In [4]:
```
import pandas as pd


file_path = "C:/Users/sunny/Downloads/Stock Time series/stocks.csv"
```

In [5]:
```
stocks_data = pd.read_csv(file_path)
```

In [6]:
```
print(stocks_data.head())  # View the first few rows
```

```
  Ticker       Date        Open        High         Low       Close  \
0      A   2/7/2023  150.639999  155.229996  150.639999  154.649994
1      A   2/8/2023  153.880005  154.580002  151.169998  151.919998
2      A   2/9/2023  153.779999  154.330002  150.419998  150.869995
3      A  2/10/2023  149.460007  151.339996  149.220001  151.009995
4      A  2/13/2023  150.949997  154.259995  150.919998  153.850006

    Adj Close    Volume
0  154.414230  83322600
1  151.688400  64120100
2  150.639999  56007100
3  151.009995  57450700
4  153.850006  62199000
```

In [7]:
```
print(stocks_data.info())  # Get information about the DataFrame
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 248 entries, 0 to 247
Data columns (total 8 columns):
 #   Column     Non-Null Count  Dtype
---  ------     --------------  -----
 0   Ticker     248 non-null    object
 1   Date       248 non-null    object
 2   Open       248 non-null    float64
 3   High       248 non-null    float64
 4   Low        248 non-null    float64
 5   Close      248 non-null    float64
 6   Adj Close  248 non-null    float64
 7   Volume     248 non-null    int64
dtypes: float64(5), int64(1), object(2)
memory usage: 15.6+ KB
None
```

In [8]:
```
stocks_data.Ticker.value_counts()
```

Out[8]:
```
A    62
B    62
C    62
D    62
Name: Ticker, dtype: int64
```

In [9]:
```
descriptive_stats = stocks_data.groupby('Ticker')
descriptive_stats['Close'].describe()
```

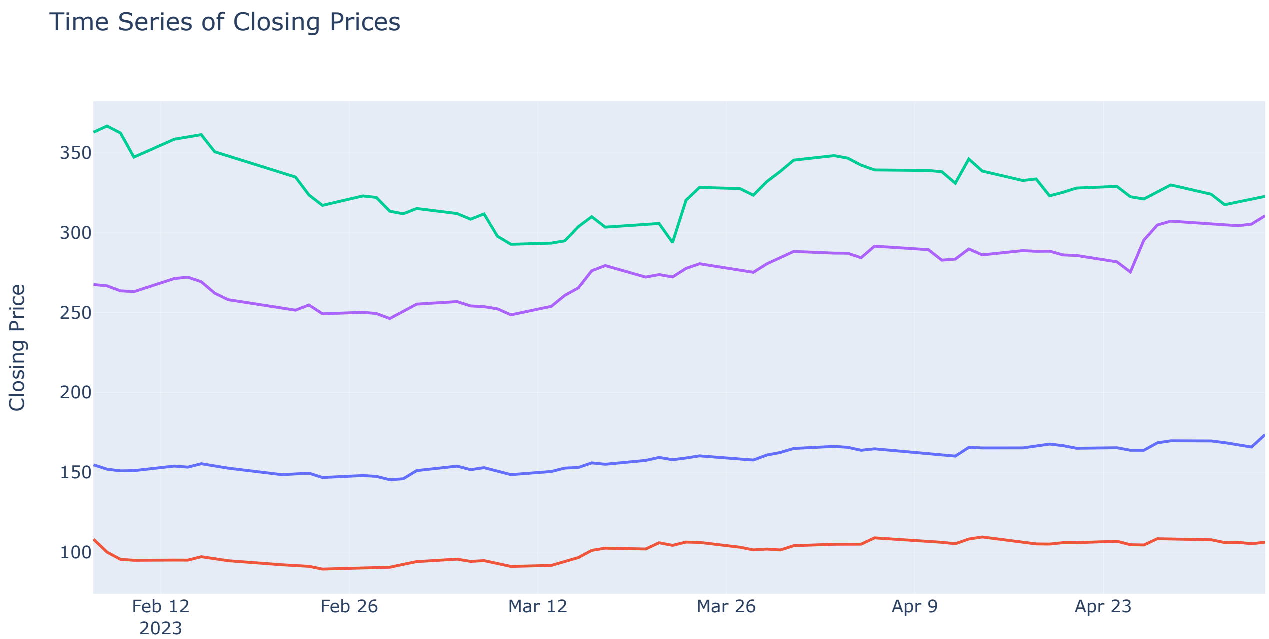
Out[9]:

Ticker	count	mean	std	min	25%	50%	75%	max
A	62.0	158.240645	7.360485	145.309998	152.077499	158.055000	165.162506	173.570007
B	62.0	275.039839	17.676231	246.270004	258.742500	275.810013	287.217506	310.649994
C	62.0	327.614677	18.554419	292.760010	315.672493	325.600006	338.899994	366.829987
D	62.0	100.631532	6.279464	89.349998	94.702501	102.759998	105.962503	109.459999

