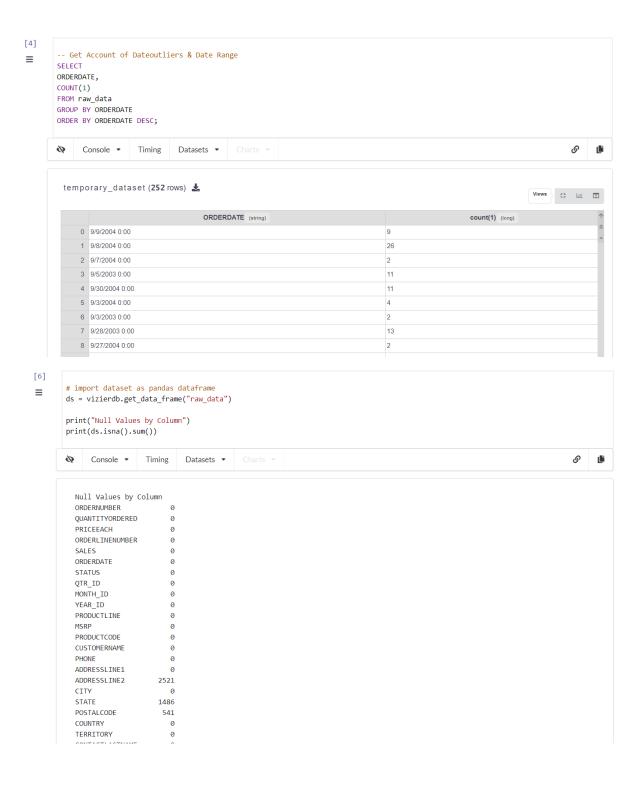
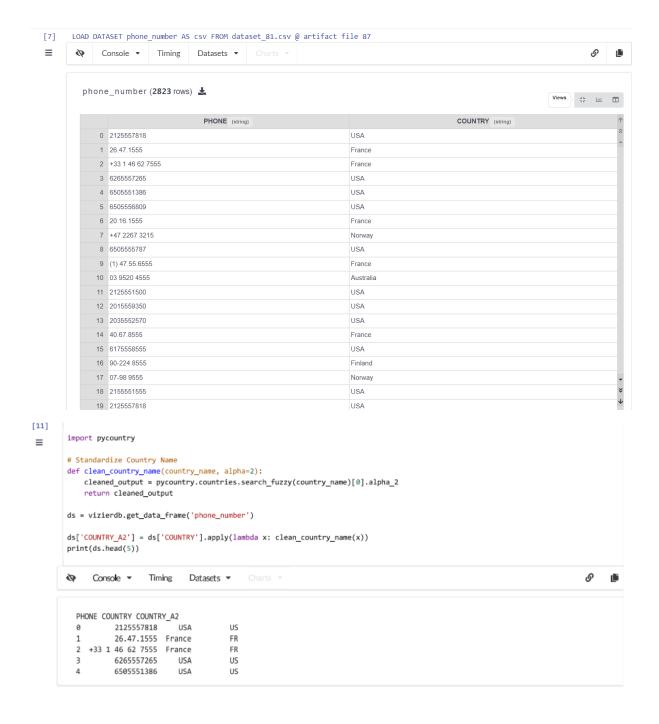


- The raw sales dataset consist of Order Details places for given products.
- The dataset consist of 25 Order Attributes and 2823 Row Orders
- The datatypes in the dataset includes INT, FLOAT, STRING, STRING, DATETIME

Column Name	Data Type	Short Description	Notes
ORDERNUMBER	int	Order Identifier	
QUANTITYORDERED	int	# of Units ordered	
PRICEEACH	float	Unit Price of Each	This can be pushed to Product Table, and references here
ORDERLINENUMBER	int	?	Can be dropped
SALES	float	Sale Price of the order	
ORDERDATE	datetime	Date on which order was placed	Remove time component
STATUS	string	Status of the order	
QTR_ID	int	Quarter Number	Can be dropped since we can derive it from Order Date
MONTH_ID	int	Month Number	Can be dropped since we can derive it from Order Date
YEAR_ID	int	Year Number	Can be dropped since we can derive it from Order Date
PRODUCTCODE	string	Product Identifier	Create Product Table using this ID
CUSTOMERNAME	string	Name of the Company	Create Customer Table
PHONE	string	Customer Phone Number	Push to Customer Table
ADDRESSLINE1	string	Customer Physical Address	Push to Customer Table
ADDRESSLINE2	string	Customer Physical Address Extended	Push to Customer Table
CITY	string	Name of the City	Standardize
STATE	string	Name of the State	Standardize, Default for Country with no State
POSTALCODE	string	ZipCode	Standardize
COUNTRY	string	Name of the country	
TERRITORY	string	Name of the territory	
CONTACTLASTNAME	string	Representative Last name of Custome	r Push to Customer Table
CONTACTFIRSTNAME	string	Representative First name of Custome	r Push to Customer Table
DEALSIZE	string	size of order	Can be populated using automatic function





```
[12]
        import phonenumbers
\equiv
        import pycountry
        def clean_phone_numbers(phone_number, country):
              ''Reformat's passed argument with corrected phone number format based on the country.'''
            formated_phone = phonenumbers.parse(phone_number, country)
            formated_phone = phonenumbers.format_number(formated_phone, phonenumbers.PhoneNumberFormat.INTERNATIONAL)
            return formated_phone
        \verb| \#phone numbers.format\_number(phone numbers.parse("8006397663", 'US'), phone numbers.Phone NumberFormat.NATIONAL)|
        # get dataframe
        ds = vizierdb.get_dataset('phone_number')
#print(phonenumbers.parse("8006397663", 'US'))
        print("|PREV FORMAT| NEW FORMAT| COUNTRY")
        for row in ds.rows[:10]:
           phone = re.sub("[^0-9]","",row[0])
            country = pycountry.countries.search_fuzzy(row[1])[0].alpha_2
            print(phone,clean_phone_numbers(phone, country), country)

    Console ▼ Timing Datasets ▼ Charts ▼
                                                                                                                                      Ø 🍱
          |PREV FORMAT| NEW FORMAT| COUNTRY
2125557818 +1 212-555-7818 US
          26471555 +33 26471555 FR
          33146627555 +33 1 46 62 75 55 FR
          6265557265 +1 626-555-7265 US
          6505551386 +1 650-555-1386 US
          6505556809 +1 650-555-6809 US
          20161555 +33 20161555 FR
          4722673215 +47 22 67 32 15 NO
          6505555787 +1 650-555-5787 US
          147556555 +33 1 47 55 65 55 FR
       SELECT PRODUCTCODE,
\equiv
        LAST(PRODUCTLINE) AS PRODUCTLINE,
        MAX(PRICEEACH) AS MAX_UNIT_PRICE,
        MIN(PRICEEACH) AS MIN_UNIT_PRICE,
        MAX(MSRP) AS MAX_RETAIL_PRICE
        -- MIN(MSRP) AS MIN_RETAIL_PRICE MSRP IS SAME
        FROM raw_data
        GROUP BY PRODUCTCODE

    Console ▼ Timing Datasets ▼ Charts ▼
                                                                                                                                        6
      SELECT DISTINCT
      CUSTOMERNAME,
      CONTACTFIRSTNAME,
      CONTACTLASTNAME,
      PHONE,
      ADDRESSLINE1,
      ADDRESSLINE2,
      CITY,
      STATE,
      POSTALCODE,
      COUNTRY
      FROM raw_data
      ORDER BY CUSTOMERNAME

    Console ▼ Timing Datasets ▼ Charts ▼
                                                                                                                                           Ø 1
```