Task 2: Time Series Analysis (Stock Prices)

Step 1: Prepare the Time Series Data¶

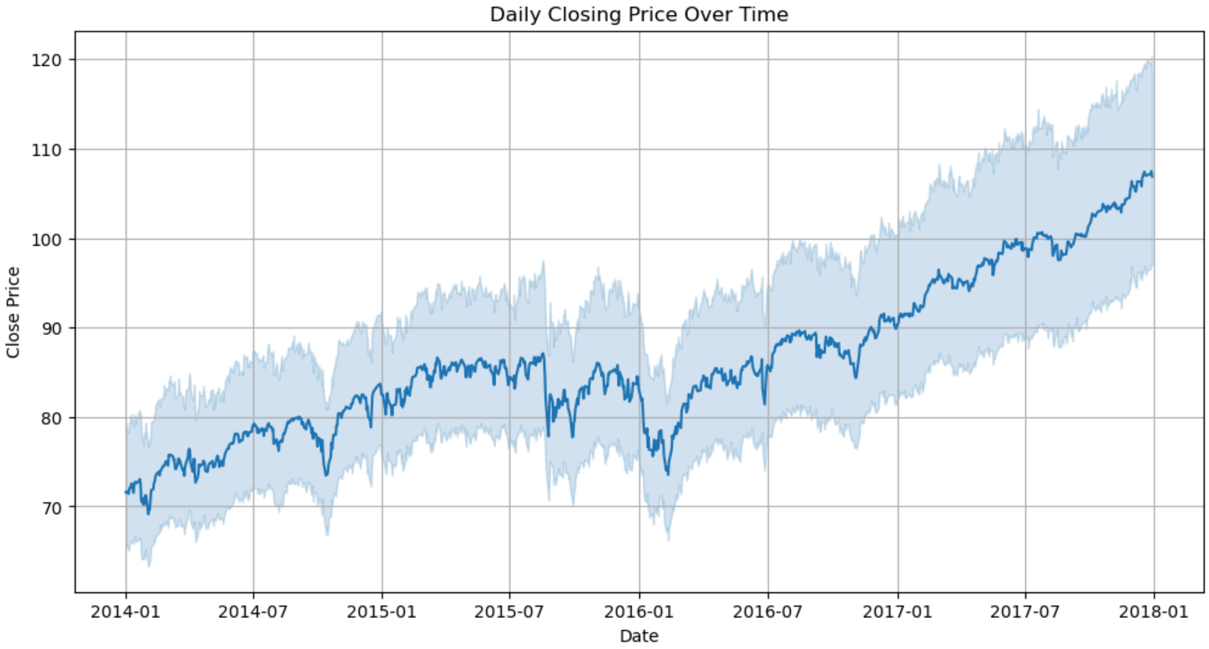
In [1]:

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

df = pd.read_csv("stock_prices_cleaned.csv")

# Convert 'date' column to datetime and set it as index
df['date'] = pd.to_datetime(df['date'])
df.set_index('date', inplace=True)

# Sort by date just in case
df = df.sort_index()

# Keep only the 'close' price for analysis
ts = df['close']
```

Step 2: Plot Time Series to Identify Patterns

In [2]:

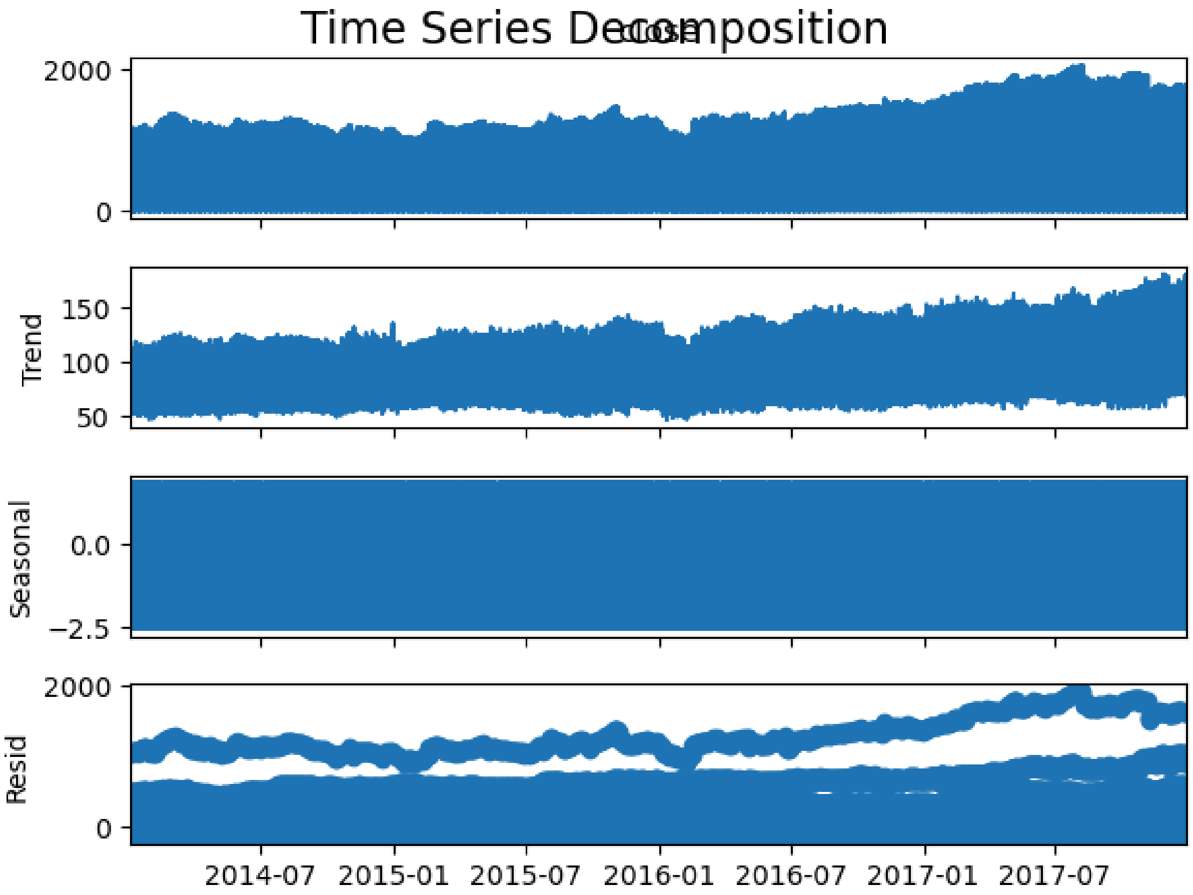
```
plt.figure(figsize=(12, 6))
sns.lineplot(data=ts)
plt.title("Daily Closing Price Over Time")
plt.xlabel("Date")
plt.ylabel("Close Price")
plt.grid(True)
plt.show()
```



Step 3: Decompose into Trend, Seasonality, Residuals

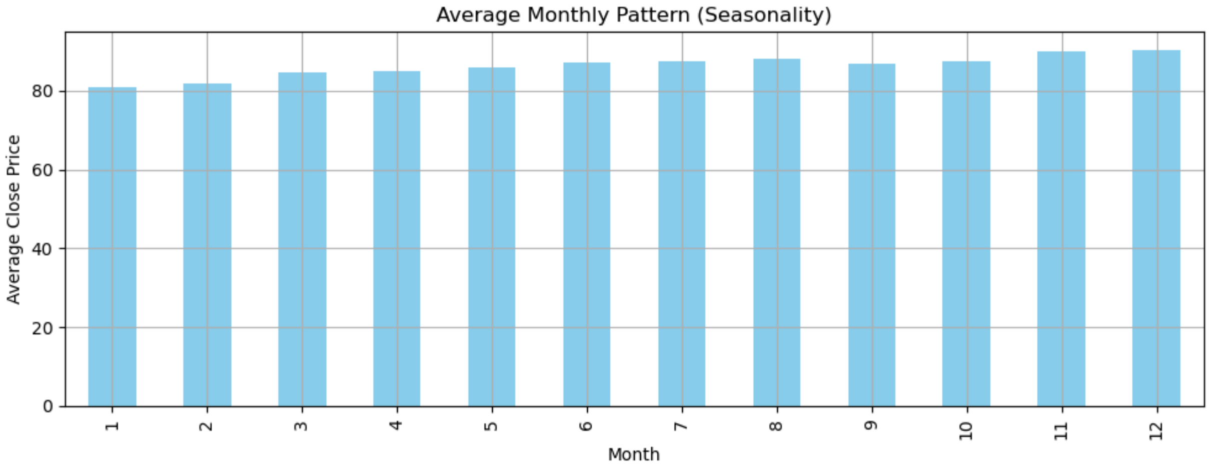
In [3]:

```
from statsmodels.tsa.seasonal import seasonal_decompose

# Decompose the time series
decomposition = seasonal_decompose(ts, model='additive', period=30)  # 30 days ~ 1
decomposition.plot()
plt.suptitle("Time Series Decomposition", fontsize=16)
plt.show()
```



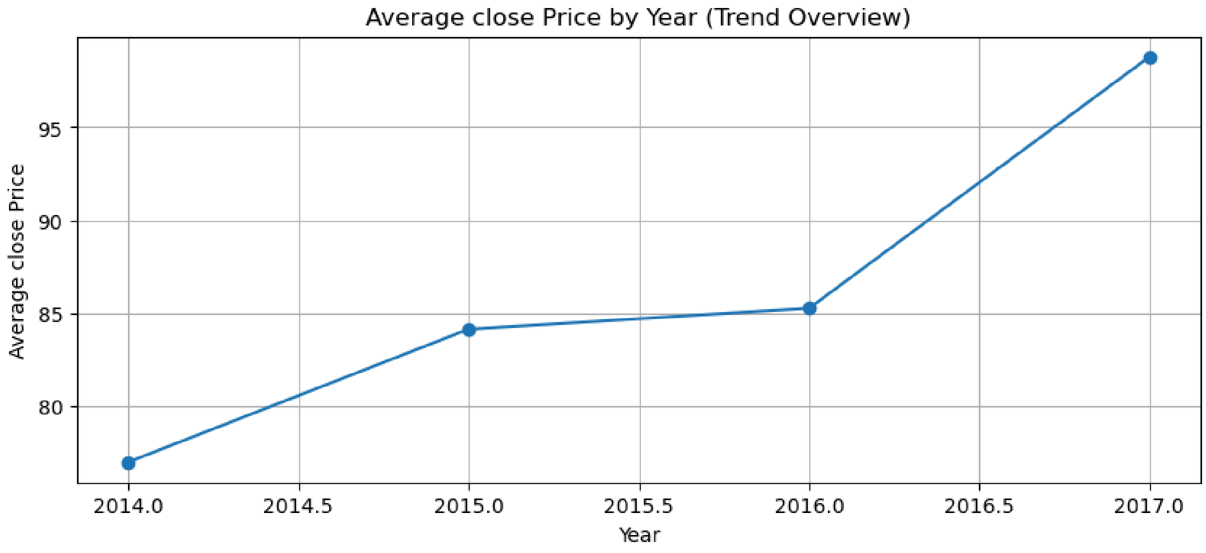
In [5]:

```
# Add year and month columns
df['Year'] = df.index.year
df['Month'] = df.index.month

# Monthly average trend (Seasonality)
monthly_avg = df.groupby('Month')['close'].mean()

plt.figure(figsize=(10, 4))
monthly_avg.plot(kind='bar', color='skyblue')
plt.title("Average Monthly Pattern (Seasonality)")
plt.xlabel("Month")
plt.ylabel("Average Close Price")
plt.grid(True)
plt.tight_layout()
plt.savefig("monthly_seasonality.png")  # ✅ Save the figure
plt.show()
```



In [6]:

```
# Yearly average
yearly_avg = df.groupby('Year')['close'].mean()

plt.figure(figsize=(10, 4))
yearly_avg.plot(marker='o')
plt.title("Average close Price by Year (Trend Overview)")
plt.xlabel("Year")
plt.ylabel("Average close Price")
plt.grid(True)
plt.show()
```



In [ ]: