Untitled1 (1)

April 30, 2025

[21]:
```
import pandas as pd
import matplotlib.pyplot as plt
```

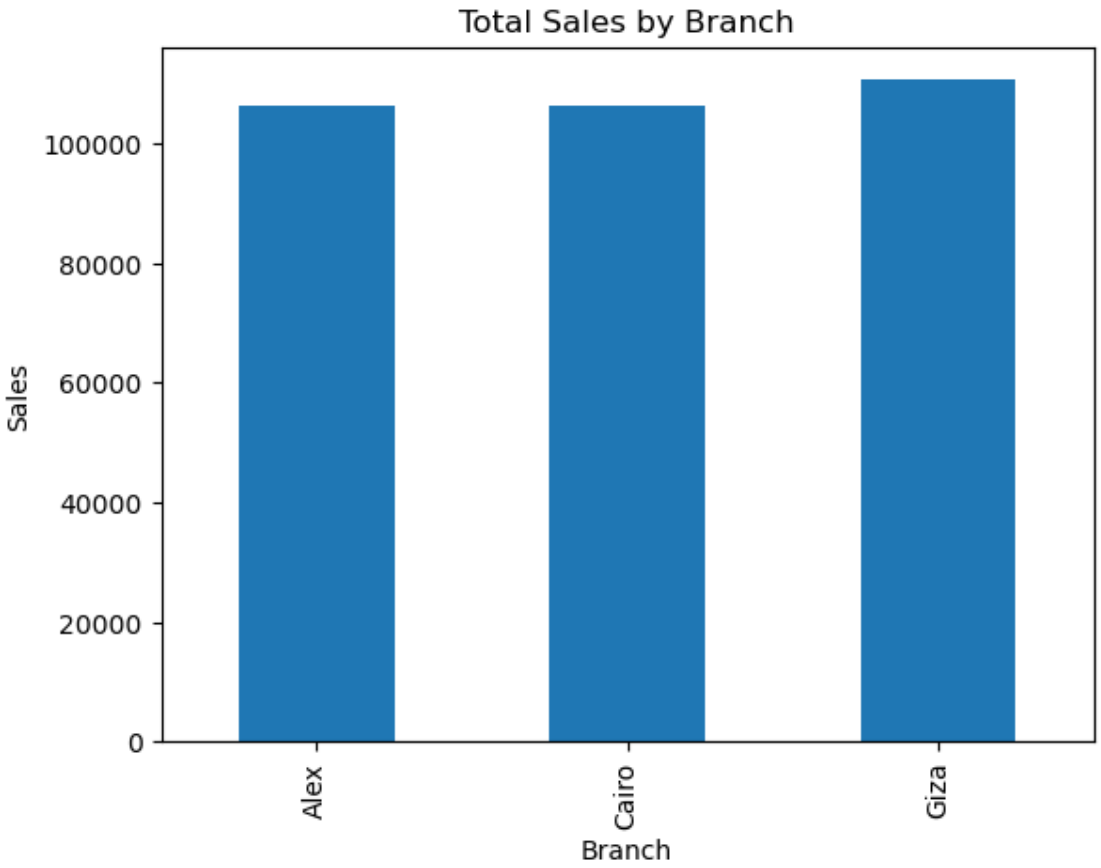
[22]:
```
df = pd.read_csv(r'C:\Users\TOP10\Desktop\SuperMarket Analysis.csv')
```

