Project 1

October 27, 2023

[ ]:

```
#conda install psycopg2
```

[4]:

```
import psycopg2
import pandas as pd

# Connection parameters
host = "pg.analytics.northwestern.edu"
port = "5432"
database = "everything2023"
user = ""
password = ""

# Establish a connection to the database
conn = psycopg2.connect(
    host=host,
    port=port,
    database=database,
    user=user,
    password=password
)
```

[5]:

```
cursor = conn.cursor()
sql_query = "SELECT * FROM group_13.deptinfo;"
cursor.execute(sql_query)
deptinfo = pd.read_sql_query(sql_query, conn)

cursor = conn.cursor()
sql_query2 = "SELECT * FROM group_13.trnsact LIMIT 100000;"
cursor.execute(sql_query2)
trnsact = pd.read_sql_query(sql_query2, conn)

cursor = conn.cursor()
sql_query3 = "SELECT * FROM group_13.skstinfo LIMIT 10000;"
cursor.execute(sql_query3)
skstinfo = pd.read_sql_query(sql_query3, conn)
skstinfo.head()

cursor = conn.cursor()
```

```
sql_query4 = "SELECT * FROM group_13.strinfo;"
cursor.execute(sql_query4)
strinfo = pd.read_sql_query(sql_query4, conn)

cursor = conn.cursor()
sql_query5 = "SELECT * FROM group_13.skuinfo;"
cursor.execute(sql_query5)
skuinfo = pd.read_sql_query(sql_query5, conn)

conn.close()
```

[6]:
```
skstinfo.head()
```

[6]:
```
       SKU  STORE  COST  RETAIL  unknown
0  5589931    102  16.5    39.0        0
1  5589931    107  16.5    39.0        0
2  5589931    204  16.5    39.0        0
3  5589931    302  16.5    39.0        0
4  5589931    304  16.5    39.0        0
```

[7]:
```
skuinfo.head()
```

[7]:
```
Empty DataFrame
Columns: [3, 6505, 113, 000400000003000, 00    F55KT2, whisperwhite, P8EA      ,
1, 5119207, TURNBURY , 0]
Index: []
```

[8]:
```
deptinfo.head()
```

[8]:
```
   DEPT  DEPTDESC  Unknow
0   800  CLINIQUE       0
1   801  LESLIE         0
2  1100  GARY F         0
3  1107  JACQUES        0
4  1202  CABERN         0
```

[9]:
```
trnsact.head()
```

[9]:
```
     SKU  STORE  REGISTER  TRANNUM        SEQ    SALEDATE STYPE  QUANTITY  \
0  21717   4404       560     3900          0  2004-12-14     P         1
1  21717   4404       560     4300  797207489  2004-12-03     P         1
2  21717   4404       560     4800          0  2004-11-27     P         1
3  21717   4404       570      600          0  2004-08-05     P         1
4  21717   4407       130      300  308802737  2004-08-02     P         1

   ORGPRICE  SPRICE   AMT    INTERID  MIC  Unknow
0      20.0    20.0  20.0  970000021  230       0
1      20.0    20.0  20.0  271100020  230       0
```

```
2      20.0    20.0  20.0  972200022  230       0
3      20.0    20.0  20.0  953100022  230       0
4      20.0    20.0  20.0  245900018  230       0
```

```
[10]: strinfo.head()
```

```
[10]:    store                  city state    zip  x
      0      2  ST. PETERSBURG          FL  33710  0
      1      3  ST. LOUIS               MO  63126  0
      2      4  LITTLE ROCK             AR  72201  0
      3      7  FORT WORTH              TX  76137  0
      4      9  TEMPE                   AZ  85281  0
```

