In [25]:
```
import numpy as np
import pandas as pd
from matplotlib import pyplot as plt
import seaborn as sns
import json
```

In [10]:
```
csv_data = pd.read_csv("DMV dataset/sales_data_sample.csv", encoding="cp1252")
```

In [11]:
```
exc_data = pd.read_excel("DMV dataset/Sample-Sales-Data.xlsx")
```

In [7]:
```
with open("DMV dataset/customers.json", "r") as json_file:
    json_data = json.load(json_file)
```

In [12]:
```
csv_data.head()
```

Out[12]:

	ORDERNUMBER	QUANTITYORDERED	PRICEEACH	ORDERLINENUMBER	SALES	OR
0	10107	30	95.70	2	2871.00	
1	10121	34	81.35	5	2765.90	
2	10134	41	94.74	2	3884.34	
3	10145	45	83.26	6	3746.70	
4	10159	49	100.00	14	5205.27	

5 rows × 25 columns

In [13]:
```
csv_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2823 entries, 0 to 2822
Data columns (total 25 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   ORDERNUMBER       2823 non-null   int64
 1   QUANTITYORDERED   2823 non-null   int64
 2   PRICEEACH         2823 non-null   float64
 3   ORDERLINENUMBER   2823 non-null   int64
 4   SALES             2823 non-null   float64
 5   ORDERDATE         2823 non-null   object
 6   STATUS            2823 non-null   object
 7   QTR_ID            2823 non-null   int64
 8   MONTH_ID          2823 non-null   int64
 9   YEAR_ID           2823 non-null   int64
 10  PRODUCTLINE       2823 non-null   object
 11  MSRP              2823 non-null   int64
 12  PRODUCTCODE       2823 non-null   object
 13  CUSTOMERNAME      2823 non-null   object
 14  PHONE             2823 non-null   object
 15  ADDRESSLINE1      2823 non-null   object
 16  ADDRESSLINE2      302 non-null    object
 17  CITY              2823 non-null   object
 18  STATE             1337 non-null   object
 19  POSTALCODE        2747 non-null   object
 20  COUNTRY           2823 non-null   object
 21  TERRITORY         1749 non-null   object
 22  CONTACTLASTNAME   2823 non-null   object
 23  CONTACTFIRSTNAME  2823 non-null   object
 24  DEALSIZE          2823 non-null   object
dtypes: float64(2), int64(7), object(16)
memory usage: 551.5+ KB
```

In [14]:
```
csv_data.describe()
```

Out[14]:

	ORDERNUMBER	QUANTITYORDERED	PRICEEACH	ORDERLINENUMBER	S
count	2823.000000	2823.000000	2823.000000	2823.000000	2823.0
mean	10258.725115	35.092809	83.658544	6.466171	3553.8
std	92.085478	9.741443	20.174277	4.225841	1841.8
min	10100.000000	6.000000	26.880000	1.000000	482.1
25%	10180.000000	27.000000	68.860000	3.000000	2203.4
50%	10262.000000	35.000000	95.700000	6.000000	3184.8
75%	10333.500000	43.000000	100.000000	9.000000	4508.0
max	10425.000000	97.000000	100.000000	18.000000	14082.8

In [15]:
```
csv_data.dropna()
```

Out[15]:

	ORDERNUMBER	QUANTITYORDERED	PRICEEACH	ORDERLINENUMBER	SALES
10	10223	37	100.00	1	3965.66
21	10361	20	72.55	13	1451.00
40	10270	21	100.00	9	4905.39
47	10347	30	100.00	1	3944.70
51	10391	24	100.00	4	2416.56
...	...	...	...	...	...
2667	10120	43	76.00	14	3268.00
2673	10223	26	67.20	15	1747.20
2685	10361	44	100.00	10	5001.92
2764	10361	35	100.00	11	4277.35
2791	10361	23	95.20	12	2189.60

147 rows × 25 columns

In [16]:
```
csv_data.drop_duplicates()
```

Out[16]:

	ORDERNUMBER	QUANTITYORDERED	PRICEEACH	ORDERLINENUMBER	SALES
0	10107	30	95.70	2	2871.00
1	10121	34	81.35	5	2765.90
2	10134	41	94.74	2	3884.34
3	10145	45	83.26	6	3746.70
4	10159	49	100.00	14	5205.27
...	...	...	...	...	...
2818	10350	20	100.00	15	2244.40
2819	10373	29	100.00	1	3978.51
2820	10386	43	100.00	4	5417.57
2821	10397	34	62.24	1	2116.16
2822	10414	47	65.52	9	3079.44

2823 rows × 25 columns

In [17]:
```
exc_data.head()
```

Out[17]:

	Postcode	Sales_Rep_ID	Sales_Rep_Name	Year	Value
0	2121	456	Jane	2011	84219.497311
1	2092	789	Ashish	2012	28322.192268
2	2128	456	Jane	2013	81878.997241
3	2073	123	John	2011	44491.142121
4	2134	789	Ashish	2012	71837.720959

In [18]:
```
exc_data.tail()
```

Out[18]:

	Postcode	Sales_Rep_ID	Sales_Rep_Name	Year	Value
385	2164	123	John	2012	88884.535217
386	2193	456	Jane	2013	79440.290813
387	2031	123	John	2011	65643.689454
388	2130	456	Jane	2012	66247.874869
389	2116	456	Jane	2013	3195.699054

In [19]:
```
exc_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 390 entries, 0 to 389
Data columns (total 5 columns):
 #   Column          Non-Null Count  Dtype  
---  ------          --------------  -----  
 0   Postcode        390 non-null    int64  
 1   Sales_Rep_ID    390 non-null    int64  
 2   Sales_Rep_Name  390 non-null    object 
 3   Year            390 non-null    int64  
 4   Value           390 non-null    float64
dtypes: float64(1), int64(3), object(1)
memory usage: 15.4+ KB
```

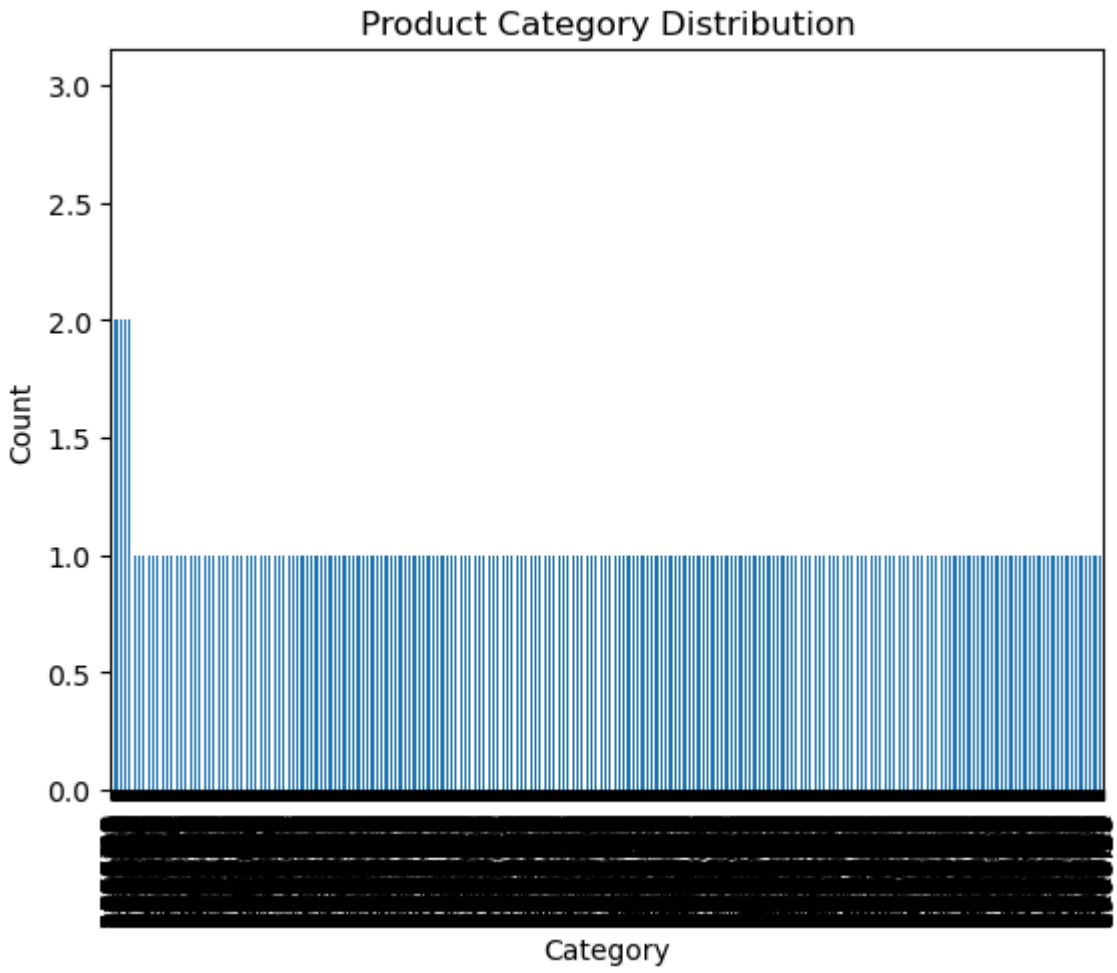
In [20]:
```
combined_data = pd.concat([csv_data, exc_data], ignore_index=True)
```

In [21]:
```
total_sales = combined_data['SALES'].sum()
print("Total Sales:", total_sales)
```

```
Total Sales: 10032628.85
```

In [22]:
```
category_sales = combined_data.groupby('ORDERNUMBER')['SALES'].mean()
```

In [23]:
```
category_counts = combined_data['SALES'].value_counts()
category_counts.plot(kind='bar')
plt.title('Product Category Distribution')
plt.xlabel('Category')
plt.ylabel('Count')
plt.show()
```

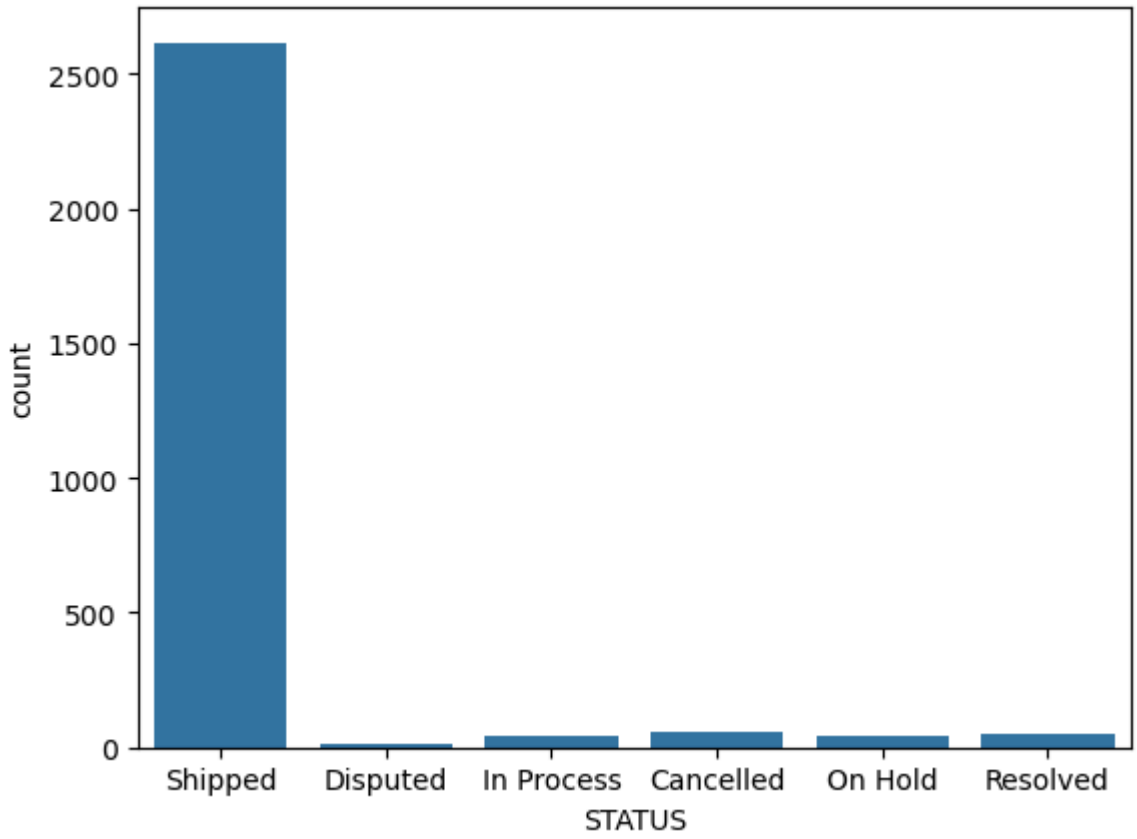


In [24]:
```
combined_data.columns
```

Out[24]:
```
Index(['ORDERNUMBER', 'QUANTITYORDERED', 'PRICEEACH', 'ORDERLINENUMBER',
       'SALES', 'ORDERDATE', 'STATUS', 'QTR_ID', 'MONTH_ID', 'YEAR_ID',
       'PRODUCTLINE', 'MSRP', 'PRODUCTCODE', 'CUSTOMERNAME', 'PHONE',
       'ADDRESSLINE1', 'ADDRESSLINE2', 'CITY', 'STATE', 'POSTALCODE',
       'COUNTRY', 'TERRITORY', 'CONTACTLASTNAME', 'CONTACTFIRSTNAME',
       'DEALSIZE', 'Postcode', 'Sales_Rep_ID', 'Sales_Rep_Name', 'Year',
       'Value'],
      dtype='object')
```

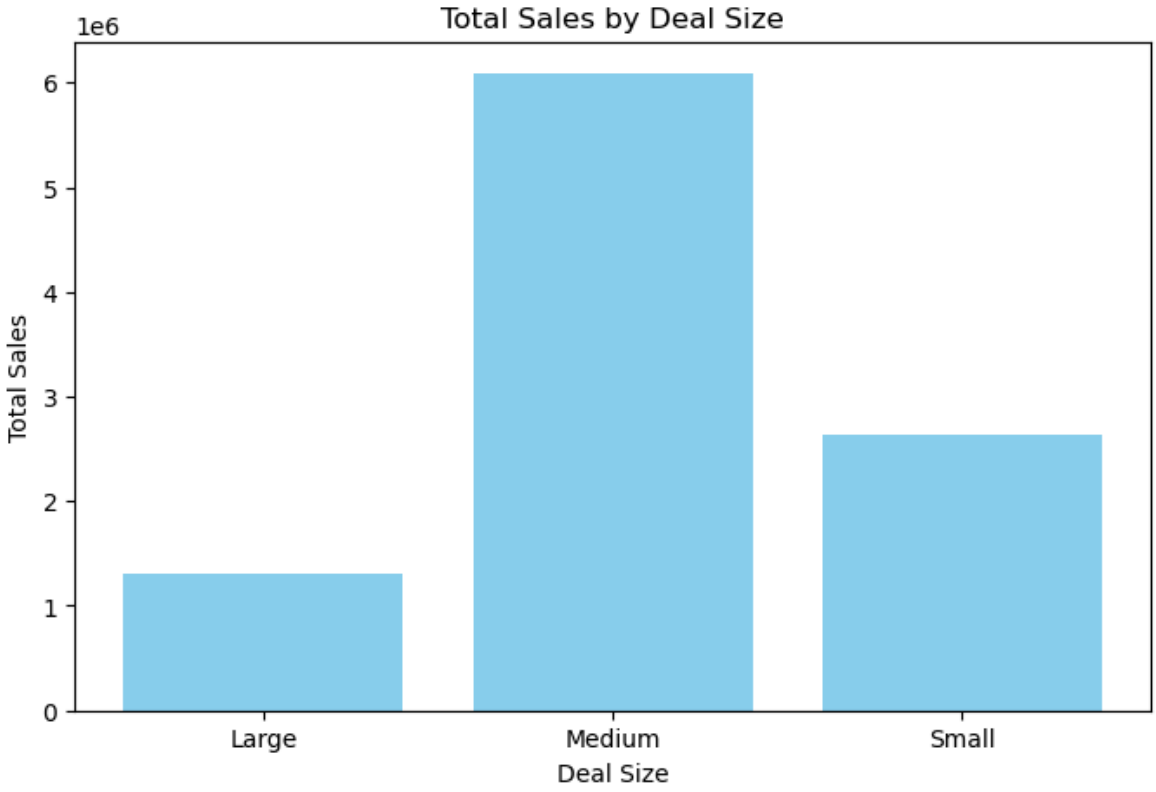
In [26]:
```
sns.countplot(x ='STATUS', data = combined_data)

plt.show()
```

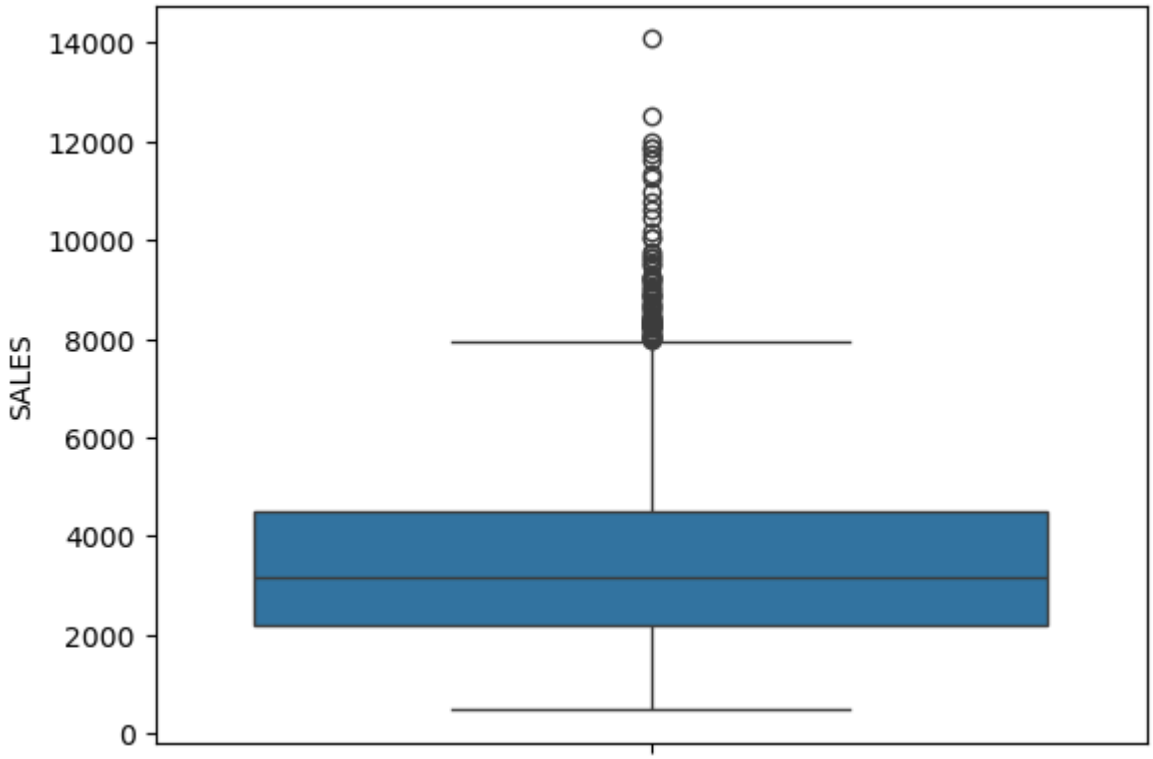


In [41]:
```
sales_by_dealsize = combined_data.groupby('DEALSIZE')['SALES'].sum().reset_index

plt.figure(figsize=(8,5))
plt.bar(sales_by_dealsize['DEALSIZE'], sales_by_dealsize['SALES'], color='skyblu
plt.xlabel('Deal Size')
plt.ylabel('Total Sales')
plt.title('Total Sales by Deal Size')
plt.show()
```



In [42]:
```
sns.boxplot(data=combined_data['SALES'])
plt.show()
```



In [43]:
```
sns.boxplot(data=combined_data['QUANTITYORDERED'])
plt.show()
```