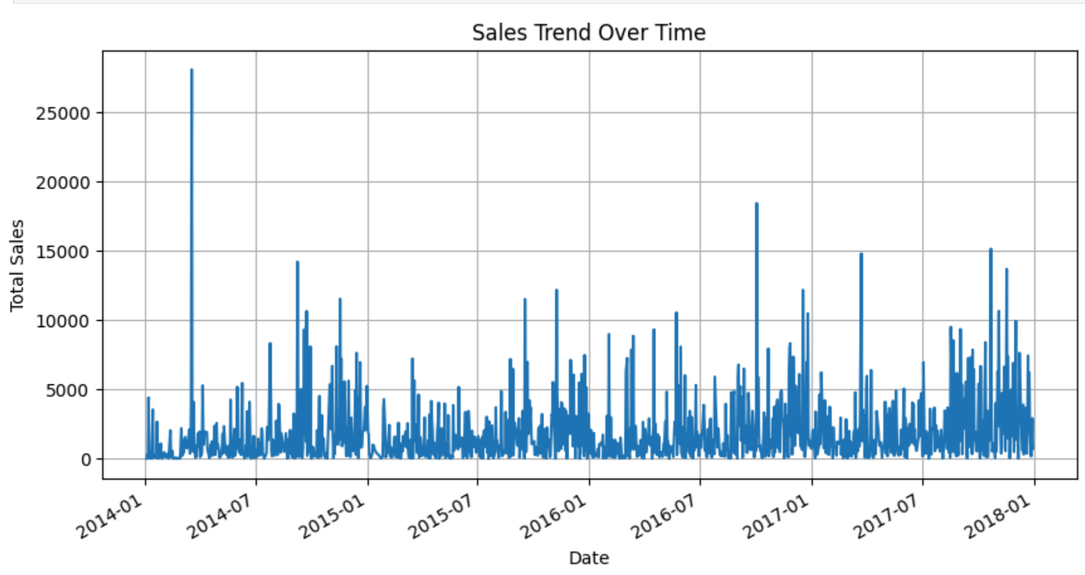
In [1]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

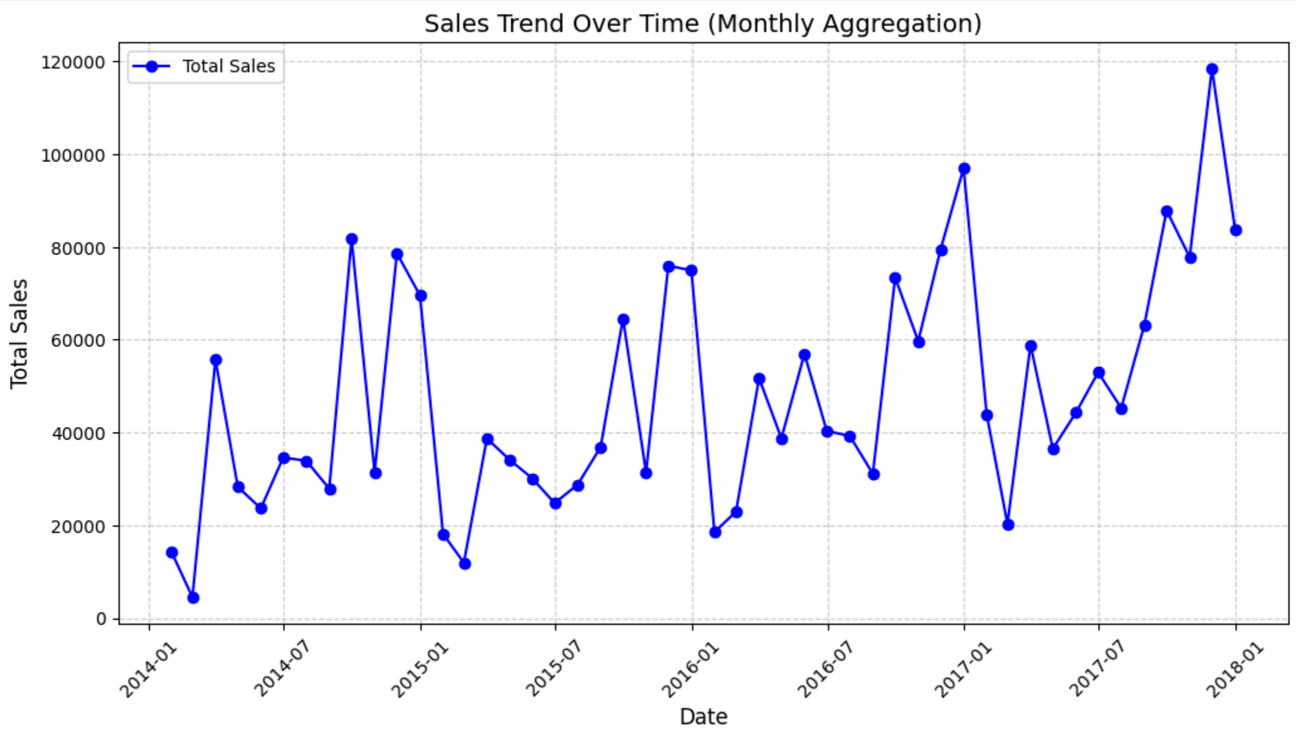
In [2]:
```
#  **Extract: Load the dataset**
df = pd.read_csv("superstore_sales.csv", encoding="ISO-8859-1")

#  **Transform: Data Cleaning and Processing**
# Convert date columns to datetime format
df['Order Date'] = pd.to_datetime(df['Order Date'])
df['Ship Date'] = pd.to_datetime(df['Ship Date'])

# Fill missing values if any
df.fillna(0, inplace=True)

# Remove duplicate records
df.drop_duplicates(inplace=True)

#  **Load: Save the cleaned data**
df.to_csv("cleaned_superstore_sales.csv", index=False)
```

