Basic data analysis queries

In [11]:
```
#Given a data frame containing information about the prices of different products,
#find the following information:

# 1.The average price of each product category.
# 2.The highest-priced product in each category.
# 3.The total number of products sold in each category.
# 4.The total revenue generated by each category.
```

In [1]:
```
import pandas as pd

# Create a sample dataframe
data = {'product_name': ['Apple', 'Banana', 'Mango', 'Grapes',
                         'Watermelon', 'Pineapple','Apple','Banana',
                         'Mango','Grapes','Watermelon','Pineapple'],
        'price': [1.5, 0.5, 2.5, 3.5, 5, 3,2,0.7,2.8,3.2,5.2,3.5],
        'category': ['Fruit', 'Fruit', 'Fruit', 'Fruit', 'Fruit',
                     'Fruit','Fruit', 'Fruit', 'Fruit', 'Fruit', 'Fruit', 'Fruit'],
        'quantity': [10, 20, 15, 25, 30, 20,25,30,15,25,30,20]}
df = pd.DataFrame(data)
```

In [3]:
```
df
```

Out[3]:

	product_name	price	category	quantity
0	Apple	1.5	Fruit	10
1	Banana	0.5	Fruit	20
2	Mango	2.5	Fruit	15
3	Grapes	3.5	Fruit	25
4	Watermelon	5.0	Fruit	30
5	Pineapple	3.0	Fruit	20
6	Apple	2.0	Fruit	25
7	Banana	0.7	Fruit	30
8	Mango	2.8	Fruit	15
9	Grapes	3.2	Fruit	25
10	Watermelon	5.2	Fruit	30
11	Pineapple	3.5	Fruit	20

In [2]:
```
# 1.The average price of each product category.
df2 = df.groupby('category')['price'].mean()
print(df2)

#or
df3 = df.pivot_table(index='category', values='price', aggfunc='mean')
print(df3)
```

```
category
Fruit    2.783333
Name: price, dtype: float64
              price
category
Fruit     2.783333
```

In [3]:
```
# 2.The highest-priced product in each category.
df4 = df.groupby('product_name')['price'].sum()
high_pricepr = df4.idxmax()
print(high_pricepr)
```

```
Watermelon
```

In [4]:
```
# 3.The total number of products sold in each category.
df5 = df.groupby('product_name')['quantity'].sum()
print(df5)

#or
df6 = df.pivot_table(
    index='product_name', values='quantity', aggfunc='sum')
print(df6)
```

```
product_name
Apple         35
Banana        50
Grapes        50
Mango         30
Pineapple     40
Watermelon    60
Name: quantity, dtype: int64
              quantity
product_name
Apple               35
Banana              50
Grapes              50
Mango               30
Pineapple           40
Watermelon          60
```

In [5]:
```
# 4.The total revenue generated by each category.
df
```

Out[5]:

	product_name	price	category	quantity
0	Apple	1.5	Fruit	10
1	Banana	0.5	Fruit	20
2	Mango	2.5	Fruit	15
3	Grapes	3.5	Fruit	25
4	Watermelon	5.0	Fruit	30
5	Pineapple	3.0	Fruit	20
6	Apple	2.0	Fruit	25
7	Banana	0.7	Fruit	30
8	Mango	2.8	Fruit	15
9	Grapes	3.2	Fruit	25
10	Watermelon	5.2	Fruit	30
11	Pineapple	3.5	Fruit	20

In [6]:
```
# 4.The total revenue generated by each category.
#creating a column name revenue

df['revenue'] = df['quantity']*df['price']
```

In [7]:
```
df
```

Out[7]:

	product_name	price	category	quantity	revenue
0	Apple	1.5	Fruit	10	15.0
1	Banana	0.5	Fruit	20	10.0
2	Mango	2.5	Fruit	15	37.5
3	Grapes	3.5	Fruit	25	87.5
4	Watermelon	5.0	Fruit	30	150.0
5	Pineapple	3.0	Fruit	20	60.0
6	Apple	2.0	Fruit	25	50.0
7	Banana	0.7	Fruit	30	21.0
8	Mango	2.8	Fruit	15	42.0
9	Grapes	3.2	Fruit	25	80.0
10	Watermelon	5.2	Fruit	30	156.0
11	Pineapple	3.5	Fruit	20	70.0

In [8]:
```
#total revenue collected by each product name
df8 = df.groupby('category')['revenue'].sum()
print(df8)

#or

df9 = df.pivot_table(
    index='category', values='revenue', aggfunc='sum')
print(df9)
```

```
category
Fruit    779.0
Name: revenue, dtype: float64
          revenue
category
Fruit       779.0
```

In [9]:
```python
#total revenue collected by each product name
dft = df.groupby('product_name')['revenue'].sum()
print(dft)

#or

dfr = df.pivot_table(
    index='product_name', values='revenue', aggfunc='sum')
print(dfr)
```

```
product_name
Apple          65.0
Banana         31.0
Grapes        167.5
Mango          79.5
Pineapple     130.0
Watermelon    306.0
Name: revenue, dtype: float64
              revenue
product_name
Apple            65.0
Banana           31.0
Grapes          167.5
Mango            79.5
Pineapple       130.0
Watermelon      306.0
```

In [ ]: