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

In [2]:
```
# Comparison of 2 companies
#1) INFOSYS
#2) TCS
```

In [3]:
```
# Load datasets
infosys_data = pd.read_csv("C:/Users/HP/Desktop/comparision_model/INFY.NS.csv")
tcs_data = pd.read_csv("C:/Users/HP/Desktop/comparision_model/TCS.NS.csv")
```

In [4]:
```
# Check the first few rows of each dataset
```

In [5]:
```
print("Infosys Data:")
print(infosys_data.head())
```

```
Infosys Data:
         Date         Open         High          Low        Close  \
0  2023-03-15  1438.000000  1442.849976  1416.849976  1419.650024
1  2023-03-16  1417.800049  1417.800049  1398.699951  1404.099976
2  2023-03-17  1430.000000  1442.900024  1411.400024  1420.699951
3  2023-03-20  1419.699951  1419.699951  1384.849976  1403.650024
4  2023-03-21  1399.000000  1405.000000  1390.000000  1390.699951

     Adj Close   Volume
0  1382.921875  7556182
1  1367.774048  7323117
2  1383.944580  9090018
3  1367.335693  8221829
4  1354.720703  8536755
```

In [6]:
```
print("\nTCS Data:")
print(tcs_data.head())
```

```

TCS Data:
         Date         Open         High          Low        Close  \
0  2023-03-15  3250.000000  3260.350098  3192.000000  3198.899902
1  2023-03-16  3208.000000  3219.800049  3172.000000  3185.000000
2  2023-03-17  3150.500000  3221.399902  3144.000000  3179.300049
3  2023-03-20  3169.649902  3169.649902  3095.050049  3143.300049
4  2023-03-21  3143.300049  3156.750000  3097.449951  3106.100098

     Adj Close   Volume
0  3144.305908  1780522
1  3130.643311  1901060
2  3125.040771  6739966
3  3089.655273  2289468
4  3053.089844  1815297
```

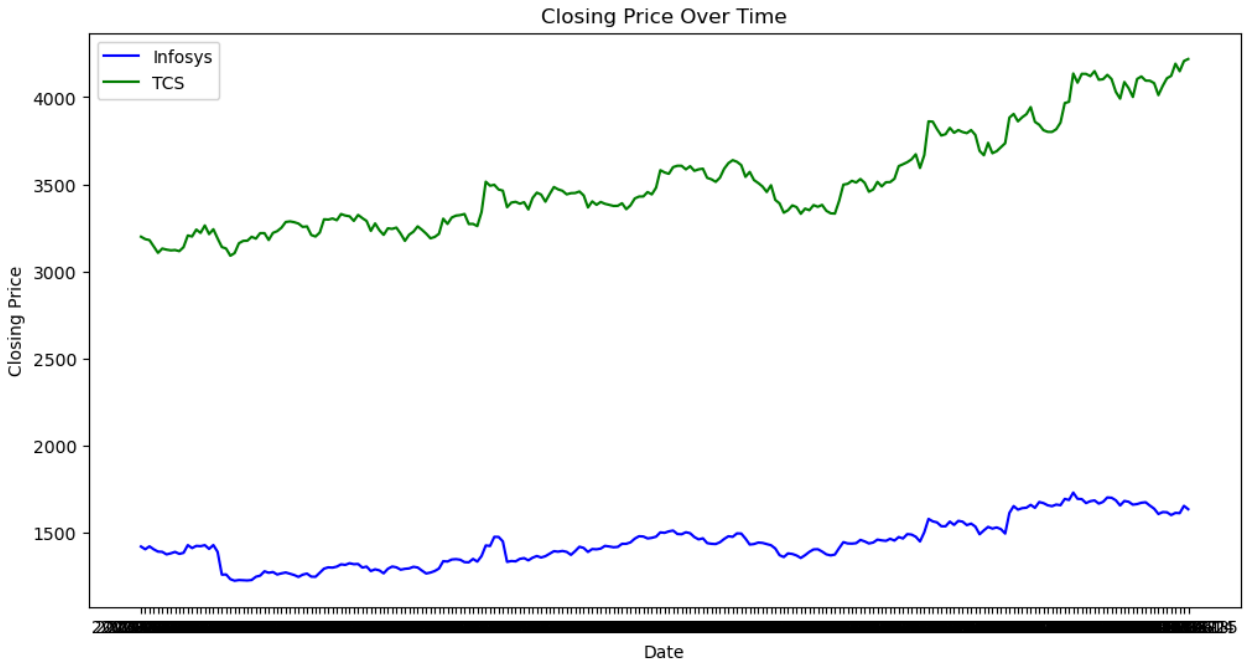
In [7]:
```
# Check summary statistics for Infosys
infosys_summary = infosys_data.describe()
print("\nInfosys Summary Statistics:")
print(infosys_summary)
```

```
Infosys Summary Statistics:
              Open         High          Low        Close    Adj Close  \
count   247.000000   247.000000   247.000000   247.000000   247.000000
mean   1439.120442  1450.515181  1427.024902  1439.657286  1425.413222
std     128.535312   131.229393   126.840419   129.293164   138.512096
min    1225.949951  1230.000000  1185.300049  1223.400024  1191.749146
25%    1342.775024  1350.450012  1332.325012  1344.525024  1327.344421
50%    1425.599976  1436.650024  1414.300049  1427.250000  1405.408936
75%    1505.875000  1519.075012  1490.775024  1504.349976  1493.242249
max    1729.000000  1733.000000  1687.949951  1729.449951  1729.449951

             Volume
count  2.470000e+02
mean   6.635708e+06
std    4.945585e+06
min    2.272209e+06
25%    4.239817e+06
50%    5.618346e+06
75%    7.557654e+06
max    5.317170e+07
```