Time Series Analysis

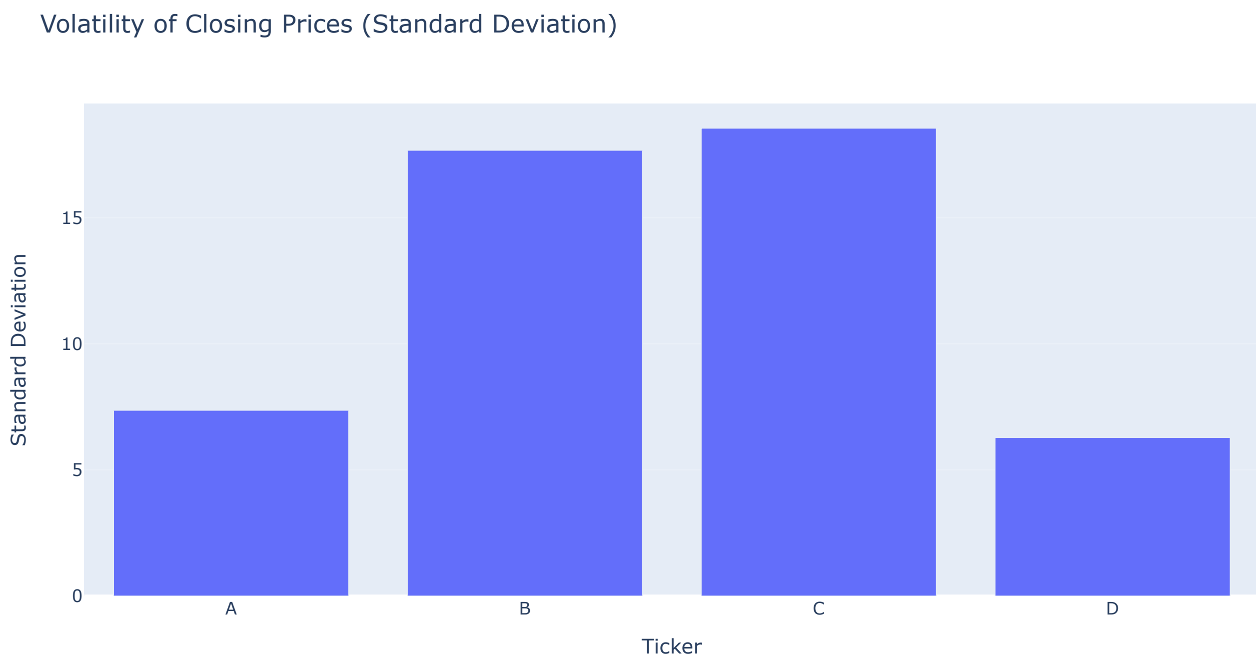
In [10]:
```
import plotly.express as px
from plotly.subplots import make_subplots
import plotly.graph_objects as go
```

In [11]:
```
stocks_data['Date'] = pd.to_datetime(stocks_data['Date'])
pivot_data = stocks_data.pivot(index='Date',columns='Ticker',values='Close')
fig = make_subplots(rows=1,cols=1)
fig.add_trace(go.Scatter(x=pivot_data.index,y=pivot_data['A'],name='A'))
fig.add_trace(go.Scatter(x=pivot_data.index,y=pivot_data['D'],name='D'))
fig.add_trace(go.Scatter(x=pivot_data.index,y=pivot_data['C'],name='C'))
fig.add_trace(go.Scatter(x=pivot_data.index,y=pivot_data['B'],name='B'))

fig.update_layout(
    title_text="Time Series of Closing Prices",
    xaxis_title='Date',
    yaxis_title='Closing Price',
    legend_title='Ticker',
    showlegend=True
)

fig.show()
```



Volatility Analysis

In [12]:
```
volatility = pivot_data.std()
fig = px.bar(
    volatility,
    x=volatility.index,
    y=volatility.values,
    labels={
        'y':'Standard Deviation',
        'x':'Ticker'
    },
    title='Volatility of Closing Prices (Standard Deviation)'
    )
fig.show()
```




```
It indicates that C and B stocks were more prone to price fluctuations during this period compared to A and D.
```

Correlation Analysis

In [14]:
```
correlation_matrix = pivot_data.corr()
fig = go.Figure(
    data=go.Heatmap(
        z=correlation_matrix,
        x=correlation_matrix.columns,
        y=correlation_matrix.columns,
        colorscale='blues',
        colorbar=dict(title='correlation'),
        text=correlation_matrix.round(2).values,
        texttemplate="%{text}"
    )
)
fig.update_layout(
    title='Correlation Matrix of Closing Prices',
    xaxis_title="Ticker",
    yaxis_title="Ticker",
)
fig.show()
```


```
- Values close to +1 indicate a strong positive correlation, meaning that as one stock’s price increases,
  the other tends to increase as well.
- Values close to -1 indicate a strong negative correlation, where one stock’s price increase corresponds
  to a decrease in the other.
- Values around 0 indicate a lack of correlation.
From the heatmap, we can observe that there are varying degrees of positive correlations between the stock prices,
with some pairs showing stronger correlations than others. For instance, A and B seem to have a relatively higher
positive correlation.
```

Comparative Analysis

In [16]:
```
# Calculating the percentage change in closing prices
percentage_change = ((pivot_data.iloc[-1] - pivot_data.iloc[0]) / pivot_data.iloc[0]) * 100
fig = px.bar(
    percentage_change,
    x=percentage_change.index,
    y=percentage_change.values,
    labels={'y': 'Percentage Change (%)', 'x': 'Ticker'},
    title='Percentage Change in Closing Prices'
    )
fig.show()
```

- B: The highest positive change of approximately 16.10%.

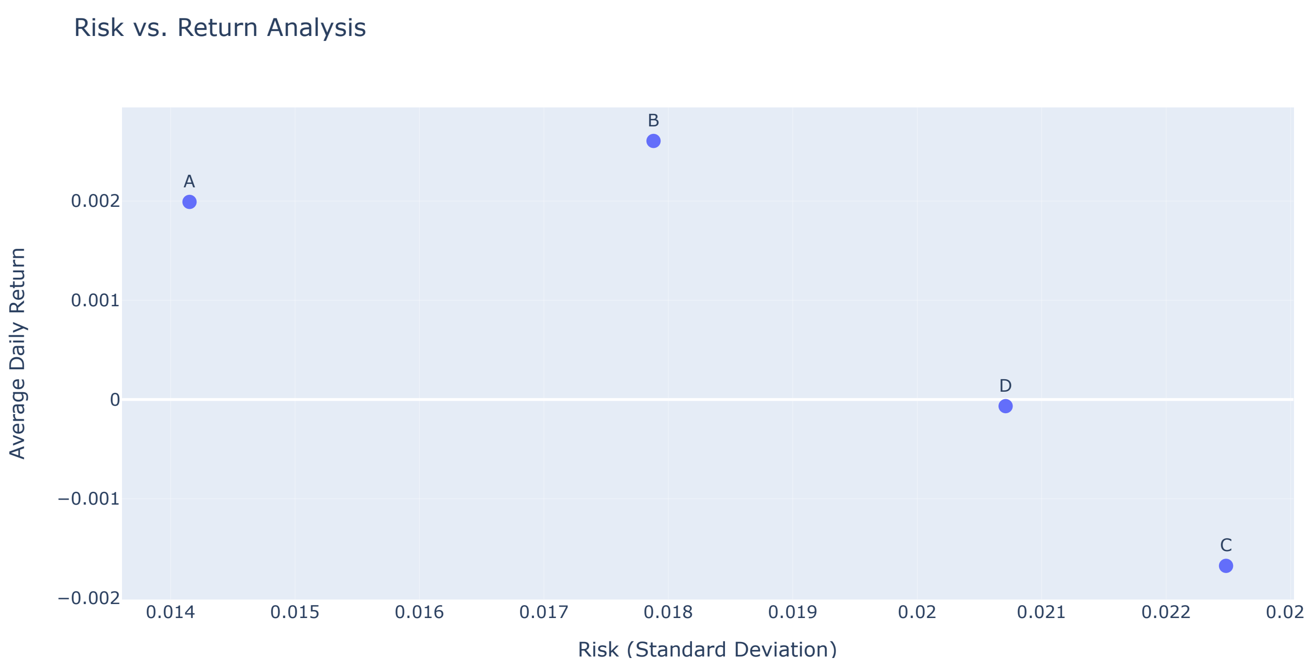
- A: Exhibited a positive change of approximately 12.23%. It indicates a solid performance, though slightly lower than B's.

- D: Showed a slight negative change of about -1.69%. It indicates a minor decline in its stock price over the observed period.

- C: Experienced the most significant negative change, at approximately -11.07%. It suggests a notable decrease in its stock price during the period.

Daily Risk Vs. Return Analysis

In [17]:
```
daily_returns = pivot_data.pct_change().dropna()
avg_daily_return = daily_returns.mean()
risk = daily_returns.std()

risk_return_df = pd.DataFrame({'Risk':risk,'Average Daily Return':avg_daily_return})

fig = go.Figure()

fig.add_trace(
    go.Scatter(
        x=risk_return_df["Risk"],
        y=risk_return_df['Average Daily Return'],
        mode="markers+text",
        text=risk_return_df.index,
        textposition="top center",
        marker=dict(size=10)
    )
)
fig.update_layout(
    title='Risk vs. Return Analysis',
    xaxis_title='Risk (Standard Deviation)',
    yaxis_title='Average Daily Return',
    showlegend=False
)
fig.show()
```



So, A shows the lowest risk combined with a positive average daily return, suggesting a more stable investment with consistent returns. D has higher volatility than A and, on average, a slightly negative daily return, indicating a riskier and less rewarding investment during this period.

B shows moderate risk with the highest average daily return, suggesting a potentially more rewarding investment, although with higher volatility compared to A. C exhibits the highest risk and a negative average daily return, indicating it was the most volatile and least rewarding investment among these stocks over the analyzed period.