0.1 Clean Data

```
[11]: # Drop unknow column (the last column):
      deptinfo.drop(columns=["Unknow"],inplace=True)
      deptinfo.head()
```

```
[11]:    DEPT  DEPTDESC
      0   800  CLINIQUE
      1   801  LESLIE
      2  1100  GARY F
      3  1107  JACQUES
      4  1202  CABERN
```

```
[12]: # Drop the last unknown column:
      trnsact.drop(columns=["Unknow"],inplace=True)
      trnsact
```

```
[12]:          SKU  STORE  REGISTER  TRANNUM        SEQ    SALEDATE STYPE  QUANTITY  \
      0      21717   4404       560     3900          0  2004-12-14     P         1
      1      21717   4404       560     4300  797207489  2004-12-03     P         1
      2      21717   4404       560     4800          0  2004-11-27     P         1
      3      21717   4404       570      600          0  2004-08-05     P         1
      4      21717   4407       130      300  308802737  2004-08-02     P         1
      …      …    …       …      …        …         …   …       …
      99995  29633    903       410     1900          0  2005-02-23     P         1
      99996  29633    903       410     2100  562109125  2005-06-05     P         1
      99997  29633    903       410     2100          0  2005-07-02     P         1
      99998  29633    903       410     2200  837605210  2005-02-05     P         1
      99999  29633    903       410     2200  837605210  2005-07-23     P         1

             ORGPRICE  SPRICE   AMT    INTERID  MIC
      0          20.0    20.0  20.0  970000021  230
      1          20.0    20.0  20.0  271100020  230
      2          20.0    20.0  20.0  972200022  230
```

```
3           20.0    20.0  20.0  953100022  230
4           20.0    20.0  20.0  245900018  230
…          …     …    …         …   …
99995       21.0    21.0  21.0   84400297  281
99996       21.0    21.0  21.0  839200055  281
99997       21.0    21.0  21.0  157200142  281
99998       21.0    21.0  21.0  972500193  281
99999       22.5    22.5  22.5  410100129  281

[100000 rows x 13 columns]
```

[13]:
```
# Drop the last unknown column:
skstinfo.drop(columns=["unknown"],inplace=True)
skstinfo
```

[13]:
```
          SKU  STORE   COST  RETAIL
0     5589931    102  16.50    39.0
1     5589931    107  16.50    39.0
2     5589931    204  16.50    39.0
3     5589931    302  16.50    39.0
4     5589931    304  16.50    39.0
…         …     …     …      …
9995  5591189   5304  31.15    89.0
9996  5591189   5503  31.15    89.0
9997  5591189   5602  31.15    89.0
9998  5591189   5604  31.15    89.0
9999  5591189   5704  31.15    89.0

[10000 rows x 4 columns]
```

[14]:
```
# Drop the last unknown column:
strinfo.drop(columns=['x'],inplace=True)
strinfo
```

[14]:
```
     store                  city state   zip
0        2  ST. PETERSBURG          FL  33710
1        3  ST. LOUIS               MO  63126
2        4  LITTLE ROCK             AR  72201
3        7  FORT WORTH              TX  76137
4        9  TEMPE                   AZ  85281
..     …                       …    …     …
448   9808  GILBERT                 AZ  85233
449   9812  METAIRIE                LA  70006
450   9900  LITTLE ROCK             AR  72201
451   9906  LITTLE ROCK             AR  72201
452   9909  CHEYENNE                WY  82009
```

```
[453 rows x 4 columns]
```

```
[15]: strinfo.columns = ['STORE','CITY','STATE','ZIP']
strinfo
```

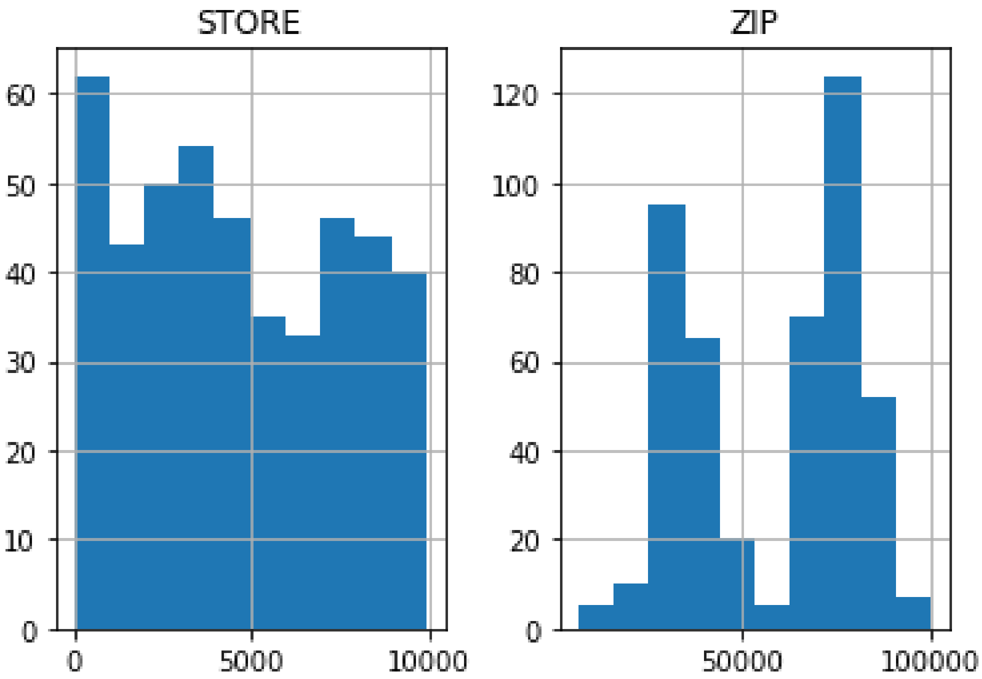
```
[15]:      STORE                 CITY STATE    ZIP
0        2  ST. PETERSBURG          FL   33710
1        3  ST. LOUIS               MO   63126
2        4  LITTLE ROCK             AR   72201
3        7  FORT WORTH              TX   76137
4        9  TEMPE                   AZ   85281
..     …                  …    …     …
448   9808  GILBERT                 AZ   85233
449   9812  METAIRIE                LA   70006
450   9900  LITTLE ROCK             AR   72201
451   9906  LITTLE ROCK             AR   72201
452   9909  CHEYENNE                WY   82009

[453 rows x 4 columns]
```

```
[16]: # merge_table = pd.merge(strinfo, skstinfo, on='STORE', how='inner')
# #merge_table = pd.merge(merge_table, skuinfo, on='SKU', how='inner')
# merge_table = pd.merge(merge_table, deptinfo, on='DEPT', how='inner')
# merge_table = pd.merge(merge_table, trnsaact, on='SKU', how='inner')
# merge_table
```

0.1.1 strinfo:

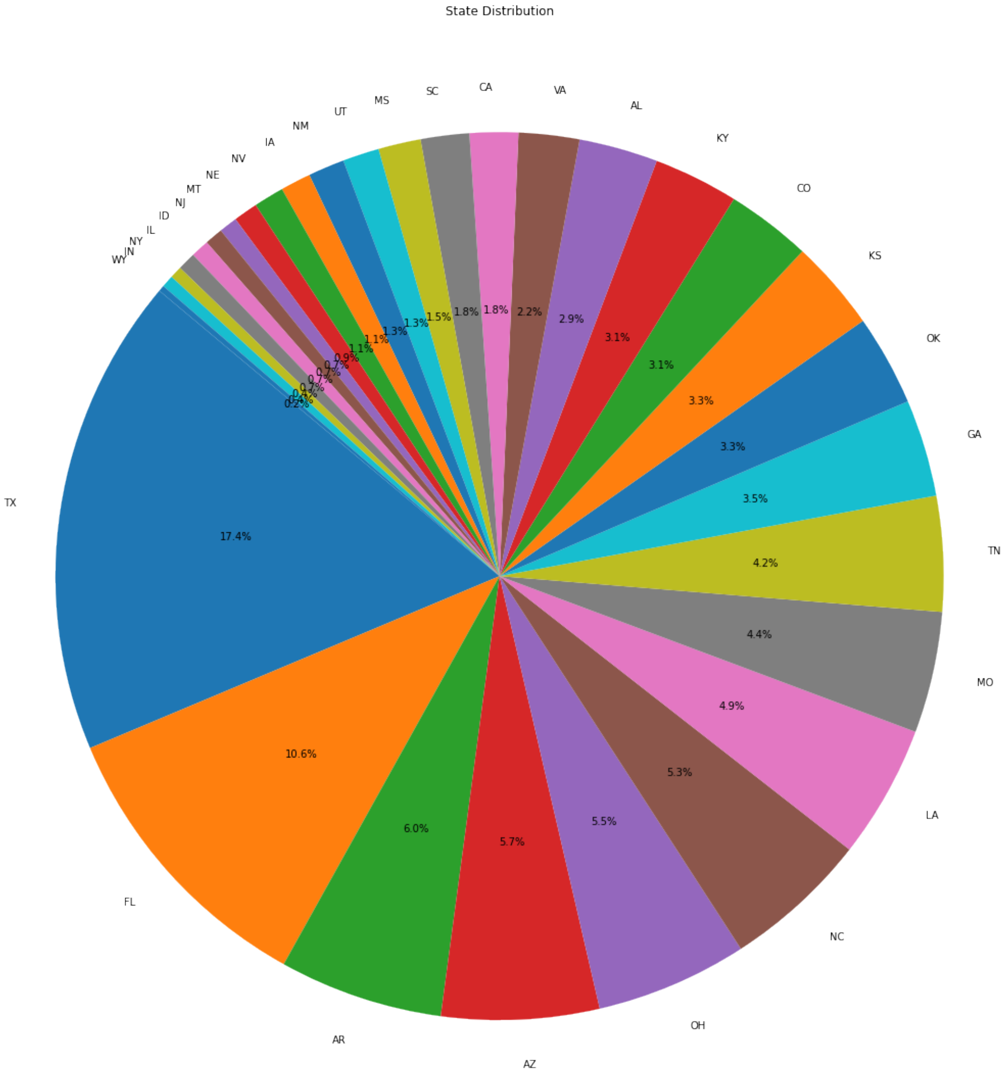
```
[17]: strinfo.hist()
```

```
[17]: array([[<AxesSubplot:title={'center':'STORE'}>,
        <AxesSubplot:title={'center':'ZIP'}>]], dtype=object)
```



```
[18]: import matplotlib.pyplot as plt

      state_counts = strinfo['STATE'].value_counts()
      plt.figure(figsize=(20, 20))  # Optional: Set the figure size
      plt.pie(state_counts, labels=state_counts.index, autopct='%1.1f%%',␣
       ↪startangle=140)
      plt.title('State Distribution')
```

```
[18]: Text(0.5, 1.0, 'State Distribution')
```



0.1.2 trnsact:

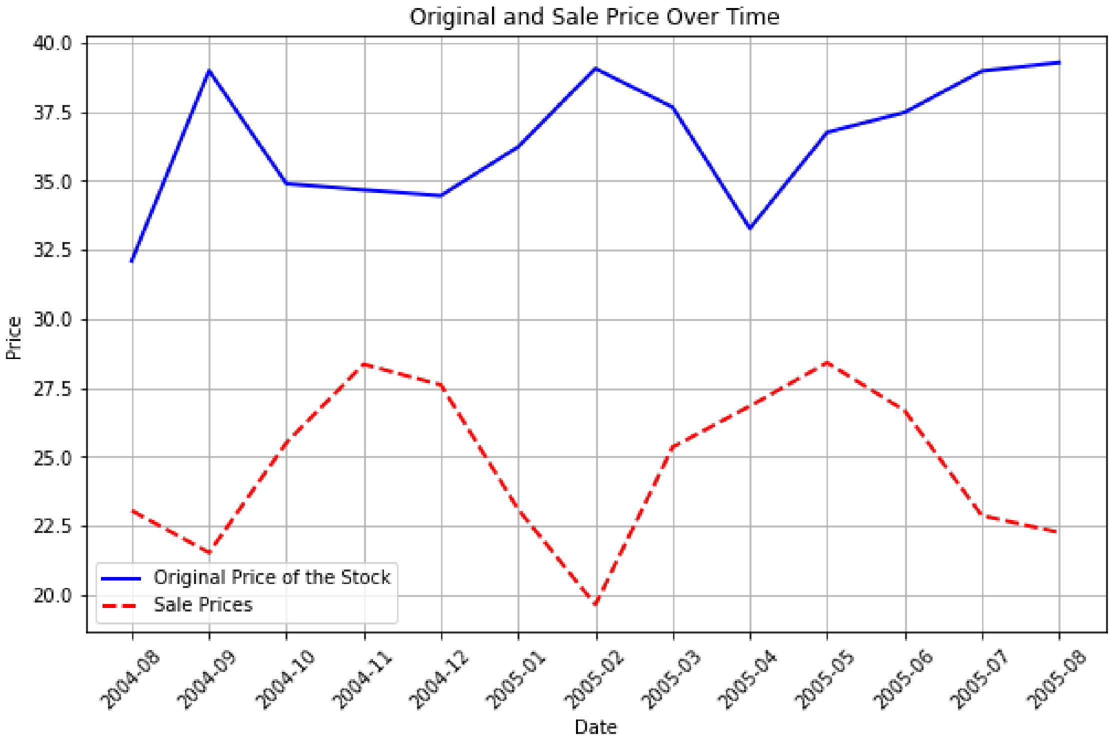
```
[19]: trnsact.dtypes
```

```
[19]: SKU           int64
      STORE         int64
      REGISTER      int64
      TRANNUM       int64
      SEQ           int64
      SALEDATE     object
      STYPE        object
      QUANTITY      int64
```

```
ORGPRICE    float64
SPRICE      float64
AMT         float64
INTERID       int64
MIC           int64
dtype: object
```

[20]:

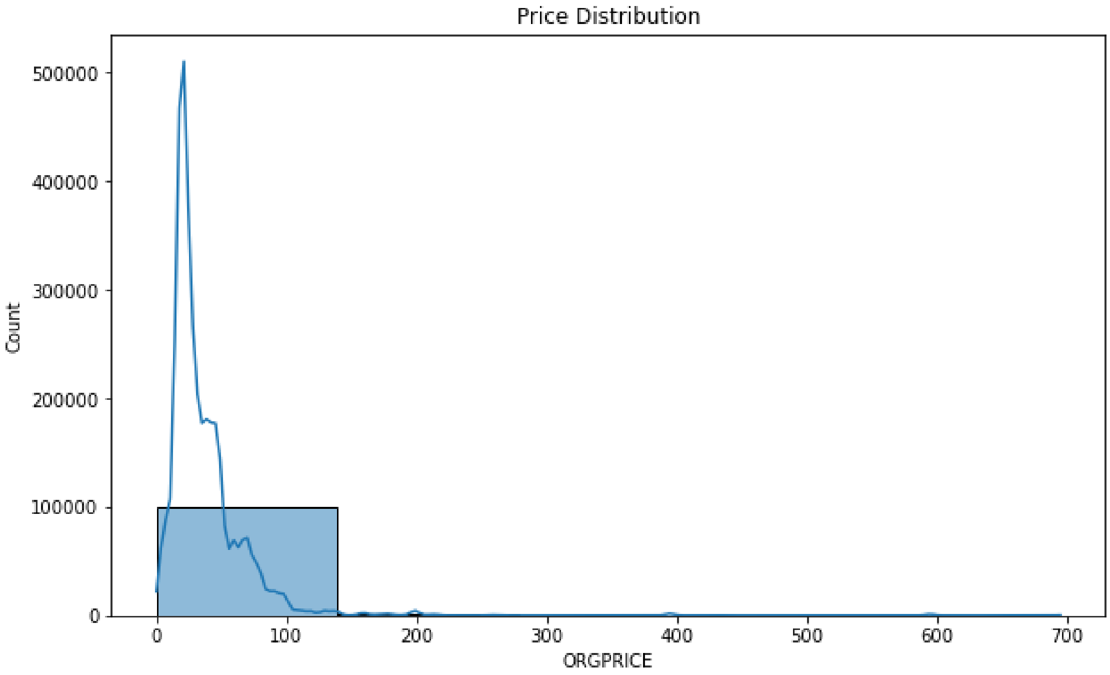
```
# Assuming 'SALEDATE' is in a datetime format
trnsact['SALEDATE'] = pd.to_datetime(trnsact['SALEDATE'])

# Extract year and month from 'SALEDATE'
trnsact['Year'] = trnsact['SALEDATE'].dt.year
trnsact['Month'] = trnsact['SALEDATE'].dt.month

# Group by year and month and calculate the mean
trnsact_group_price = trnsact.groupby(['Year', 'Month']).mean()

plt.figure(figsize=(10, 6))  # Optional: Set the figure size

# Assuming 'Year' and 'Month' are now separate columns
date_labels = [f"{year}-{month:02d}" for year, month in zip(trnsact_group_price.
 ↪index.get_level_values('Year'), trnsact_group_price.index.
 ↪get_level_values('Month'))]

plt.plot(date_labels, trnsact_group_price['ORGPRICE'], label='Original Price of␣
 ↪the Stock', color='blue', linestyle='-', linewidth=2)
plt.plot(date_labels, trnsact_group_price['SPRICE'], label='Sale Prices',␣
 ↪color='red', linestyle='--', linewidth=2)

# Add labels and a legend
plt.xlabel('Date')
plt.ylabel('Price')
plt.title('Original and Sale Price Over Time')
plt.legend()

# Display the line chart
plt.grid(True)  # Optional: Display grid lines
plt.xticks(rotation=45)  # Optional: Rotate x-axis labels for better readability
plt.show()
```

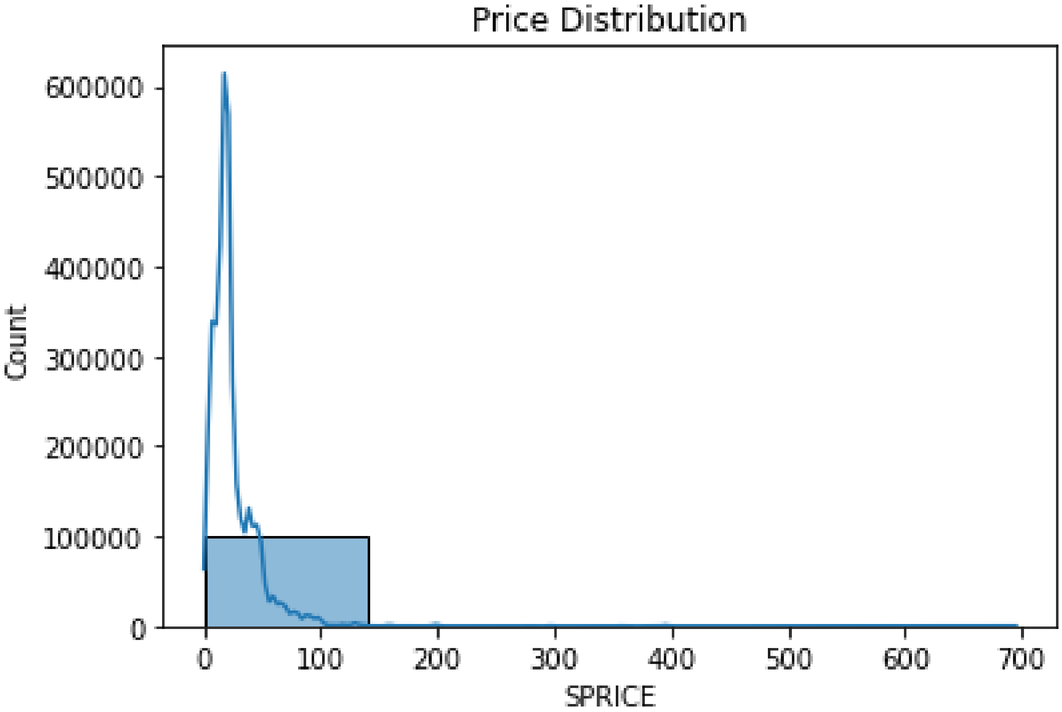


[21]:
```
trnsact.columns
```

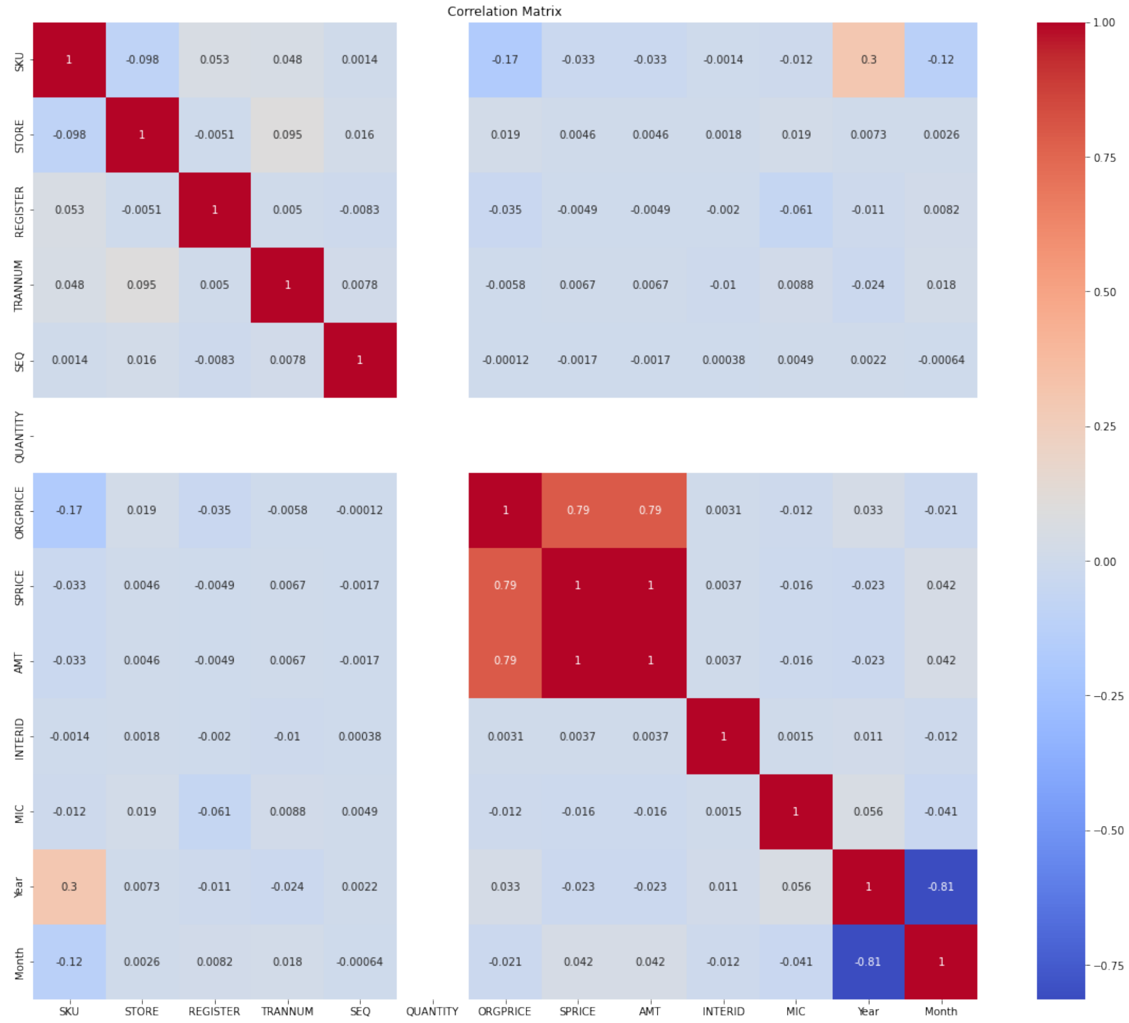
[21]:
```
Index(['SKU', 'STORE', 'REGISTER', 'TRANNUM', 'SEQ', 'SALEDATE', 'STYPE',
       'QUANTITY', 'ORGPRICE', 'SPRICE', 'AMT', 'INTERID', 'MIC', 'Year',
       'Month'],
      dtype='object')
```

[22]:
```
import seaborn as sns
# Data Distribution and Visualization
plt.figure(figsize=(10, 6))
sns.histplot(trnsact['ORGPRICE'], bins=5, kde=True)
plt.title('Price Distribution')
plt.show()
sns.histplot(trnsact['SPRICE'], bins=5, kde=True)
plt.title('Price Distribution')
plt.show()
```


Price Distribution
Count
500000
400000
300000
200000
100000
0
0
100
200
300
400
500
600
700
ORGPRICE


Price Distribution
Count
600000
500000
400000
300000
200000
100000
0
0
100
200
300
400
500
600
700
SPRICE

```
[23]: # Correlation Analysis
      correlation_matrix = trnsact.corr()
      plt.figure(figsize=(20, 17))
      sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm')
      plt.title('Correlation Matrix')
      plt.show()
```



```
[ ]:
```