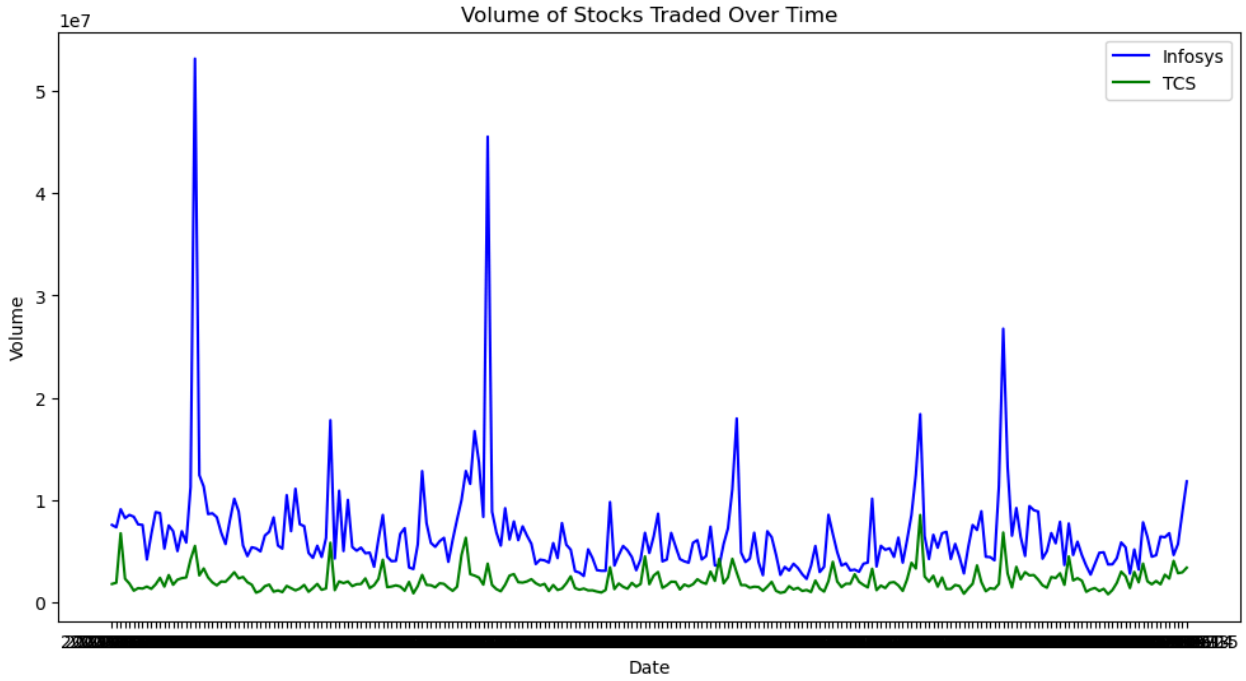
In [8]:
```
# Check summary statistics for TCS
tcs_summary = tcs_data.describe()
print("\nTCS Summary Statistics:")
print(tcs_summary)
```

```
TCS Summary Statistics:
              Open         High          Low        Close    Adj Close  \
count   247.000000   247.000000   247.000000   247.000000   247.000000
mean   3510.739276  3539.761731  3483.434626  3512.581987  3485.456994
std     289.014118   297.856074   285.290231   292.635255   308.074583
min    3090.000000  3113.000000  3070.250000  3089.600098  3036.871582
25%    3284.000000  3307.474976  3262.974976  3282.699951  3233.065064
50%    3434.949951  3464.899902  3413.600098  3443.550049  3418.821045
75%    3658.349976  3708.699951  3641.000000  3669.675049  3652.749268
max    4205.000000  4241.000000  4177.000000  4219.250000  4219.250000

             Volume
count  2.470000e+02
mean   2.043061e+06
std    1.062203e+06
min    7.722910e+05
25%    1.376618e+06
50%    1.775689e+06
75%    2.355943e+06
max    8.531230e+06
```

In [9]:
```
# Plotting stock prices for Infosys and TCS
plt.figure(figsize=(12, 6))
plt.plot(infosys_data['Date'], infosys_data['Close'], label='Infosys', color='blue')
plt.plot(tcs_data['Date'], tcs_data['Close'], label='TCS', color='green')
plt.xlabel('Date')
plt.ylabel('Closing Price')
plt.title('Closing Price Over Time')
plt.legend()
plt.show()
```



In [10]:
```
# Plotting volume of stocks traded for Infosys and TCS
plt.figure(figsize=(12, 6))
plt.plot(infosys_data['Date'], infosys_data['Volume'], label='Infosys', color='blue')
plt.plot(tcs_data['Date'], tcs_data['Volume'], label='TCS', color='green')
plt.xlabel('Date')
plt.ylabel('Volume')
plt.title('Volume of Stocks Traded Over Time')
plt.legend()
plt.show()
```



In [11]:
```
# Calculate daily returns for Infosys and TCS
infosys_data['Daily Return'] = infosys_data['Close'].pct_change()
tcs_data['Daily Return'] = tcs_data['Close'].pct_change()

# Plotting daily returns for Infosys and TCS
plt.figure(figsize=(12, 6))
plt.plot(infosys_data['Date'], infosys_data['Daily Return'], label='Infosys', color='blue')
plt.plot(tcs_data['Date'], tcs_data['Daily Return'], label='TCS', color='green')
plt.xlabel('Date')
plt.ylabel('Daily Return')
plt.title('Daily Returns Over Time')
plt.legend()
plt.show()
```

In [12]:
```
from scipy import stats

# Perform t-test for daily returns
t_stat, p_value = stats.ttest_ind(infosys_data['Daily Return'].dropna(), tcs_data['Daily Ret

# Print t-statistic and p-value
print("T-statistic:", t_stat)
print("P-value:", p_value)
```

```
T-statistic: -0.40466223677997676
P-value: 0.6859023881446381
```

In [13]:
```
# Interpret the results
if p_value < 0.05:
    print("There is a significant difference between the daily returns of Infosys and TCS.")
else:
    print("There is no significant difference between the daily returns of Infosys and TCS."
```

```
There is no significant difference between the daily returns of Infosys and TCS.
```

In [14]:
```
# Create a DataFrame to store the findings
findings_data = pd.DataFrame({
    'Metric': ['Closing Prices', 'Volume of Stocks Traded', 'Daily Returns'],
    'Infosys': [infosys_summary.loc['mean', 'Close'], infosys_summary.loc['mean', 'Volume'],
    'TCS': [tcs_summary.loc['mean', 'Close'], tcs_summary.loc['mean', 'Volume'], tcs_data['D
})
findings_data
```

Out[14]:

	Metric	Infosys	TCS
0	Closing Prices	1.439657e+03	3.512582e+03
1	Volume of Stocks Traded	6.635708e+06	2.043061e+06
2	Daily Returns	6.922902e-04	1.194668e-03

In [15]:
```
# Export the findings to an Excel file to import it to tableau public
findings_data.to_excel("C:/Users/HP/Desktop/comparision_model/outputComparison_Module_Results
```

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]: