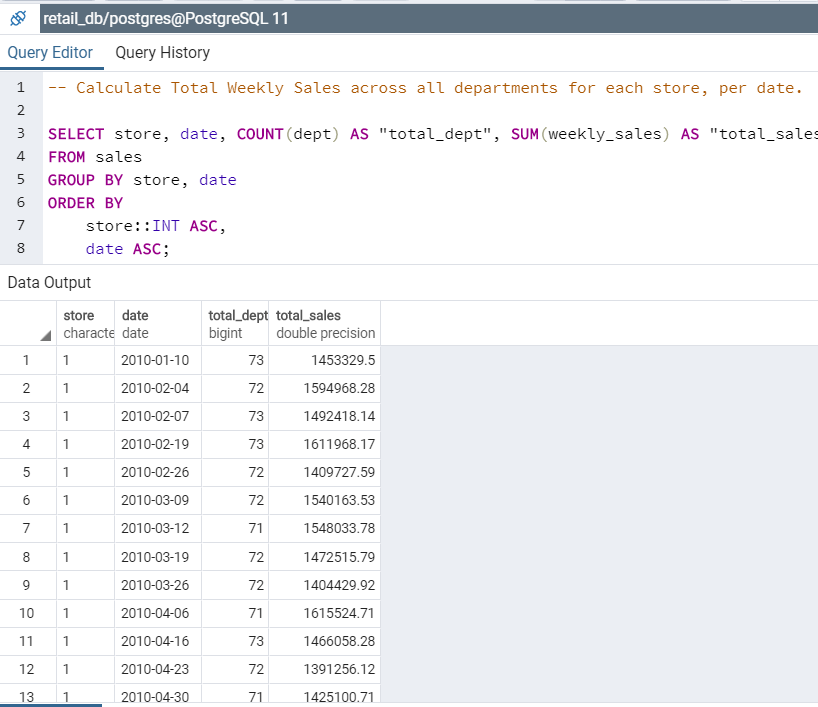
* Confirm available tables in PostgreSQL retail\_db



* Load Transformed Dataframes into PostgreSQL retail\_db



# Total Weekly Sales Per Store SQL Queries



# Total Weekly Sales Per Store SQL Queries – with Sales and Features Tables Join

