

```
In [1]: import pandas as pd
import matplotlib.pyplot as plt
```

```
In [2]: data = pd.read_csv('sales_data_sample.csv', encoding='ISO-8859-1')
data.head()
```

Out[2]:

	ORDERNUMBER	QUANTITYORDERED	PRICEEACH	ORDERLINENUMBER	SALES	ORDERDATE	STATUS	QTR_ID	MONTH_ID	YEAR_ID	...	A
0	10107	30	95.70	2	2871.00	2/24/2003 0:00	Shipped	1	2	2003	...	{
1	10121	34	81.35	5	2765.90	5/7/2003 0:00	Shipped	2	5	2003	...	
2	10134	41	94.74	2	3884.34	7/1/2003 0:00	Shipped	3	7	2003	...	
3	10145	45	83.26	6	3746.70	8/25/2003 0:00	Shipped	3	8	2003	...	
4	10159	49	100.00	14	5205.27	10/10/2003 0:00	Shipped	4	10	2003	...	

5 rows × 25 columns



```
In [3]: 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 [4]: data.isnull().sum()
```

```
Out[4]: ORDERNUMBER          0
        QUANTITYORDERED      0
        PRICEEACH            0
        ORDERLINENUMBER      0
        SALES                 0
        ORDERDATE            0
        STATUS               0
        QTR_ID               0
        MONTH_ID             0
        YEAR_ID              0
        PRODUCTLINE          0
        MSRP                 0
        PRODUCTCODE          0
        CUSTOMERNAME         0
        PHONE                0
        ADDRESSLINE1         0
        ADDRESSLINE2        2521
        CITY                 0
        STATE                1486
        POSTALCODE           76
        COUNTRY              0
        TERRITORY            1074
        CONTACTLASTNAME      0
        CONTACTFIRSTNAME     0
        DEALSIZE             0
        dtype: int64
```

```
In [5]: data.describe()
```

```
Out[5]:
```

	ORDERNUMBER	QUANTITYORDERED	PRICEEACH	ORDERLINENUMBER	SALES	QTR_ID	MONTH_ID	YEAR_ID	MSRP
count	2823.000000	2823.000000	2823.000000	2823.000000	2823.000000	2823.000000	2823.000000	2823.000000	2823.000000
mean	10258.725115	35.092809	83.658544	6.466171	3553.889072	2.717676	7.092455	2003.81509	100.715551
std	92.085478	9.741443	20.174277	4.225841	1841.865106	1.203878	3.656633	0.69967	40.187912
min	10100.000000	6.000000	26.880000	1.000000	482.130000	1.000000	1.000000	2003.00000	33.000000
25%	10180.000000	27.000000	68.860000	3.000000	2203.430000	2.000000	4.000000	2003.00000	68.000000
50%	10262.000000	35.000000	95.700000	6.000000	3184.800000	3.000000	8.000000	2004.00000	99.000000
75%	10333.500000	43.000000	100.000000	9.000000	4508.000000	4.000000	11.000000	2004.00000	124.000000
max	10425.000000	97.000000	100.000000	18.000000	14082.800000	4.000000	12.000000	2005.00000	214.000000



```
In [6]: total_sales = data['SALES'].sum()
average_order_value = data['SALES'].mean()

category_distribution = data['PRODUCTLINE'].value_counts()
```

```
In [7]: category_distribution.plot(kind='bar', title='Product Category Distribution')
plt.xlabel('Product Category')
plt.ylabel('Count')
plt.show()
```