[23]:
```
print(df.head())
```

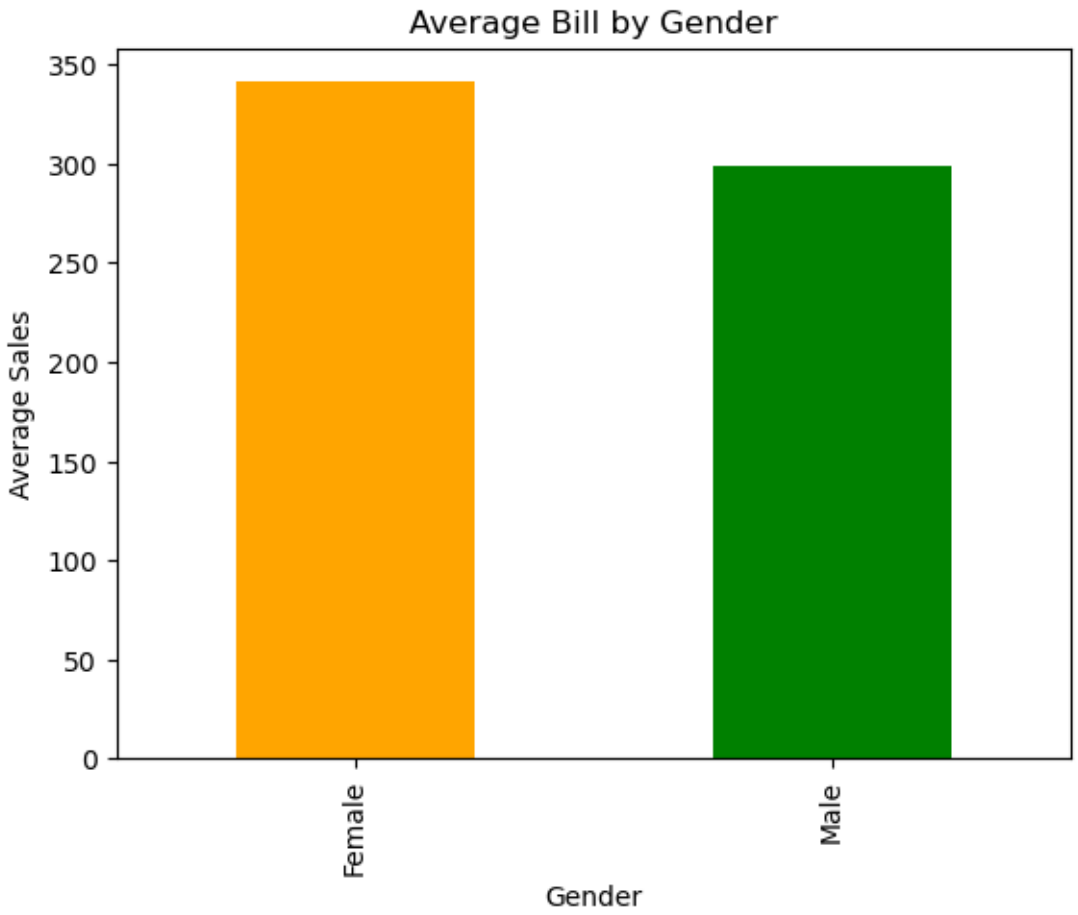
```
    Invoice ID Branch       City Customer type  Gender  \
0  750-67-8428   Alex     Yangon        Member  Female
1  226-31-3081   Giza  Naypyitaw        Normal  Female
2  631-41-3108   Alex     Yangon        Normal  Female
3  123-19-1176   Alex     Yangon        Member  Female
4  373-73-7910   Alex     Yangon        Member  Female

             Product line  Unit price  Quantity   Tax 5%     Sales       Date  \
0       Health and beauty       74.69         7  26.1415  548.9715   1/5/2019
1  Electronic accessories       15.28         5   3.8200   80.2200   3/8/2019
2      Home and lifestyle       46.33         7  16.2155  340.5255   3/3/2019
3       Health and beauty       58.22         8  23.2880  489.0480  1/27/2019
4       Sports and travel       86.31         7  30.2085  634.3785   2/8/2019

          Time      Payment    cogs  gross margin percentage  gross income  \
0   1:08:00 PM      Ewallet  522.83                 4.761905       26.1415
1  10:29:00 AM         Cash   76.40                 4.761905        3.8200
2   1:23:00 PM  Credit card  324.31                 4.761905       16.2155
3   8:33:00 PM      Ewallet  465.76                 4.761905       23.2880
4  10:37:00 AM      Ewallet  604.17                 4.761905       30.2085

   Rating
0     9.1
1     9.6
2     7.4
3     8.4
4     5.3
```

[24]:
```
sales_by_branch = df.groupby('Branch')['Sales'].sum()
```

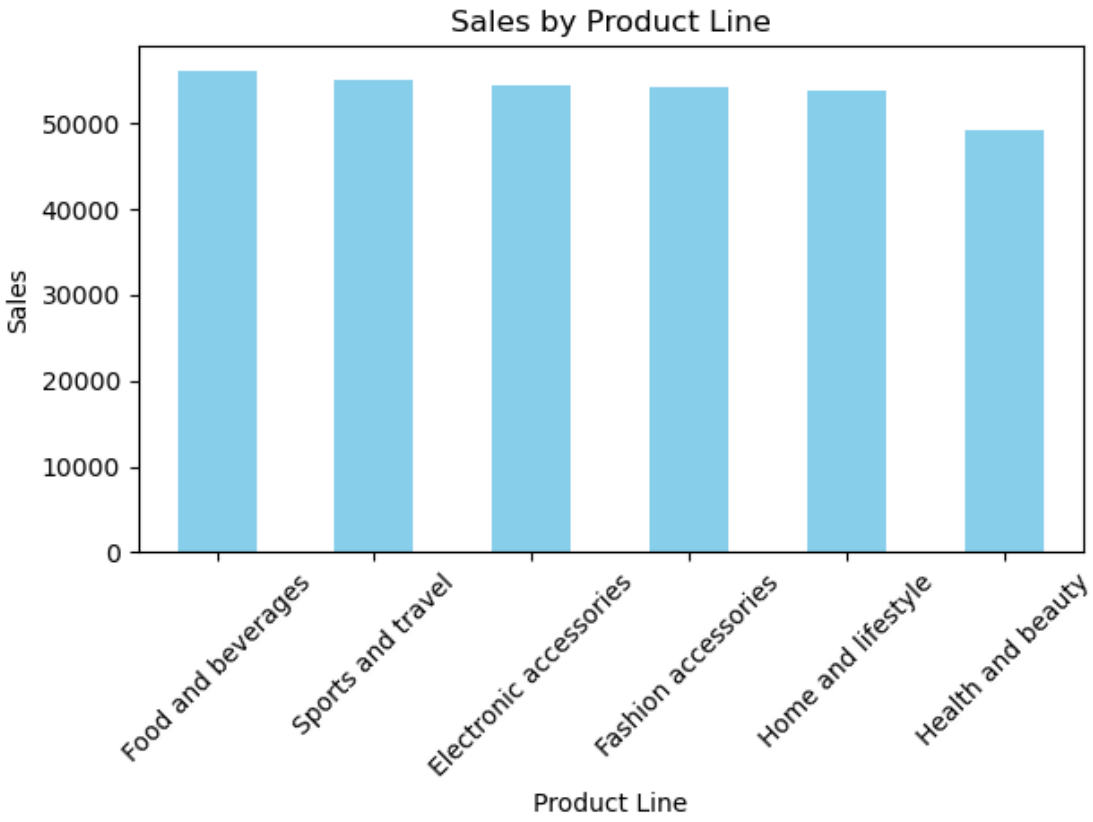
[25]:
```
sales_by_branch.plot(kind='bar', title='Total Sales by Branch')
plt.xlabel('Branch')
plt.ylabel('Sales')
plt.show()
```



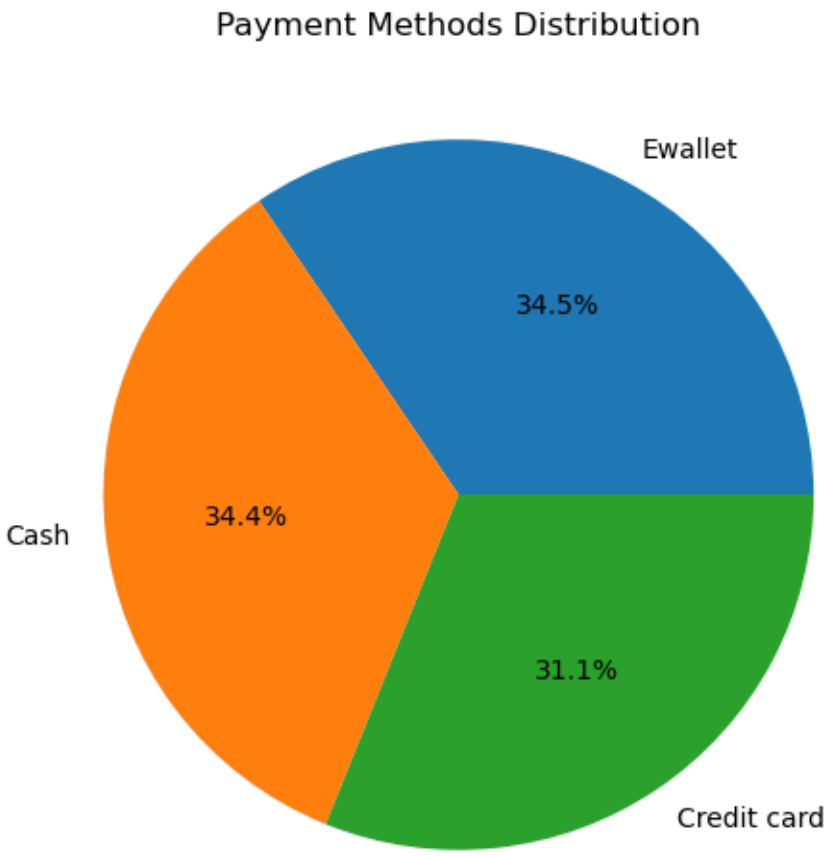
[26]:
```
avg_by_gender = df.groupby('Gender')['Sales'].mean()
avg_by_gender.plot(kind='bar', title='Average Bill by Gender', color=['orange',␣
↪'green'])
plt.xlabel('Gender')
plt.ylabel('Average Sales')
plt.show()
```



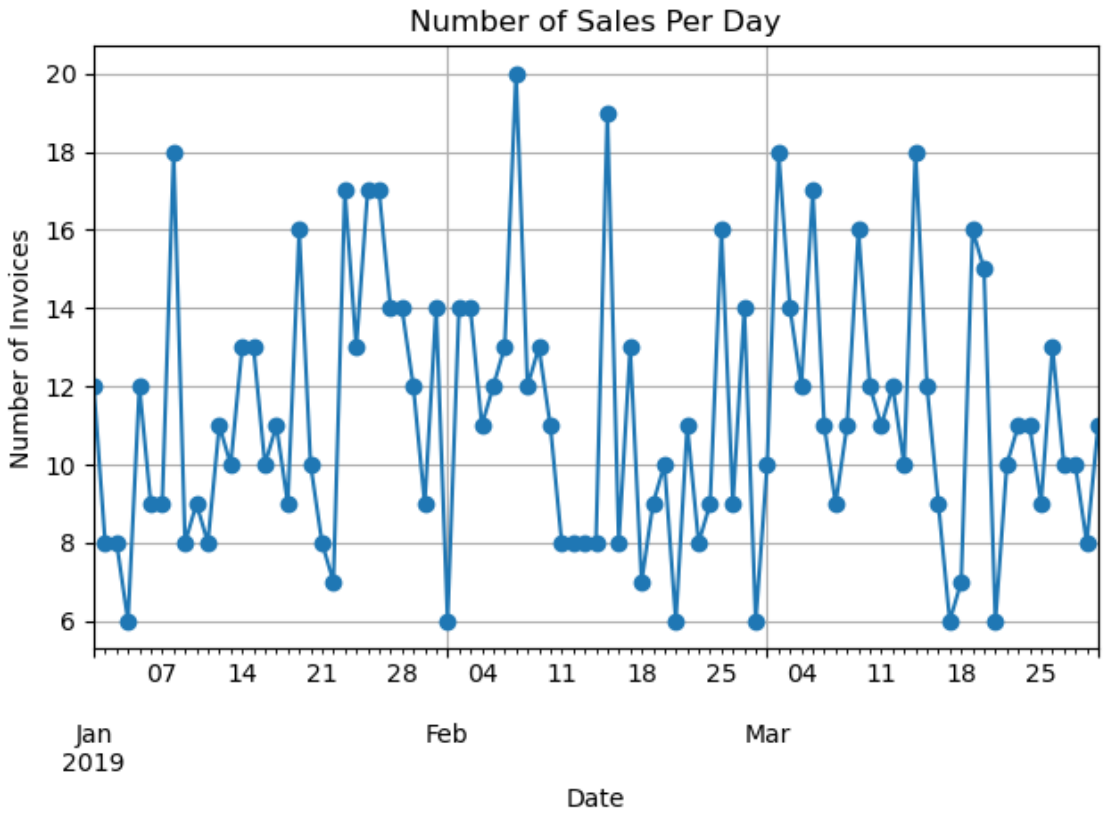
[27]:
```
sales_by_product = df.groupby('Product line')['Sales'].sum().
 ↪sort_values(ascending=False)
sales_by_product.plot(kind='bar', title='Sales by Product Line',␣
 ↪color='skyblue')
plt.xlabel('Product Line')
plt.ylabel('Sales')
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```



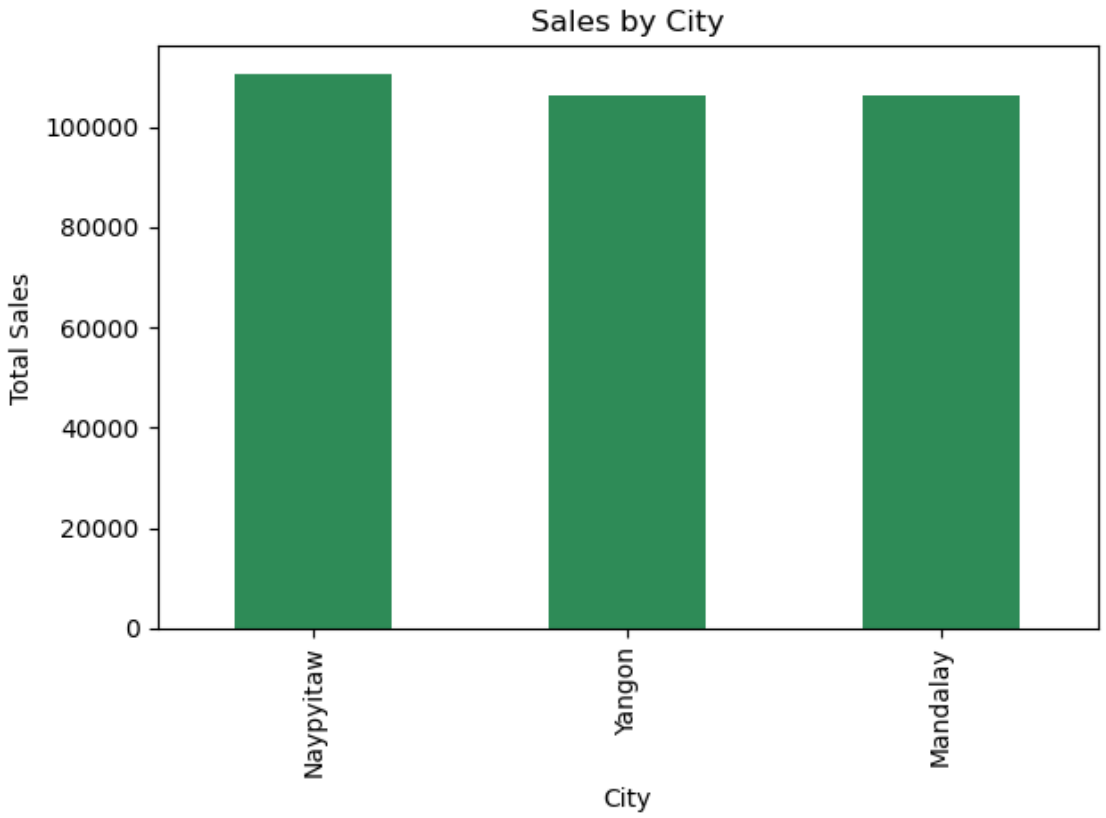
[28]:
```
payment_counts = df['Payment'].value_counts()
payment_counts.plot(kind='pie', autopct='%1.1f%%', title='Payment Methods␣
↪Distribution', figsize=(6,6))
plt.ylabel('')
plt.show()
```



[29]:
```
df['Date'] = pd.to_datetime(df['Date'])
daily_sales = df.groupby('Date').size()

daily_sales.plot(kind='line', title='Number of Sales Per Day', marker='o')
plt.xlabel('Date')
plt.ylabel('Number of Invoices')
plt.grid(True)
plt.tight_layout()
plt.show()
```



[30]:
```
sales_by_city = df.groupby('City')['Sales'].sum().sort_values(ascending=False)
sales_by_city.plot(kind='bar', title='Sales by City', color='seagreen')
plt.xlabel('City')
plt.ylabel('Total Sales')
plt.tight_layout()
plt.show()
```


Sales by City
Total Sales
100000
80000
60000
40000
20000
0
Naypyitaw
Yangon
Mandalay
City

[ ]: