Super Market Analysis

Import Libraries

```
!pip install pandas
!pip install numpy
!pip install matplotlib
import pandas as pd
import numpy as np
from matplotlib import pyplot as plt
```

```
Requirement already satisfied: pandas in c:\users\shiza\anaconda3\lib\
site-packages (1.5.3)
Requirement already satisfied: python-dateutil>=2.8.1 in c:\users\
shiza\anaconda3\lib\site-packages (from pandas) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in c:\users\shiza\
anaconda3\lib\site-packages (from pandas) (2022.7)
Requirement already satisfied: numpy>=1.21.0 in c:\users\shiza\
anaconda3\lib\site-packages (from pandas) (1.24.3)
Requirement already satisfied: six>=1.5 in c:\users\shiza\anaconda3\
lib\site-packages (from python-dateutil>=2.8.1->pandas) (1.16.0)
```

Load Dataset

```
df = pd.read_csv(r'C:\Users\shiza\Downloads\
Stores.csv',encoding='unicode_escape')
```

Summary statistics

```
print(df.describe())
```

```
       ï»¿Store ID    Store_Area  Items_Available
Daily_Customer_Count  \
count    896.000000   896.000000       896.000000
896.000000
mean     448.500000  1485.409598      1782.035714
786.350446
std      258.797218   250.237011       299.872053
265.389281
min        1.000000   775.000000       932.000000
10.000000
25%      224.750000  1316.750000      1575.500000
```

```
600.000000
50%      448.500000  1477.000000      1773.500000
780.000000
75%      672.250000  1653.500000      1982.750000
970.000000
max      896.000000  2229.000000      2667.000000
1560.000000

         Store_Sales
count     896.000000
mean    59351.305804
std     17190.741895
min     14920.000000
25%     46530.000000
50%     58605.000000
75%     71872.500000
max    116320.000000
```

Check for null values

```
print(df.isnull().sum())
```

```
ï»¿Store ID              0
Store_Area               0
Items_Available          0
Daily_Customer_Count     0
Store_Sales              0
dtype: int64
```

Correlation between variables

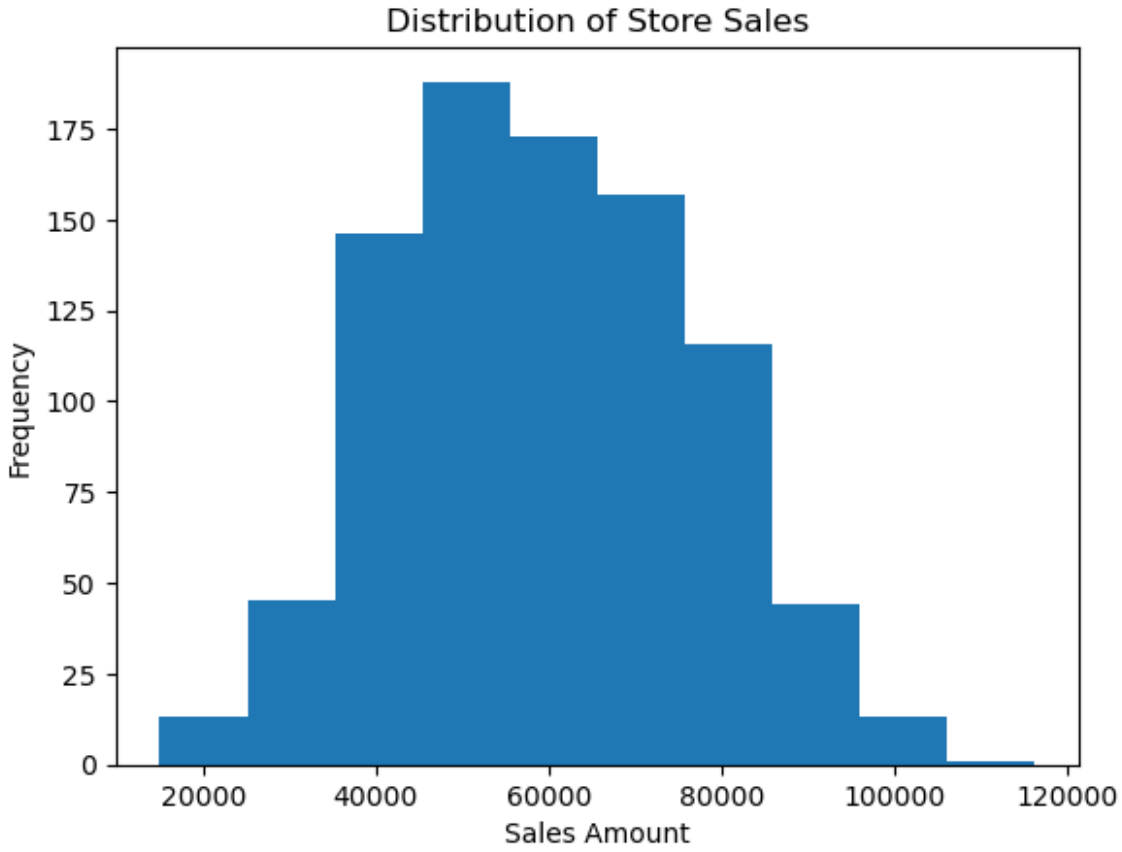
```
print(df.corr())
```

```
                      ï»¿Store ID   Store_Area  Items_Available  \
ï»¿Store ID              1.000000   -0.058705        -0.055707
Store_Area              -0.058705    1.000000         0.998891
Items_Available         -0.055707    0.998891         1.000000
Daily_Customer_Count    -0.010029   -0.041423        -0.040978
Store_Sales              0.071486    0.097474         0.098849

                      Daily_Customer_Count  Store_Sales
ï»¿Store ID                      -0.010029     0.071486
Store_Area                      -0.041423     0.097474
Items_Available                 -0.040978     0.098849
Daily_Customer_Count             1.000000     0.008629
Store_Sales                      0.008629     1.000000
```

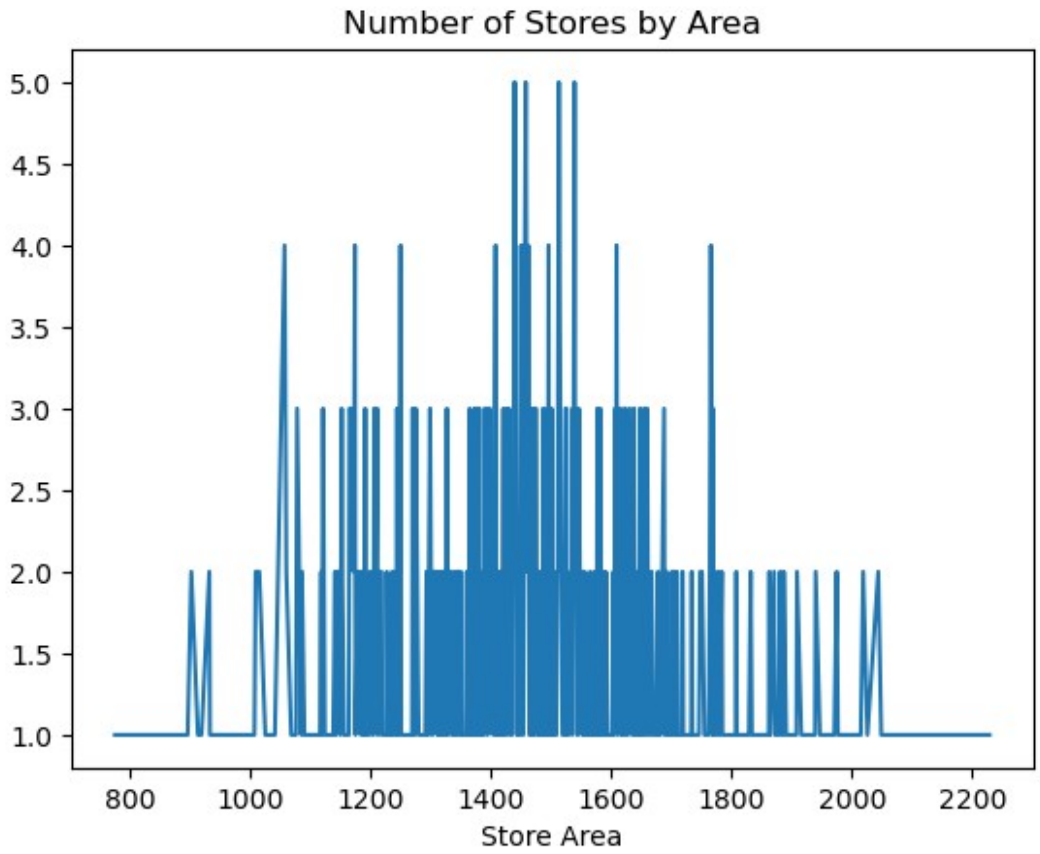
Distribution of store sales

```
df['Store_Sales'].plot(kind='hist')
plt.title('Distribution of Store Sales')
plt.xlabel('Sales Amount')
plt.show()
```



Stores by area

```
df.groupby('Store_Area')
['Daily_Customer_Count'].count().plot(kind='line')
plt.title('Number of Stores by Area')
plt.xlabel('Store Area')
plt.show()
```



Average sales by customer count

```
customer_groups = df.groupby(pd.cut(df['Daily_Customer_Count'],5))
print(customer_groups['Store_Sales'].mean())

Daily_Customer_Count
(8.45, 320.0]       63921.290323
(320.0, 630.0]      57531.673820
(630.0, 940.0]      59587.467700
(940.0, 1250.0]     60041.586538
(1250.0, 1560.0]    60630.540541
Name: Store_Sales, dtype: float64
```

Highest sales stores

```
top_stores = df.nlargest(10,'Store_Sales')
print(top_stores)

     ï»¿Store ID   Store_Area  Items_Available
Daily_Customer_Count  \
```

```
649           650         1989                2414                  860
868           869         1775                2104                  980
432           433         1365                1638                  680
408           409         1303                1587                 1310
758           759         1486                1758                  820
557           558         1137                1374                  700
866           867         1565                1854                  900
166           167         1465                1763                  680
692           693         1548                1858                  480
871           872         1800                2158                 1100


     Store_Sales
649       116320
868       105150
432       102920
408       102310
758       101820
557       101780
866       100900
166        99570
692        99480
871        98260
```

Lowest sales stores

```
bottom_stores = df.nsmallest(10,'Store_Sales')
print(bottom_stores)
```

```
     ï»¿Store ID   Store_Area  Items_Available
Daily_Customer_Count  \
31             32        1250             1508                  990
852           853        1477             1790                  880
775           776        1537             1877                  660
593           594        1624             1946                  870
352           353        1397             1686                  850
```

```
725            726          1445                1734                  900

372            373          1876                2254                 1340

277            278          1572                1869                 1030

252            253          1583                1907                  680

670            671          1461                1739                 1250
```

```
     Store_Sales
31         14920
852        16370
775        17670
593        20270
352        21300
725        21470
372        21650
277        21750
252        21830
670        22310
```