In [3]:
```
#  **Data Visualization**

#  **Sales Trend Over Time**
plt.figure(figsize=(10,5))
df.groupby("Order Date")["Sales"].sum().plot()
plt.title("Sales Trend Over Time")
plt.xlabel("Date")
plt.ylabel("Total Sales")
plt.grid()
plt.show()
```



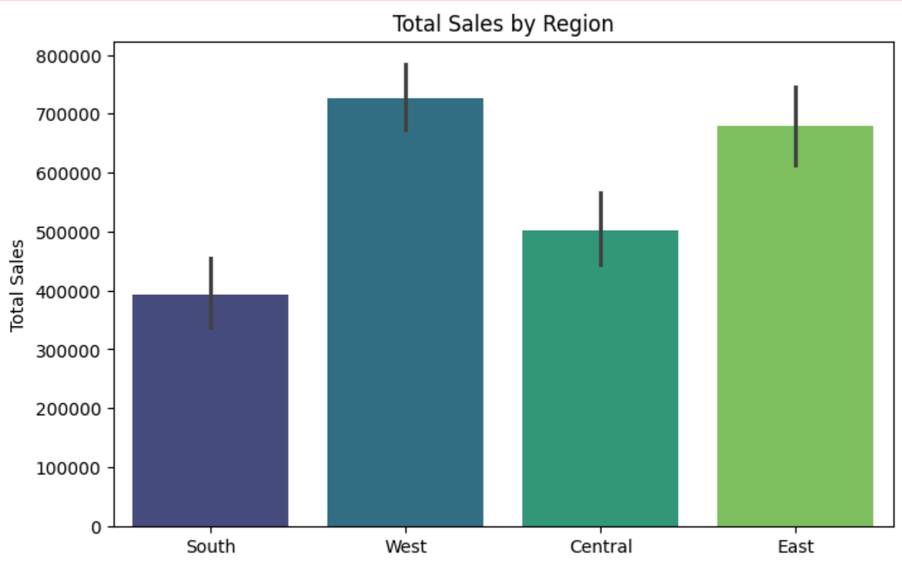
In [8]:
```
# Convert 'Order Date' to datetime if not already
df['Order Date'] = pd.to_datetime(df['Order Date'])

# Aggregate sales by month
df_monthly_sales = df.groupby(pd.Grouper(key="Order Date", freq="M"))["Sales"].sum()

# Plot
plt.figure(figsize=(12, 6))
plt.plot(df_monthly_sales.index, df_monthly_sales, marker="o", linestyle="-", color="b", label="Total Sales")

# Formatting
plt.title("Sales Trend Over Time (Monthly Aggregation)", fontsize=14)
plt.xlabel("Date", fontsize=12)
plt.ylabel("Total Sales", fontsize=12)
plt.xticks(rotation=45)
plt.grid(True, linestyle="--", alpha=0.6)
plt.legend()
plt.show()
```

```
C:\Users\Prathamesh\AppData\Local\Temp\ipykernel_14372\3135241609.py:5: FutureWarning: 'M' is deprecated and will be removed in a future version, please use 'ME' instead.
  df_monthly_sales = df.groupby(pd.Grouper(key="Order Date", freq="M"))["Sales"].sum()
```



In [5]:
```
# **Sales by Region**
plt.figure(figsize=(8,5))
sns.barplot(x=df["Region"], y=df["Sales"], estimator=sum, palette="viridis")
plt.title("Total Sales by Region")
plt.xlabel("Region")
plt.ylabel("Total Sales")
plt.show()
```

```
C:\Users\Prathamesh\AppData\Local\Temp\ipykernel_14372\3267374789.py:3: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `x` variable to `hue` and set `legend=False` for the same effect.

  sns.barplot(x=df["Region"], y=df["Sales"], estimator=sum, palette="viridis")
```



In [7]:
```
#  **Sales vs Profit Scatter Plot**
plt.figure(figsize=(8,5))
sns.scatterplot(x=df["Sales"], y=df["Profit"], alpha=0.5)
plt.title("Sales vs Profit")
plt.xlabel("Sales")
plt.ylabel("Profit")
plt.show()
```


Sales vs Profit
Profit
Sales
8000
6000
4000
2000
0
−2000
−4000
−6000
0
5000
10000
15000
20000

In [ ]: