

FINAL REPORT

Data Science for Finance

XUEJIAO XIA KEYAO FANG YAOZHU WU HAO LIN

CHAOJUNG HUANG JINGWEN REN HUAQING HU SHANWEN TANG

Background

Industry	Internet Retail
Sector	Consumer Discretionary

Here's what we'll cover:

- Alibaba Group Holding Limited (**BABA**)
- Amazon.com, Inc. (**AMZN**)
- JD.com, Inc. (**JD**)
- eBay Inc. (**EBAY**)

COMPANY PROFILE

Alibaba Group Holding Limited



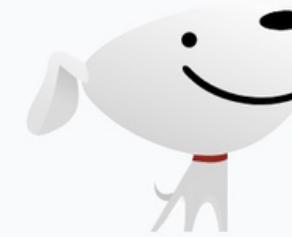
Country	China
Founded	1999
IPO Date	Sep 19, 2014
Industry	<u>Internet Retail</u>
Sector	<u>Consumer Discretionary</u>
Employees	239,740
CEO	Yong Zhang

Amazon.com, Inc.



Country	United States
Founded	1994
IPO Date	May 15, 1997
Industry	<u>Internet Retail</u>
Sector	<u>Consumer Discretionary</u>
Employees	<u>1,541,000</u>
CEO	Andy Jassy

JD.com, Inc.



Country	China
IPO Date	May 22, 2014
Industry	<u>Internet Retail</u>
Sector	<u>Consumer Discretionary</u>
Employees	385,357
CEO	Qiangdong Liu

eBay Inc.



Country	United States
Founded	1995
IPO Date	Sep 24, 1998
Industry	<u>Internet Retail</u>
Sector	<u>Consumer Discretionary</u>
Employees	11,600
CEO	James Iannone

Analytical Element

Profitability ratio

Liquidity ratio

Valuation ratio

Stock returns & Average annualized returns

Time series

Technical indicator

Import DATA

```
import pandas as pd
import yfinance as yf
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
import talib
brew install ta-lib
conda install -c conda-forge ta-lib
```

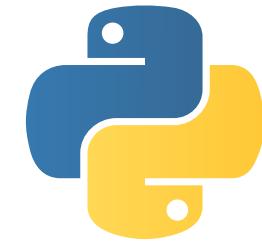
```
##### PART 1 #####
# import the financial statements data for each company over 5 years
path = '/Users/francismac/Desktop/NEOMA/S2/Data Science for Finance/'

BABA_income = pd.read_excel(path+'BABA_IS.xlsx', index_col=0)
BABA_balance = pd.read_excel(path+'BABA_BS.xlsx', index_col=0)
BABA_cash = pd.read_excel(path+'BABA_CF.xlsx', index_col=0)
BABA = pd.concat([BABA_income,BABA_balance,BABA_cash]).transpose()
BABA = BABA.loc[:, ~BABA.columns.duplicated()]

AMZN_income = pd.read_excel(path+'AMZN_IS.xlsx', index_col=0)
AMZN_balance = pd.read_excel(path+'AMZN_BS.xlsx', index_col=0)
AMZN_cash = pd.read_excel(path+'AMZN_CF.xlsx', index_col=0)
AMZN = pd.concat([AMZN_income,AMZN_balance,AMZN_cash]).transpose()
AMZN = AMZN.loc[:, ~AMZN.columns.duplicated()]

JD_income = pd.read_excel(path+'JD_IS.xlsx', index_col=0)
JD_balance = pd.read_excel(path+'JD_BS.xlsx', index_col=0)
JD_cash = pd.read_excel(path+'JD_CF.xlsx', index_col=0)
JD = pd.concat([JD_income,JD_balance,JD_cash]).transpose()
JD = JD.loc[:, ~JD.columns.duplicated()]

EBAY_income = pd.read_excel(path+'EBAY_IS.xlsx', index_col=0)
EBAY_balance = pd.read_excel(path+'EBAY_BS.xlsx', index_col=0)
EBAY_cash = pd.read_excel(path+'EBAY_CF.xlsx', index_col=0)
EBAY = pd.concat([EBAY_income,EBAY_balance,EBAY_cash]).transpose()
EBAY = EBAY.loc[:, ~EBAY.columns.duplicated()]
```



Profitability ratio

Return on Equity (ROE)

$ROE = (\text{Net Income} / \text{Average Shareholders' Equity}) * 100\%$

$\text{Net Income} = \text{Revenue} - \text{All Expenses}$

$\text{Shareholders' Equity} = \text{Total assets} - \text{Total liabilities}$

Return on Assets (ROA)

$ROA = (\text{Net Income} / \text{Average Total Assets}) * 100\%$

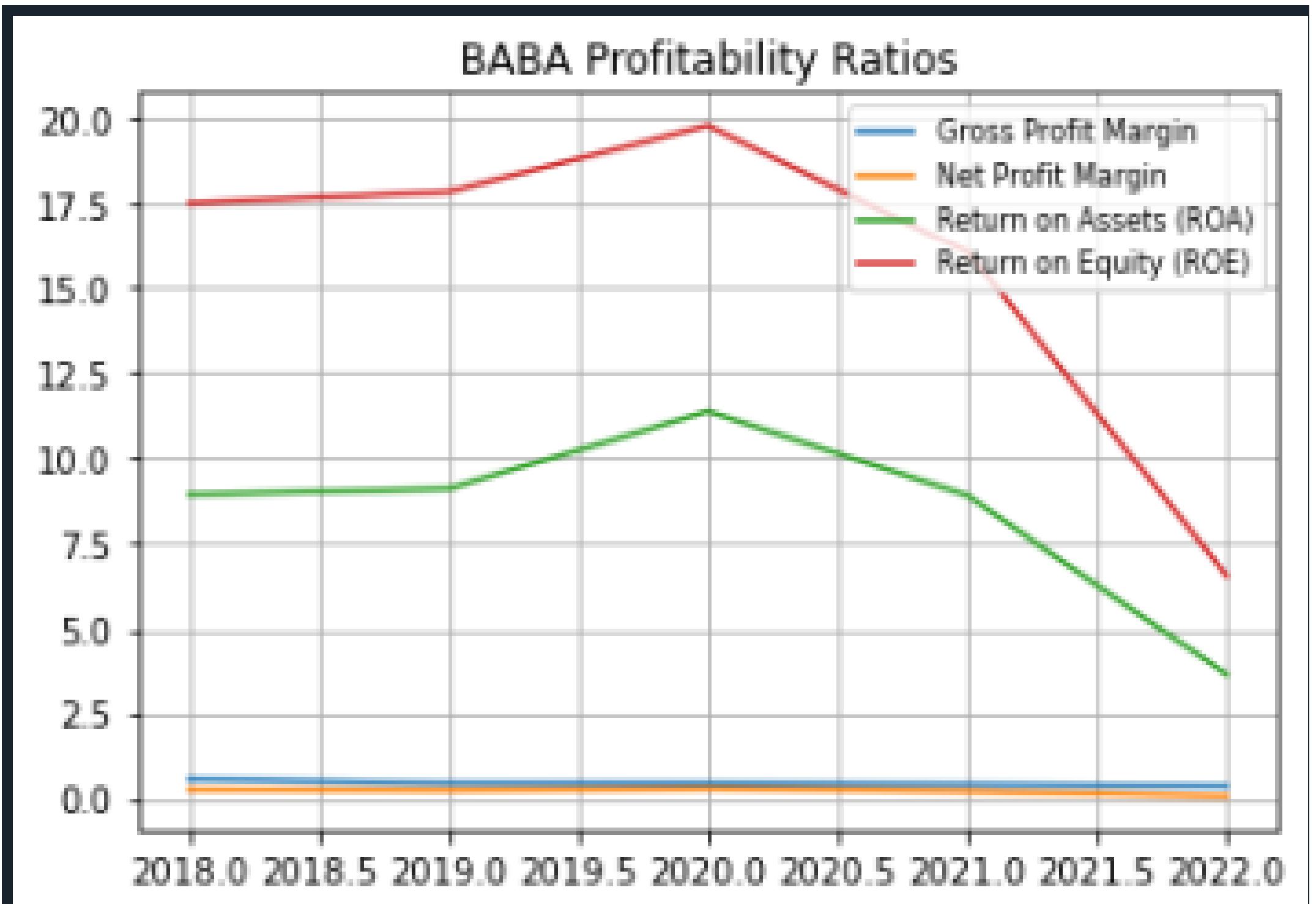
Gross profit

$\text{Gross Profit} = \text{Revenue} - \text{Cost of Revenue}$

Net Profit Margin

$\text{Profit Margin} = (\text{Net Income} / \text{Revenue}) * 100\%$

ALI BABA



```
fig1 = plt.figure(figsize=(8, 5))
BABA[BABA_profitability_ratios].plot(title='BABA Profitability Ratios', grid=True)
plt.legend(loc='upper right', fontsize=8,)
```

```
# calculate Profitability Ratios
BABA['Gross Profit Margin'] = BABA['Gross Profit']/BABA['Revenue']
BABA['Net Profit Margin'] = BABA['Net Income']/BABA['Revenue']
BABA['Return on Assets (ROA)'] = BABA['Net Income']/BABA['Total Assets']*100
BABA['Return on Equity (ROE)'] = BABA['Net Income']/BABA["Shareholders' Equity"]*100
```

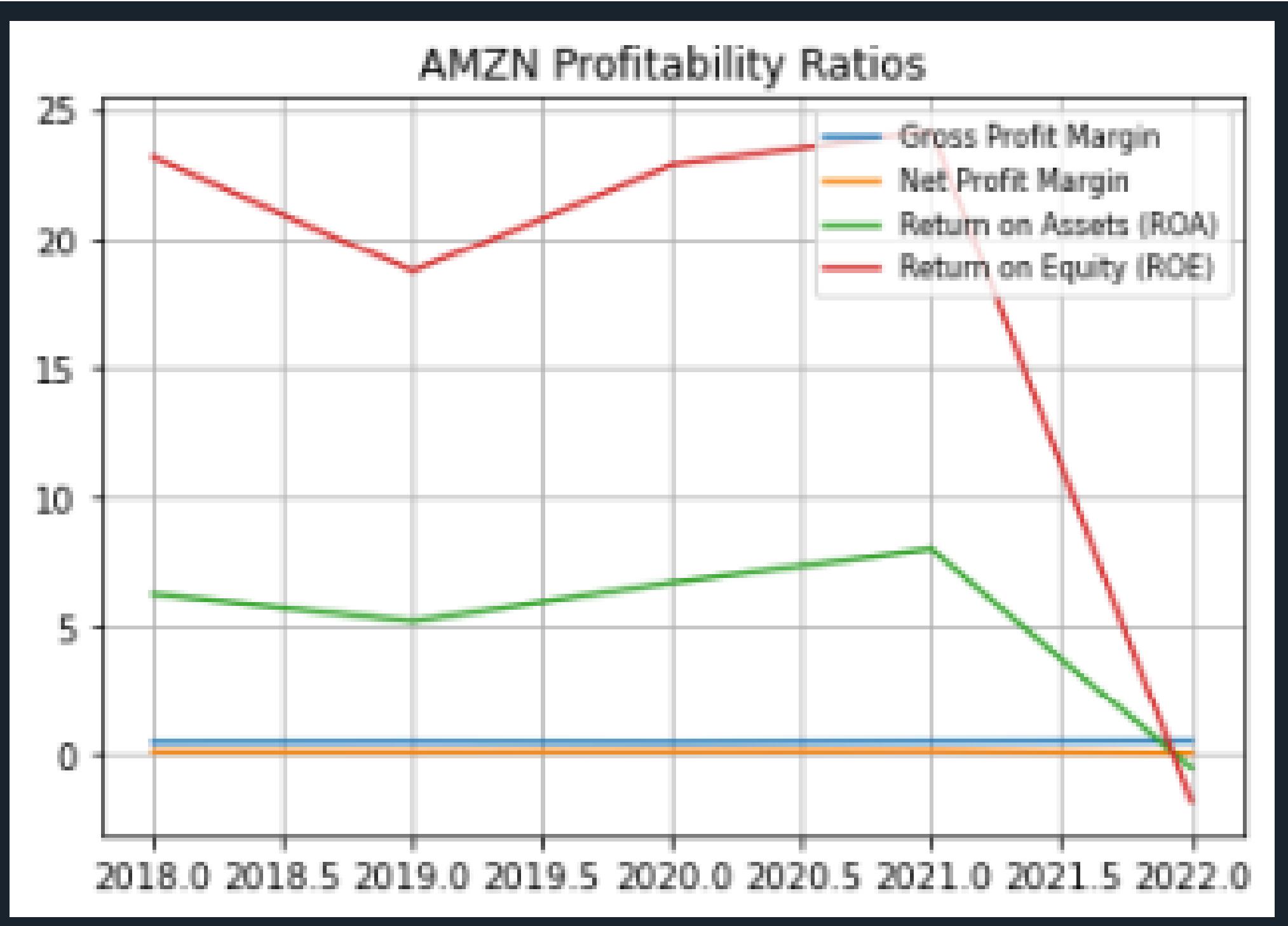
2018~2020 ↑

- Chinese e-commerce market.
- massive scale
- extensive reach

2020~2022 ↓

- competition in the Chinese e-commerce market
- changes in consumer behavior
- COVID-19 pandemic on the global economy.

AMAZON



2019~2020 ↑

- COVID-19 pandemic on the global economy.
- Demande increase.

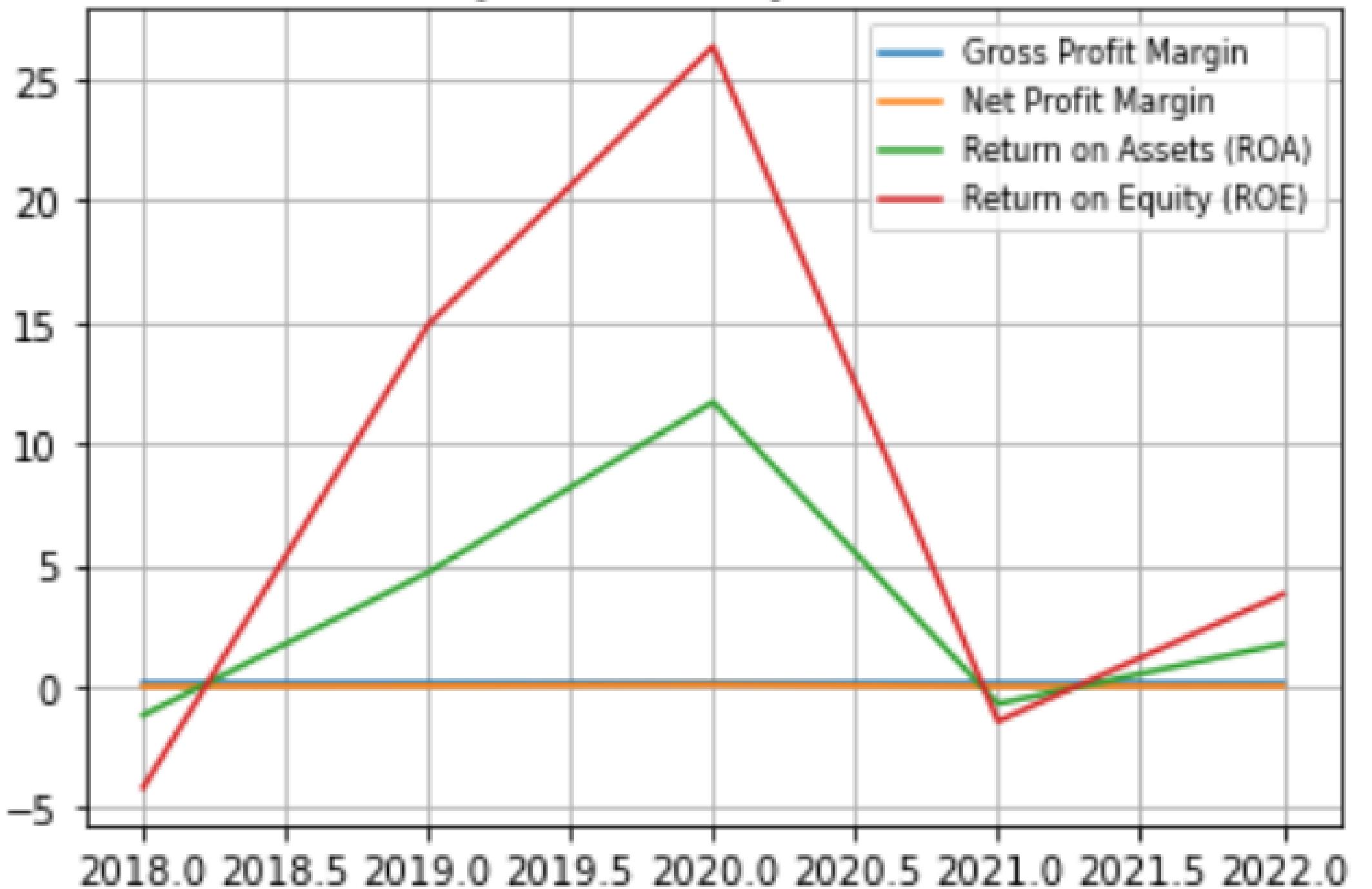
2020~2022 ↑

- Economy reopens
- Consumers begin to head back to physical stores

```
fig4 = plt.figure(figsize=(8, 5))
AMZN[AMZN_profitability_ratios].plot(title='AMZN Profitability Ratios', grid=True)
plt.legend(loc='upper right', fontsize=8,)
```

```
# calculate Profitability Ratios
AMZN['Gross Profit Margin'] = AMZN['Gross Profit']/AMZN['Revenue']
AMZN['Net Profit Margin'] = AMZN['Net Income']/AMZN['Revenue']
AMZN['Return on Assets (ROA)'] = AMZN['Net Income']/AMZN['Total Assets']*100
AMZN['Return on Equity (ROE)'] = AMZN['Net Income']/AMZN["Shareholders' Equity"]*100
```

JD Profitability Ratios



JD

2018~2020 ↑

- Before the COVID-19, the Return on equity(ROE) increased progressively each year, represented an increase efficiency in making profits of the company

2020~2021 ↓

- Global economic grew slowly due to the Covid-19 pandemic effects and administrative control, both the Return on assets(ROA) and ROE declined during this period

2021~2022 ↑

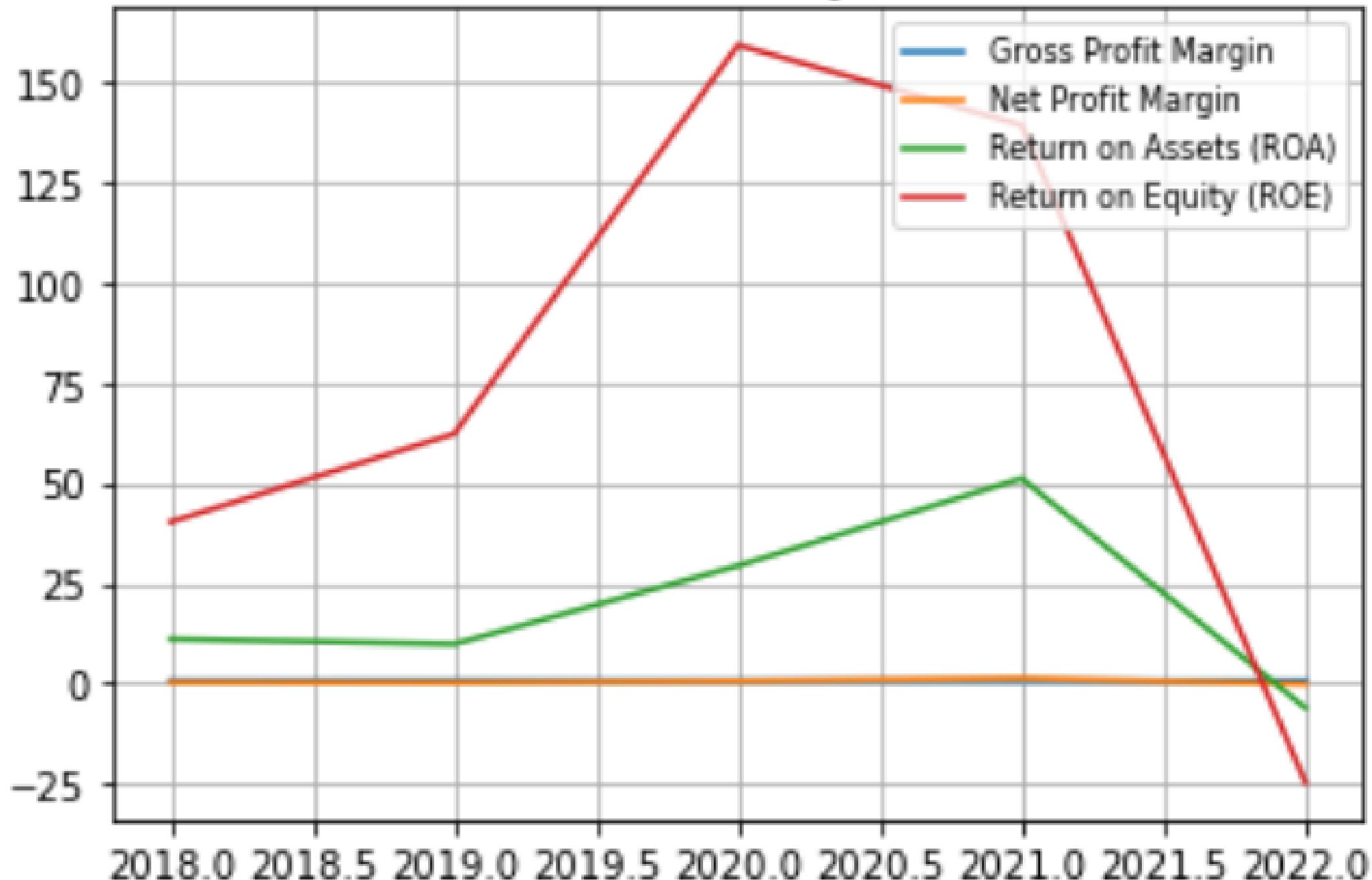
- Customers' demands bounces back under a regular control of the pandemic covid-19

```
# calculate Profitability Ratios
JD['Gross Profit Margin'] = JD['Gross Profit']/JD['Revenue']
JD['Net Profit Margin'] = JD['Net Income']/JD['Revenue']
JD['Return on Assets (ROA)'] = JD['Net Income']/JD['Total Assets']*100
JD['Return on Equity (ROE)'] = JD['Net Income']/JD["Shareholders' Equity"]*100

JD_profitability_ratios = ['Gross Profit Margin', 'Net Profit Margin', 'Return on Assets (ROA)', 'Return on Equity (ROE)']

# Create a figure for the first set of ratios
fig7 = plt.figure(figsize=(8, 5))
JD[JD_profitability_ratios].plot(title='JD Profitability Ratios', grid=True)
plt.legend(loc='upper right', fontsize=8)
# Show the plots
plt.show()
```

EBAY Profitability Ratios



```
# calculate Profitability Ratios
EBAY['Gross Profit Margin'] = EBAY['Gross Profit']/EBAY['Revenue']
EBAY['Net Profit Margin'] = EBAY['Net Income']/EBAY['Revenue']
EBAY['Return on Assets (ROA)'] = EBAY['Net Income']/EBAY['Total Assets']*100
EBAY['Return on Equity (ROE)'] = EBAY['Net Income']/EBAY["Shareholders' Equity"]*100
```

```
EBAY_profitability_ratios = ['Gross Profit Margin', 'Net Profit Margin', 'Return on Assets (ROA)', 'Return on Equity (ROE)']
# Create a figure for the first set of ratios
fig10 = plt.figure(figsize=(8, 5))
EBAY[EBAY_profitability_ratios].plot(title='EBAY Profitability Ratios', grid=True)
plt.legend(loc='upper right', fontsize=8, )
# Show the plots
plt.show()
```

EBAY

2019~2020 ↑

2021~2022 ↓

- ROE increased sharply from 2018 to 2020, and decreased after 2020, ROA increased smoothly before 2021, this implied a high financial leverage in this company
- Total ROE and ROA decreased after 2021, while the proportion of debt in assets augments

Liquidity ratio

Current Ratio

Current Ratio = Total Current Assets / Total Current Liabilities

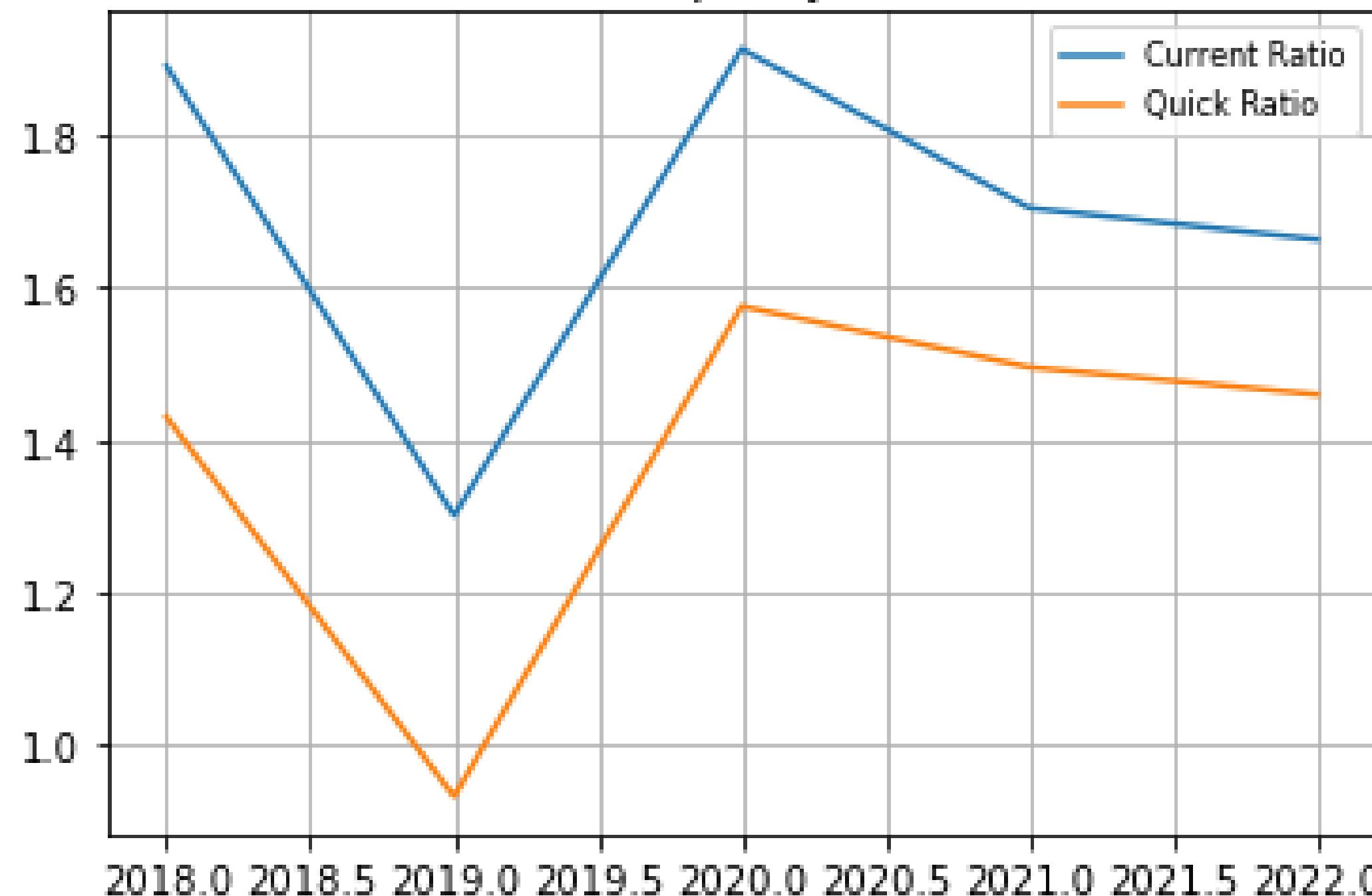
Quick Ratio

Quick Ratio = Total Current Assets - Inventory / Total Current Liabilities

```
# calculate Liquidity Ratios
BABA['Current Ratio'] = BABA['Total Current Assets']/BABA['Total Current Liabilities']
BABA['Quick Ratio'] = (BABA['Total Current Assets']-BABA['Inventory'])/BABA['Total Current Liabilities']
```

ALI BABA

BABA Liquidity Ratios



```
BABA_liquidity_ratios = ['Current Ratio', 'Quick Ratio']
```

```
# Create a figure for the second set of ratios
fig2 = plt.figure(figsize=(8, 5))
BABA[BABA_liquidity_ratios].plot(title='BABA Liquidity Ratios', grid=True)
plt.legend(loc='upper right', fontsize=8)
```

2018~2019 ↓

- increase in inventory

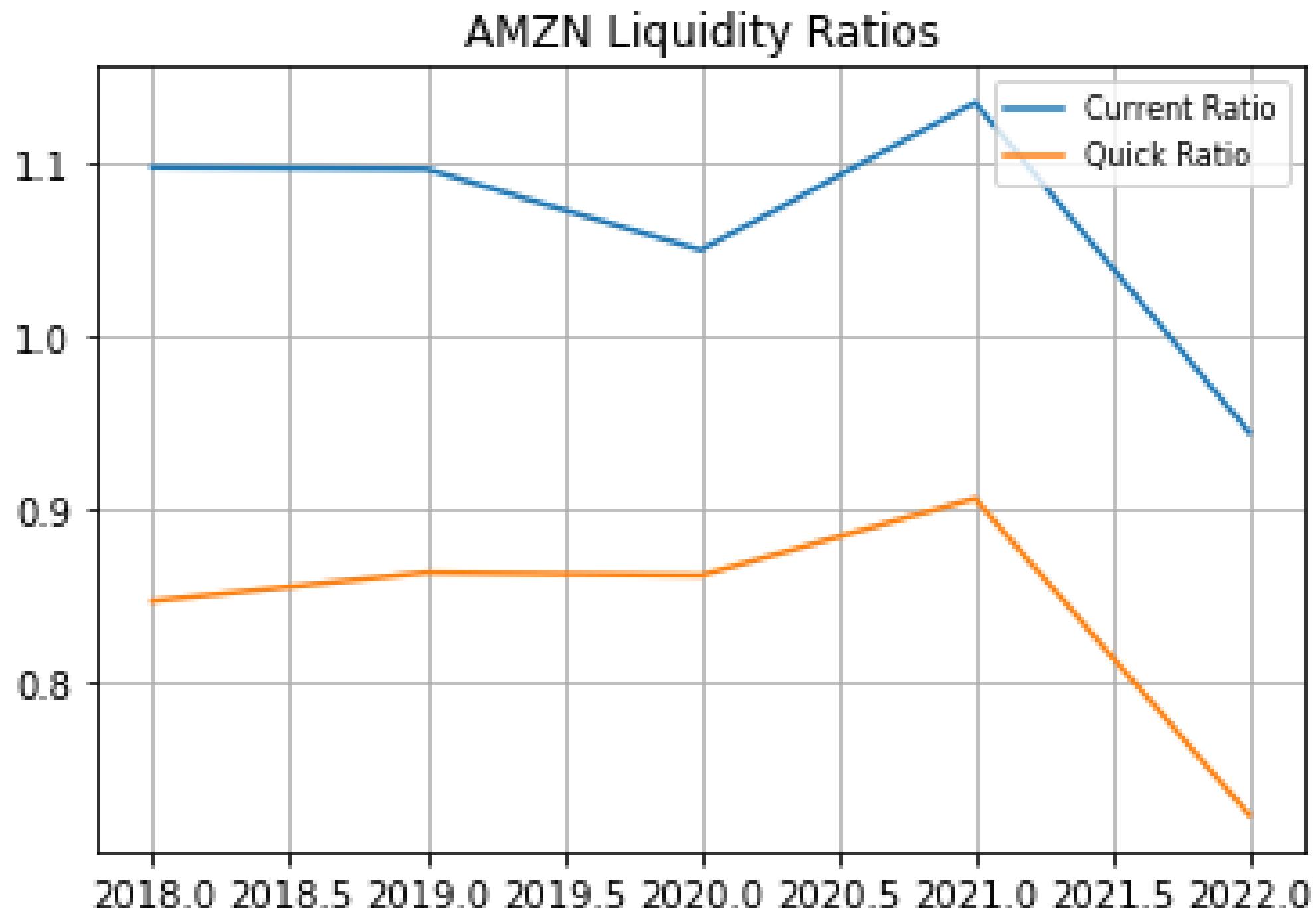
2019~2020 ↑

- increase in the company's cash and cash equivalents
- decrease in inventory and short-term debt

2020~2022 ↓

- increase in short-term debt or a decrease in current assets
- the impact of the COVID-19 pandemic

AMAZON



2018~2021 ↓ ↑

- Operating cash flow
- Changes in inventory
- Changes in short-term debt

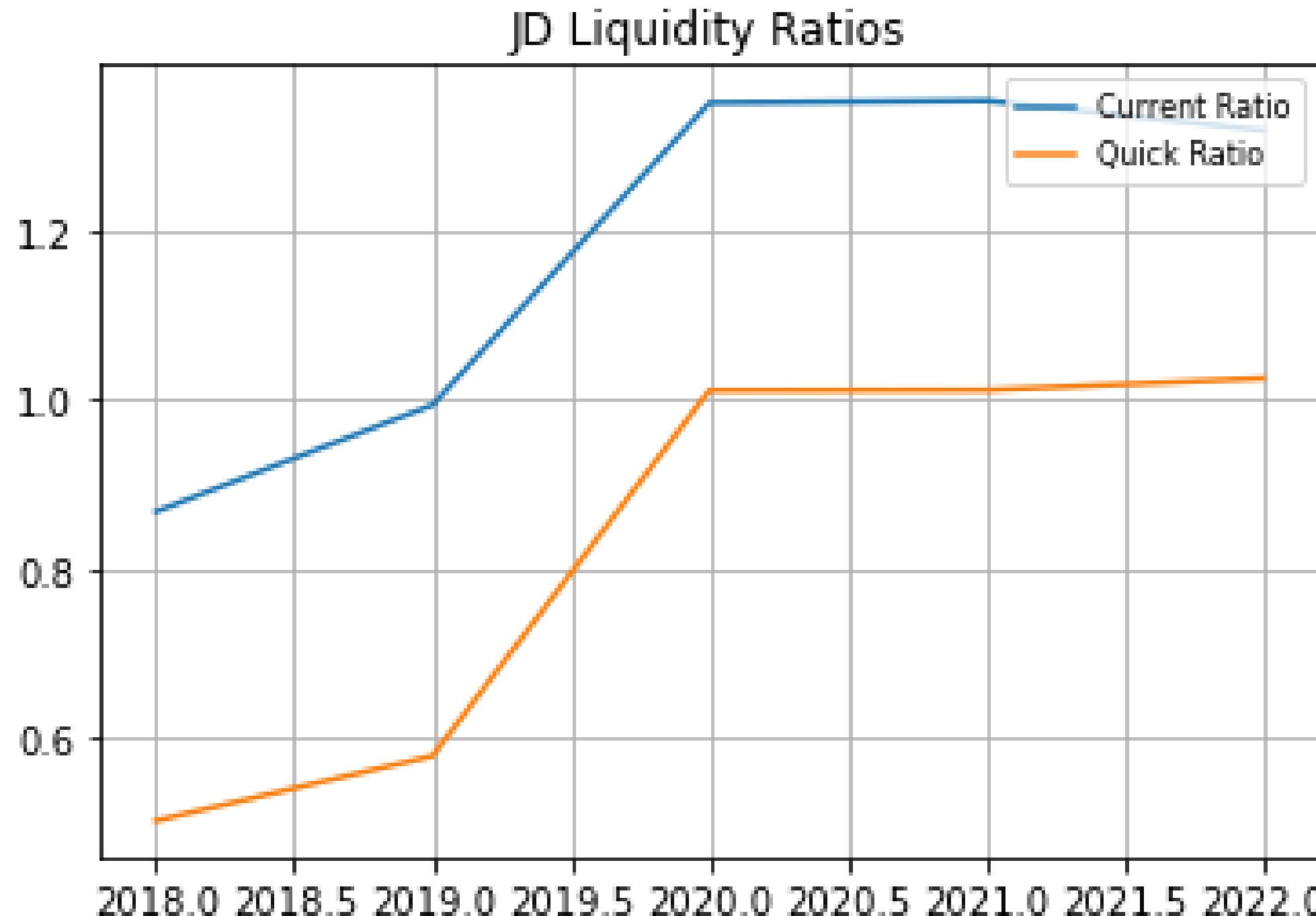
2021~2022 ↓

- Supply chain issues
- Investment activities
- Slowdown in revenue growth

```
AMZN_liquidity_ratios = ['Current Ratio', 'Quick Ratio']
```

```
# Create a figure for the second set of ratios
fig5 = plt.figure(figsize=(8, 5))
AMZN[AMZN_liquidity_ratios].plot(title='AMZN Liquidity Ratios', grid=True)
plt.legend(loc='upper right', fontsize=8)
```

JD



2018~2020 ↑

- reduce the inventory
- capital expenditures
- E-commerce market growth

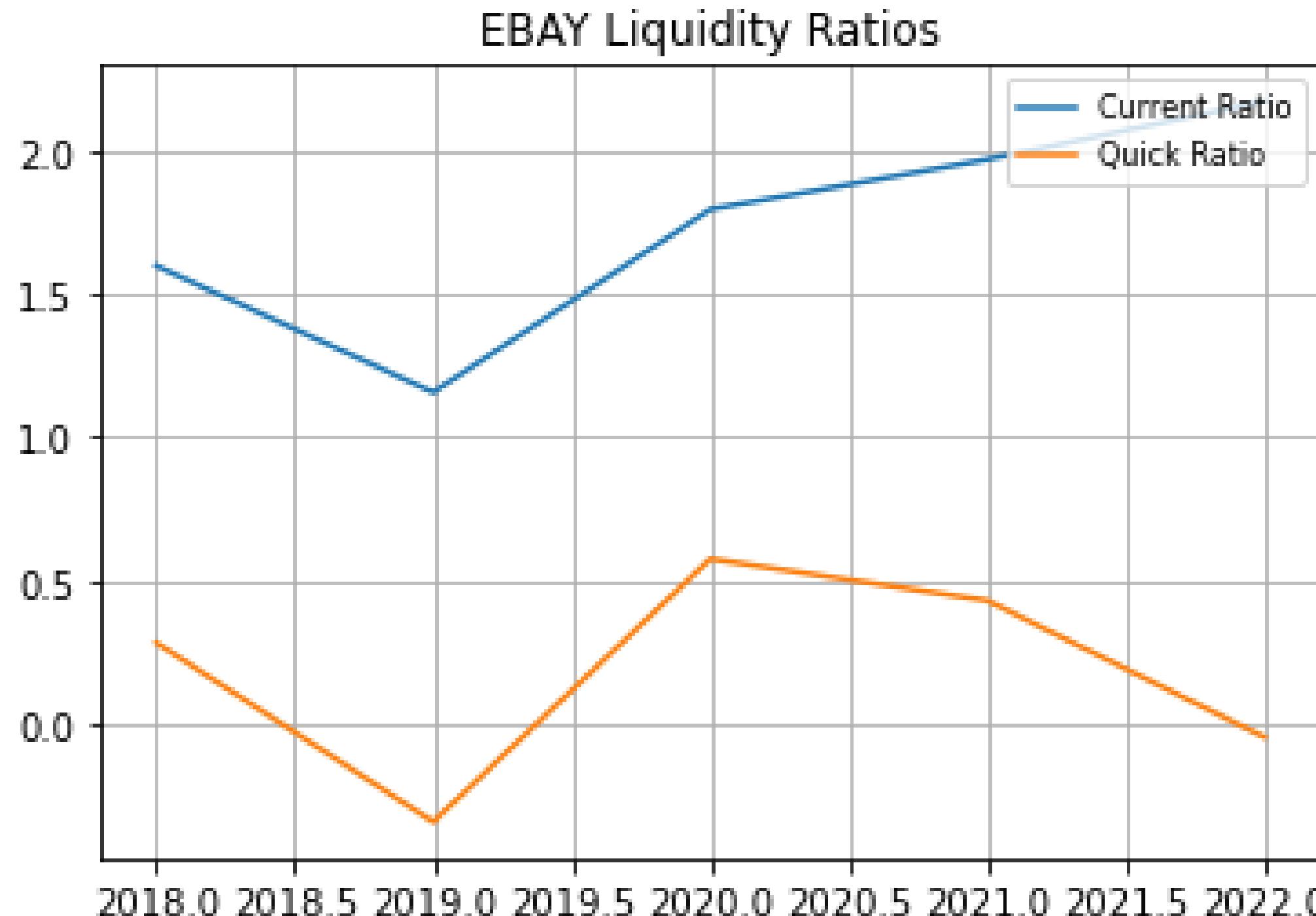
2020~2022–

- impact of COVID-19
- market competition
- debt level

```
JD_liquidity_ratios = ['Current Ratio', 'Quick Ratio']
```

```
# Create a figure for the second set of ratios
fig8 = plt.figure(figsize=(8, 5))
JD[JD_liquidity_ratios].plot(title='JD Liquidity Ratios', grid=True)
plt.legend(loc='upper right', fontsize=8)
```

EBAY



2018~2019 ↓

- Changes in business model
- economic conditions

2019~2022 ↑ / 2020~2022 ↓

- changes in inventory management
- capital expenditure
- market competition

```
EBAY liquidity ratios = ['Current Ratio', 'Quick Ratio']
```

```
# Create a figure for the second set of ratios
fig11 = plt.figure(figsize=(8, 5))
EBAY[EBAY_liquidity_ratios].plot(title='EBAY Liquidity Ratios', grid=True)
plt.legend(loc='upper right', fontsize=8)
```

Valuation ratio

Price to Earnings Ratio (P/E)

P/E Ratio= Market value (per share)/ Earnings (per share)
EPS = (Net income - Preferred dividends)/Average outstanding common shares

Price to Book Ratio (P/B)

P/B = Market Capitalisation /Shareholders' Equity

EV/EBIT Ratio

EV/EBIT = Enterprise value/ Earnings before interest and taxes

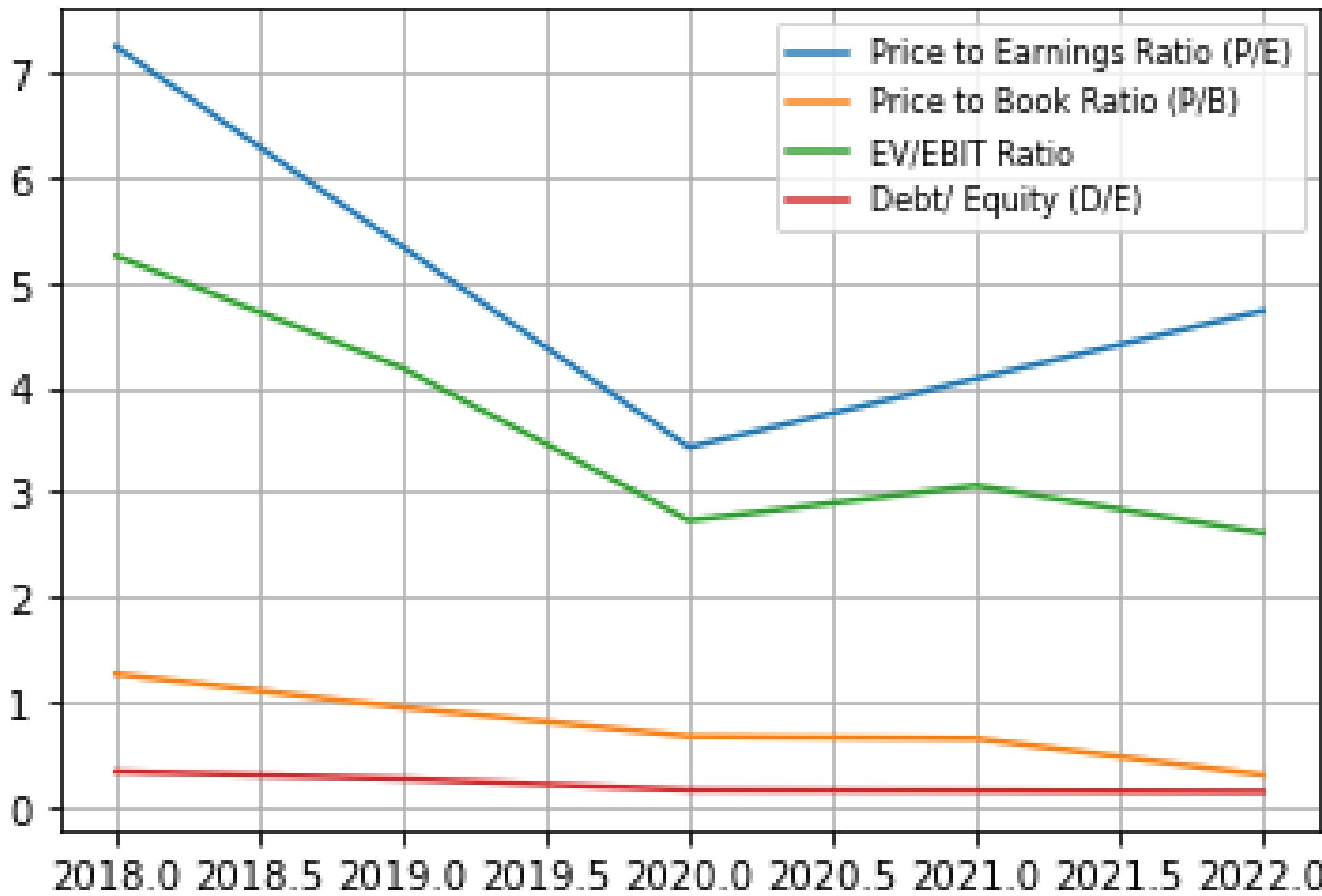
EV = Equity Market Capitalization + Total Debt – Cash (& Cash Equivalents)

Debt/ Equity (D/E)

D/E= Total Liabilities/ Total shareholders' Equity

```
# Calculate valuation ratios
BABA['Price to Earnings Ratio (P/E)'] = BABA['Market Cap'] / BABA['Net Income']
BABA['Price to Book Ratio (P/B)'] = (BABA['Market Cap'] / BABA["Shareholders' Equity"])
BABA['EV/EBIT Ratio'] = BABA['Enterprise Value'] /BABA['EBIT']
BABA['Debt/ Equity (D/E)'] = BABA['Total Debt'] / BABA["Shareholders' Equity"]
```

BABA Valuation Ratios



ALI BABA

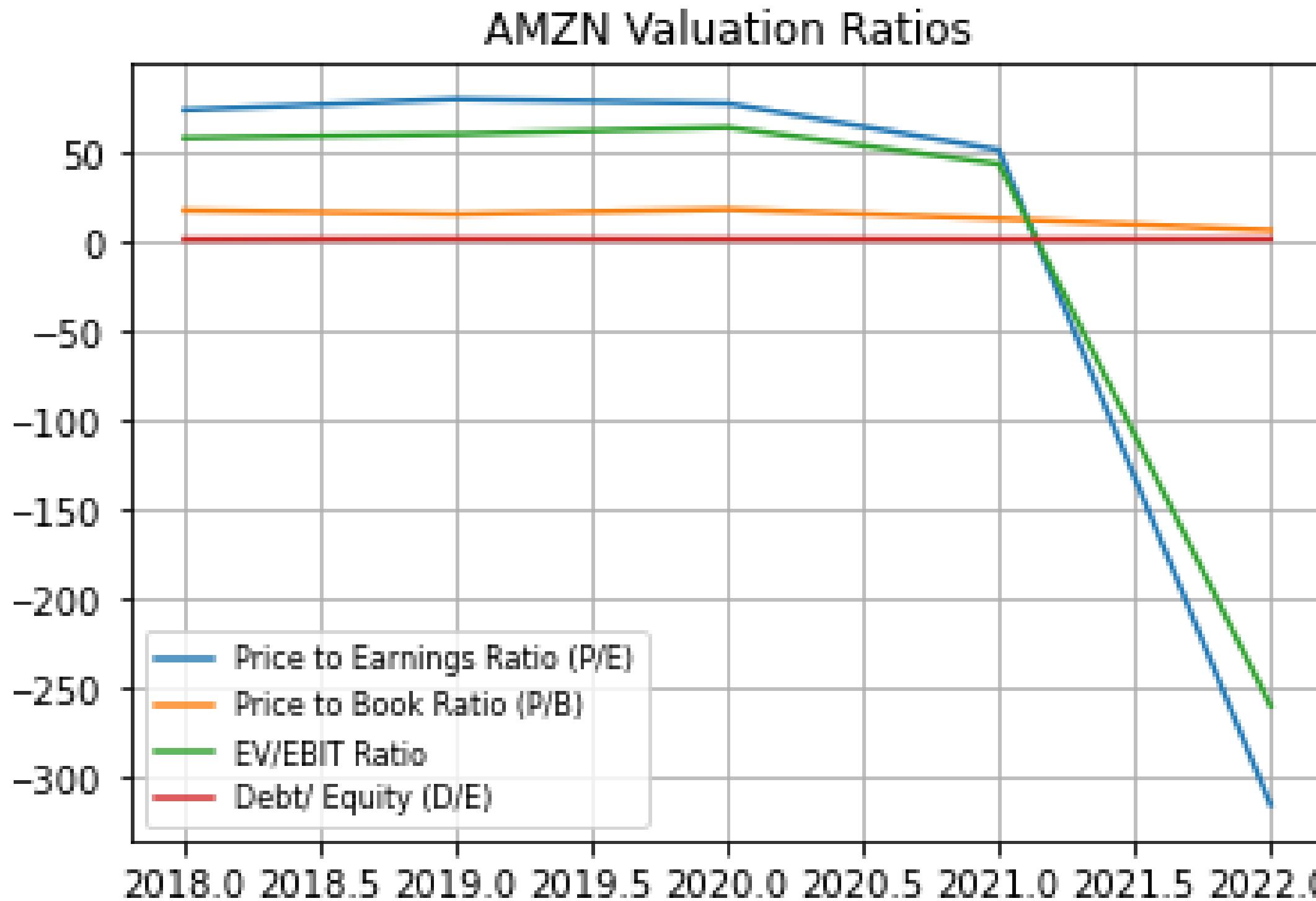
- **P/E ratio** has decreased until 2020: attributed to the COVID-19 pandemic.
- **P/B ratio** has also fluctuated over the past five years.
- **EV/EBIT ratio** decreased until 2020 then fluctuated.
- **D/E: ratio** has remained relatively stable over the past five years.
- **Total Debt** almost increases every year
- **Shareholders' Equity** grows from 2018 to 2022

```
BABA_valuation_ratios = ['Price to Earnings Ratio (P/E)', 'Price to Book Ratio (P/B)', 'EV/EBIT Ratio', 'Debt/ Equity (D/E)']

# Create a figure for the third set of ratios
fig3 = plt.figure(figsize=(8, 5))
BABA[BABA_valuation_ratios].plot(title='BABA Valuation Ratios', grid=True)
plt.legend(loc='upper right', fontsize=8)

# Show the plots
plt.show()
```

AMAZON



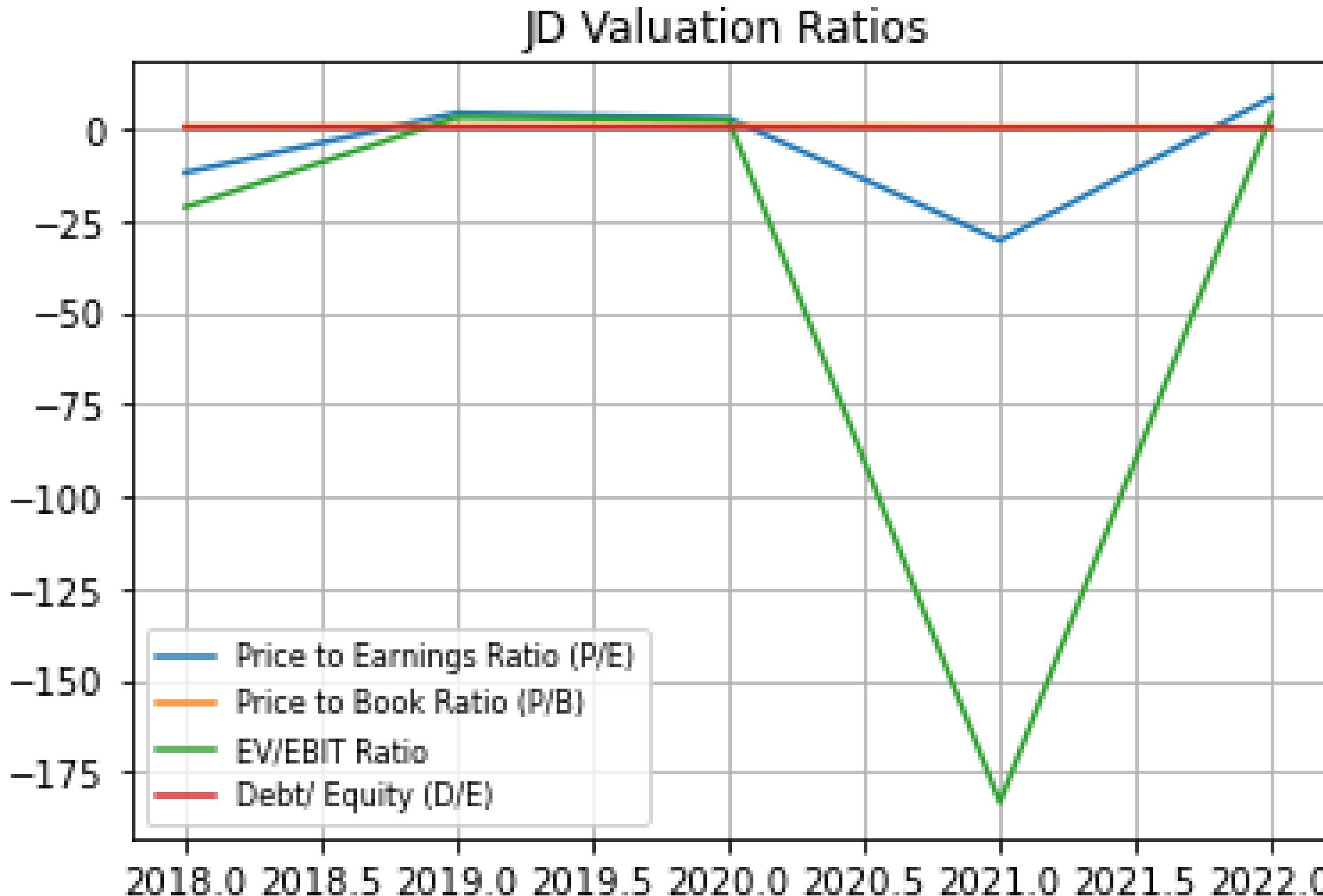
- **P/E ratio and EV/EBIT ratio** fall dramatically in 2022 due to: negative net income.
- **P/B ratio:** Overall, there has been a downward trend.(from 16.86 in 2018, to 5.87 in 2022)
- **D/E ratio** has fluctuated over the past five years. (ranging from 0.76 in 2018 to 0.96 in 2022.)
- **Total Debt** has tripled in 5 years

```
AMZN_valuation_ratios = ['Price to Earnings Ratio (P/E)', 'Price to Book Ratio (P/B)', 'EV/EBIT Ratio', 'Debt/ Equity (D/E)']
```

```
# Create a figure for the third set of ratios
fig6 = plt.figure(figsize=(8, 5))
AMZN[AMZN_valuation_ratios].plot(title='AMZN Valuation Ratios', grid=True)
plt.legend(loc='lower left', fontsize=8)

# Show the plots
plt.show()
```

JD



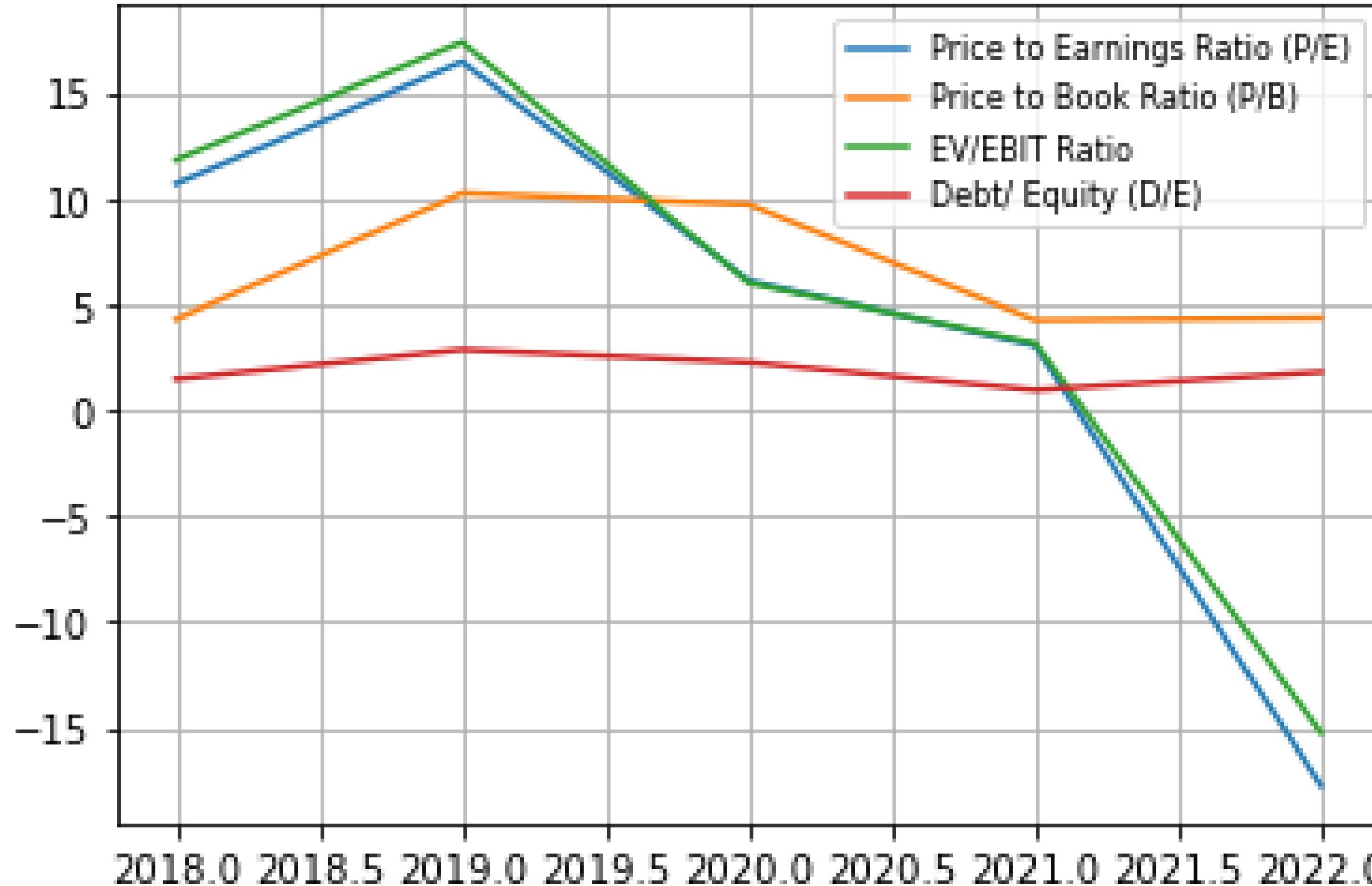
- **P/E ratio and P/B ratio** have fluctuated over the past five years.
- **EV/EBIT ratio** fall dramatically in 2021 due to the negative income in 2021.
- **D/E ratio** has fluctuated over the past five years. (ranging from 0.14 in 2021, to 0.24 in both 2018 and 2022.)
- **Total Debt** has increased 5 times in 5 years.

```
JD_valuation_ratios = ['Price to Earnings Ratio (P/E)', 'Price to Book Ratio (P/B)', 'EV/EBIT Ratio', 'Debt/ Equity (D/E)']
```

```
# Create a figure for the third set of ratios
fig9 = plt.figure(figsize=(8, 5))
JD[JD_valuation_ratios].plot(title='JD Valuation Ratios', grid=True)
plt.legend(loc='lower left', fontsize=8)

# Show the plots
plt.show()
```

EBAY Valuation Ratios



EBAY

- **P/E ratio and EV/EBIT ratio:** there has been a decreasing trend. Fall dramatically in 2022: negative net income.
- **P/B ratio** has also fluctuated over the past five years.
- **D/E ratio** has remained relatively stable over the past five years, ranging from a high of 2.86 in 2019 to a low of 0.95 in 2021.

```
EBAY_valuation_ratios = ['Price to Earnings Ratio (P/E)', 'Price to Book Ratio (P/B)', 'EV/EBIT Ratio', 'Debt/ Equity (D/E)']

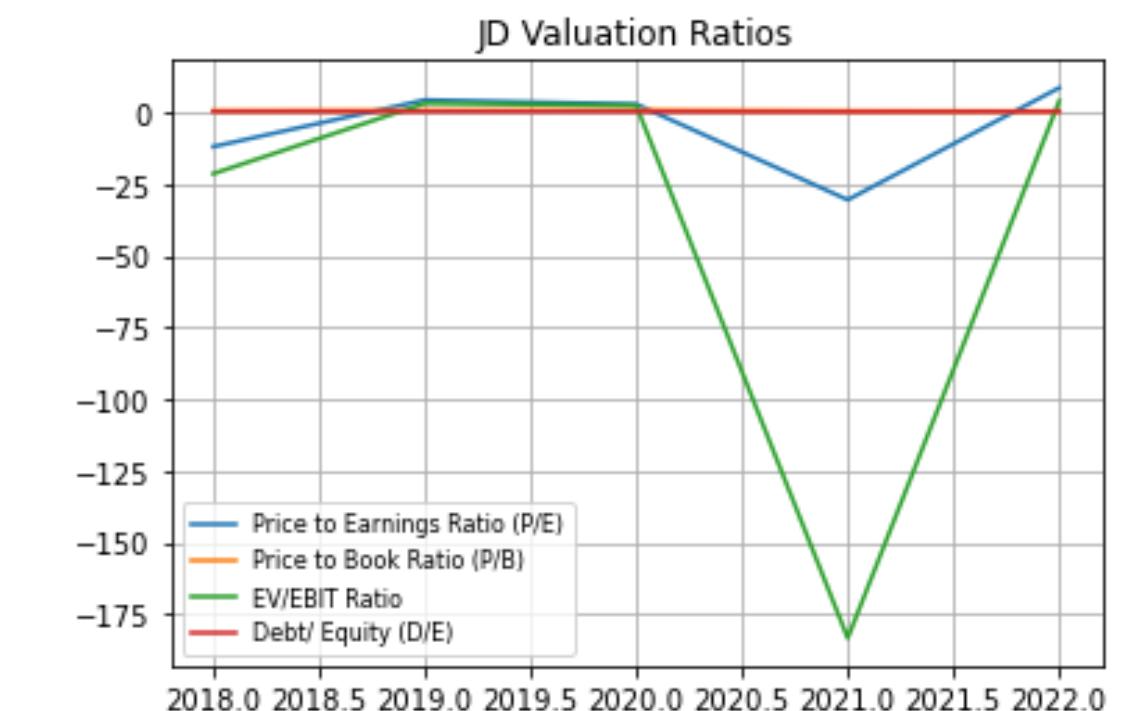
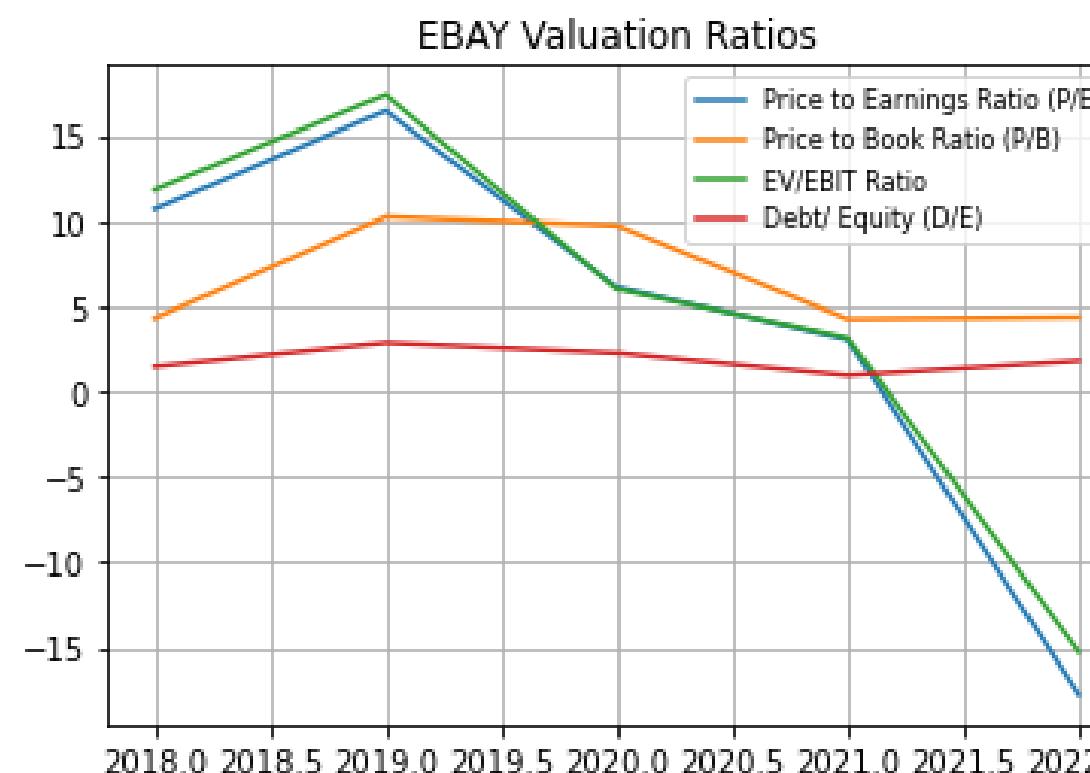
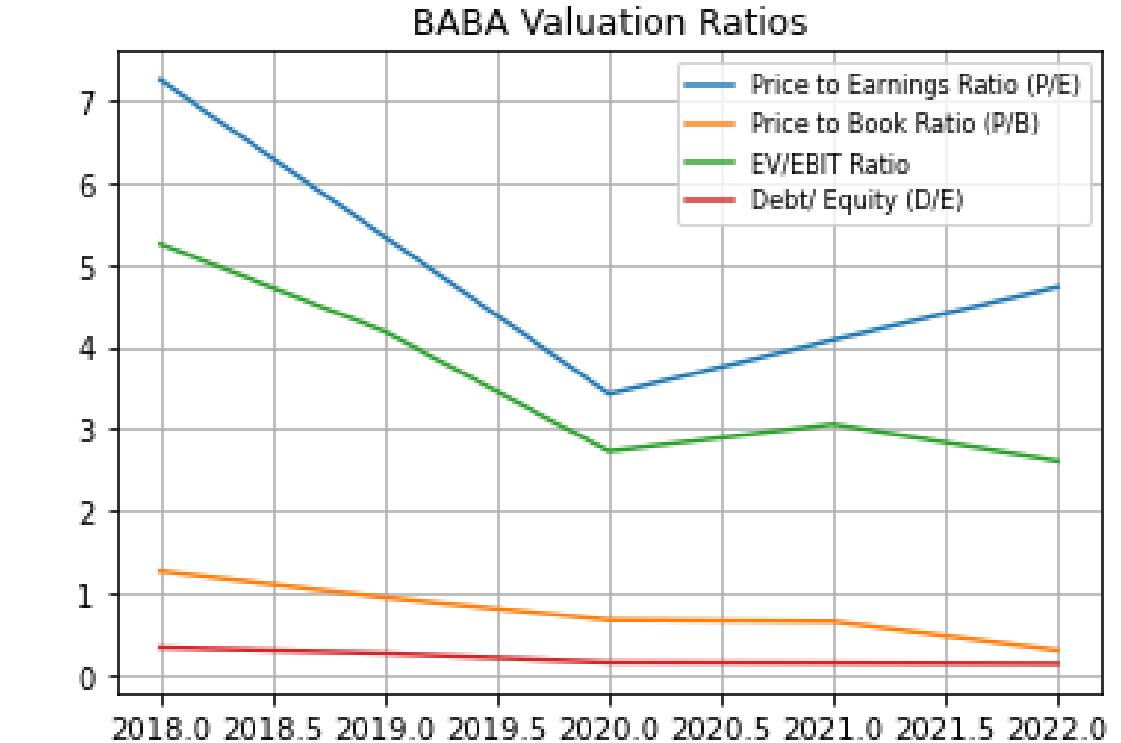
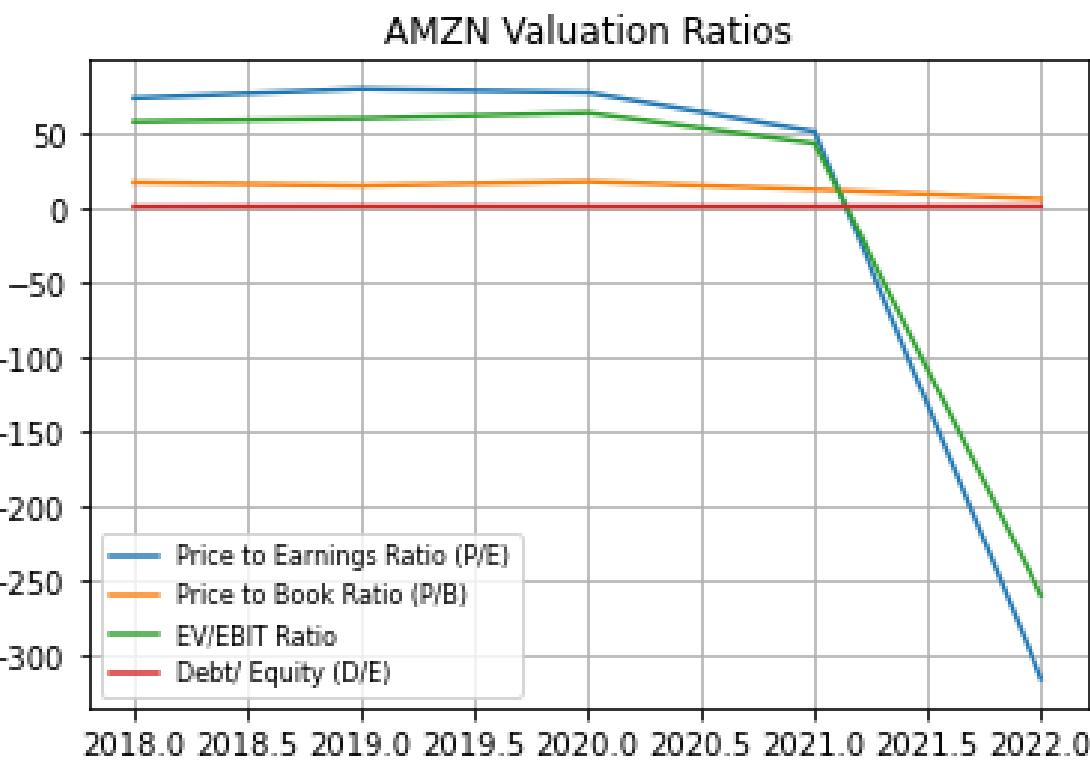
# Create a figure for the third set of ratios
fig12 = plt.figure(figsize=(8, 5))
EBAY[EBAY_valuation_ratios].plot(title='EBAY Valuation Ratios', grid=True)
plt.legend(loc='upper right', fontsize=8)

# Show the plots
plt.show()
```

- **Total Debt** is steady over the past five years.

Analysis in Valuation ratio

- **Amazon** is the largest of the four Internet retail companies by market capitalization, while **Ebay** is the smallest.
- Companies in the Internet sales industry are all down between 2021 and 2022. But **JD** and **BABA** are doing better than Amazon and Ebay in recession period.



Stock Price & Time Series

Stock Price

```
# List of firms
firms = ['BABA', 'AMZN', 'JD', 'EBAY']

# Download stock price data over the past 5 years
start_date = '2018-01-01'
end_date = '2022-12-31'
prices = pd.DataFrame()

for firm in firms:
    data = yf.download(firm, start=start_date, end=end_date, progress=False)
    prices[firm] = data['Adj Close']

# Compute stock returns over the past 5 years
returns = np.log(prices / prices.shift(1))

# Compute average annualized returns
avg_returns = returns.mean() * 252

# Print the firms with the highest and lowest returns
print('Firm with highest return: ', avg_returns.idxmax())
print('Firm with lowest return: ', avg_returns.idxmin())
```

Return

Index	0
BABA	-0.147168
AMZN	0.0692445
JD	0.05707
EBAY	0.0290053

Firm with highest return: AMZN
Firm with lowest return: BABA

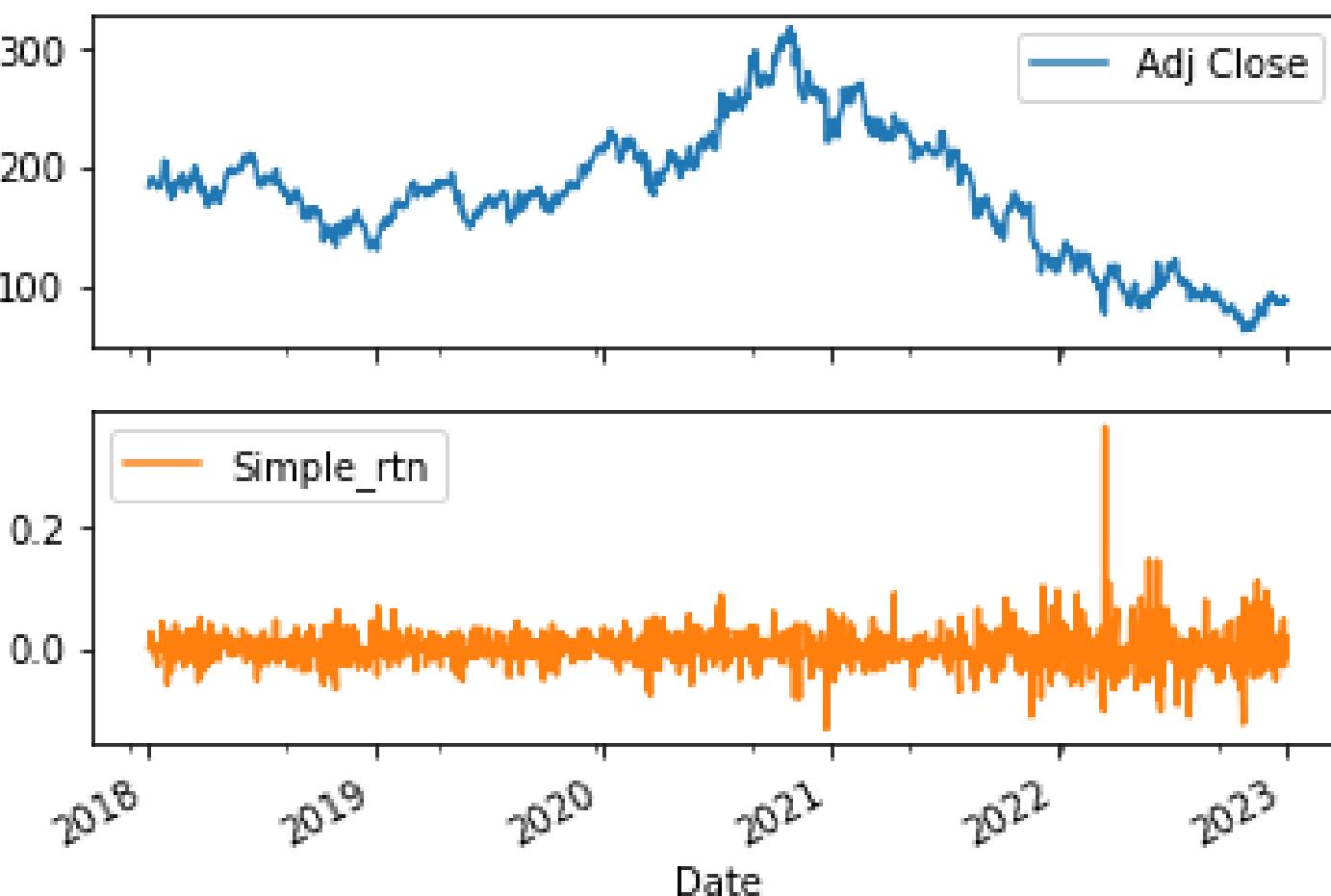
Time Series

```
##### Visualization of time series data #####
# BABA
data_1 = yf.download("BABA",
                     start="2018-01-01",
                     end="2022-12-31",
                     auto_adjust=False,
                     progress=False)

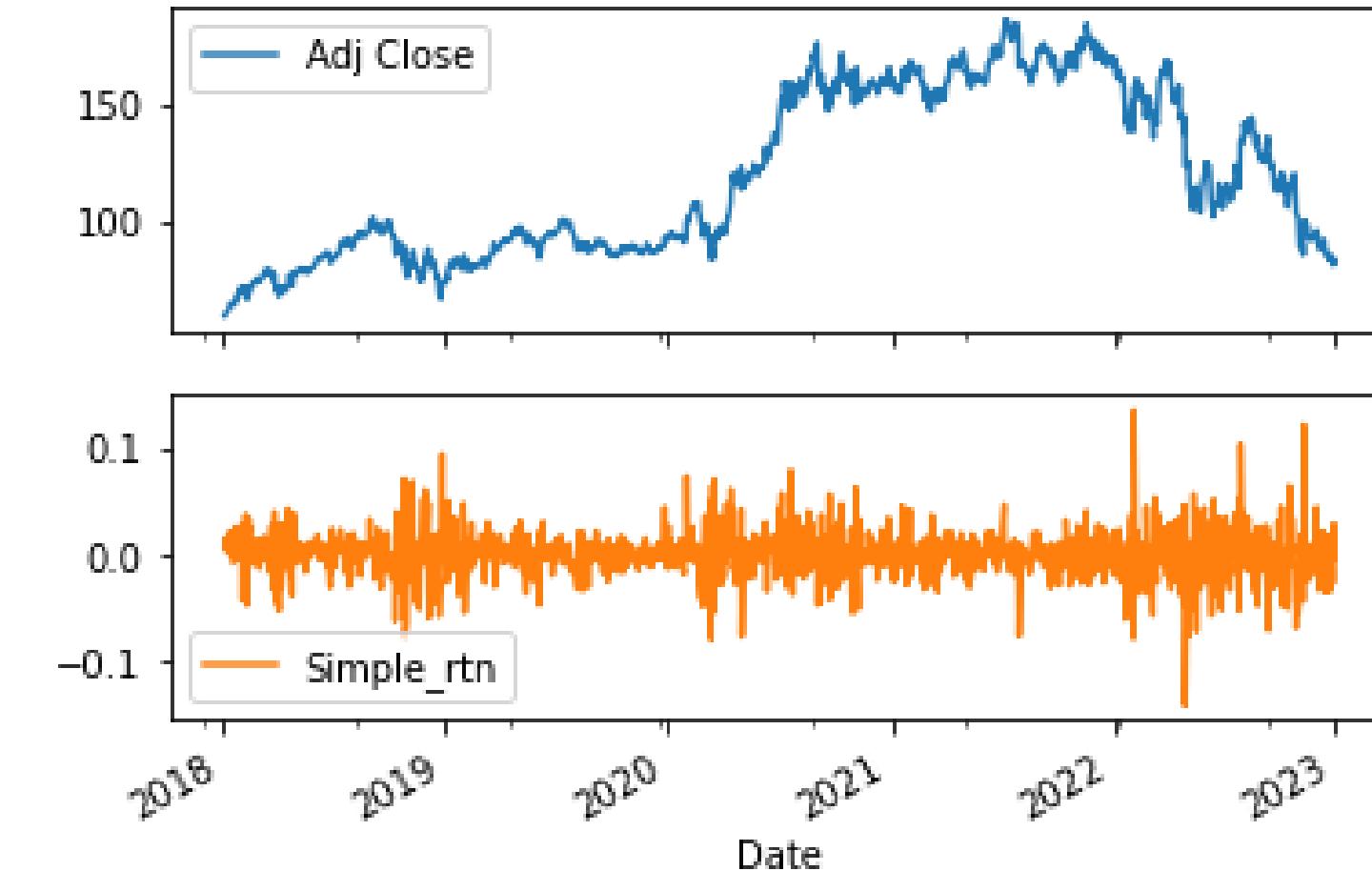
data_1["Simple rtn"] = data_1["Adj Close"].pct_change()
data_1 = data_1.dropna()

data_1["Adj Close"].plot(title = "BABA Time Series 2018-2022")
(
    data_1[["Adj Close", "Simple rtn"]]
    .plot(subplots = True, sharex = True,
          title = "BABA Time Series 2018-2022")
)
```

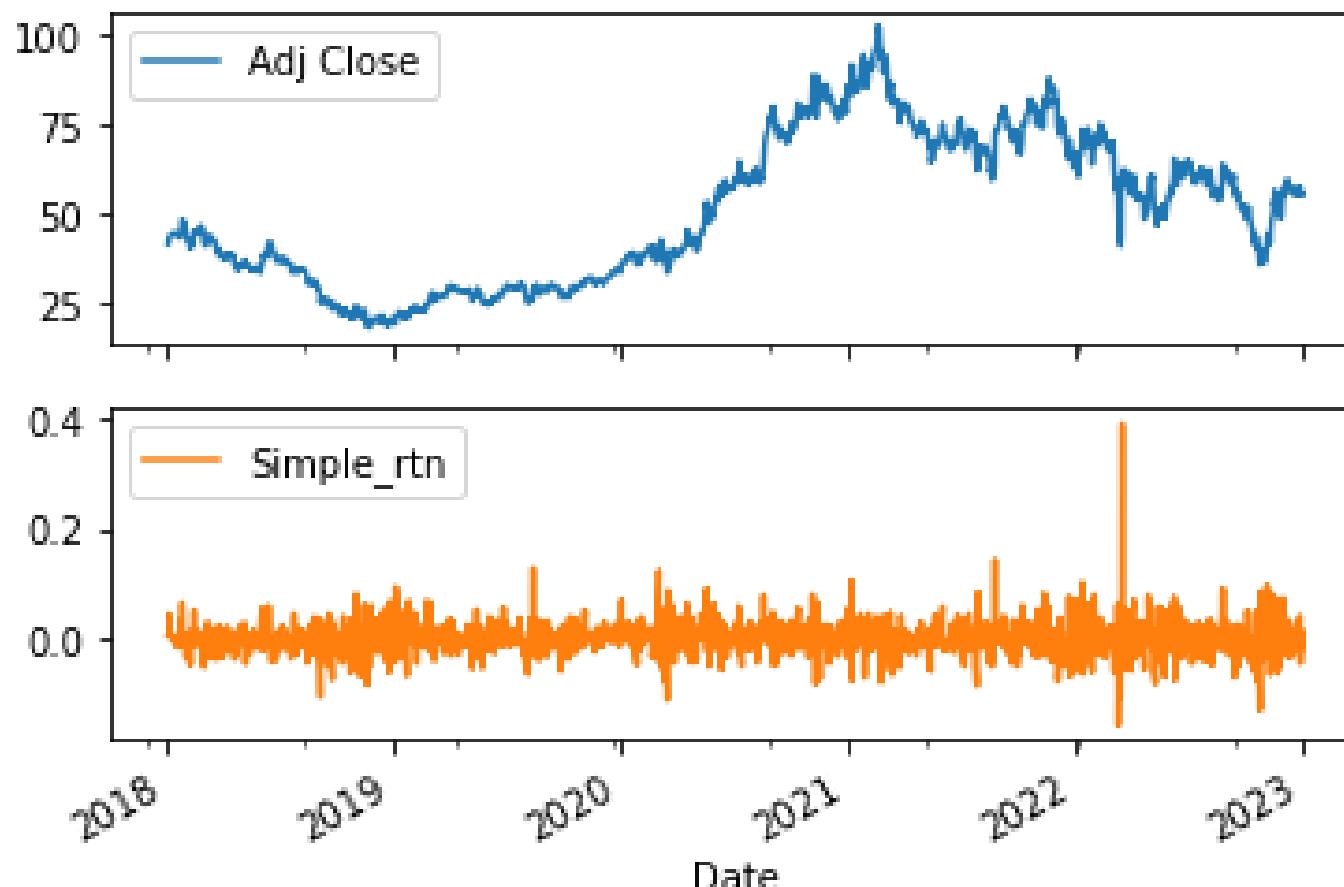
BABA Time Series 2018-2022



AMZN Time Series 2018-2022



JD Time Series 2018-2022



EBAY Time Series 2018-2022

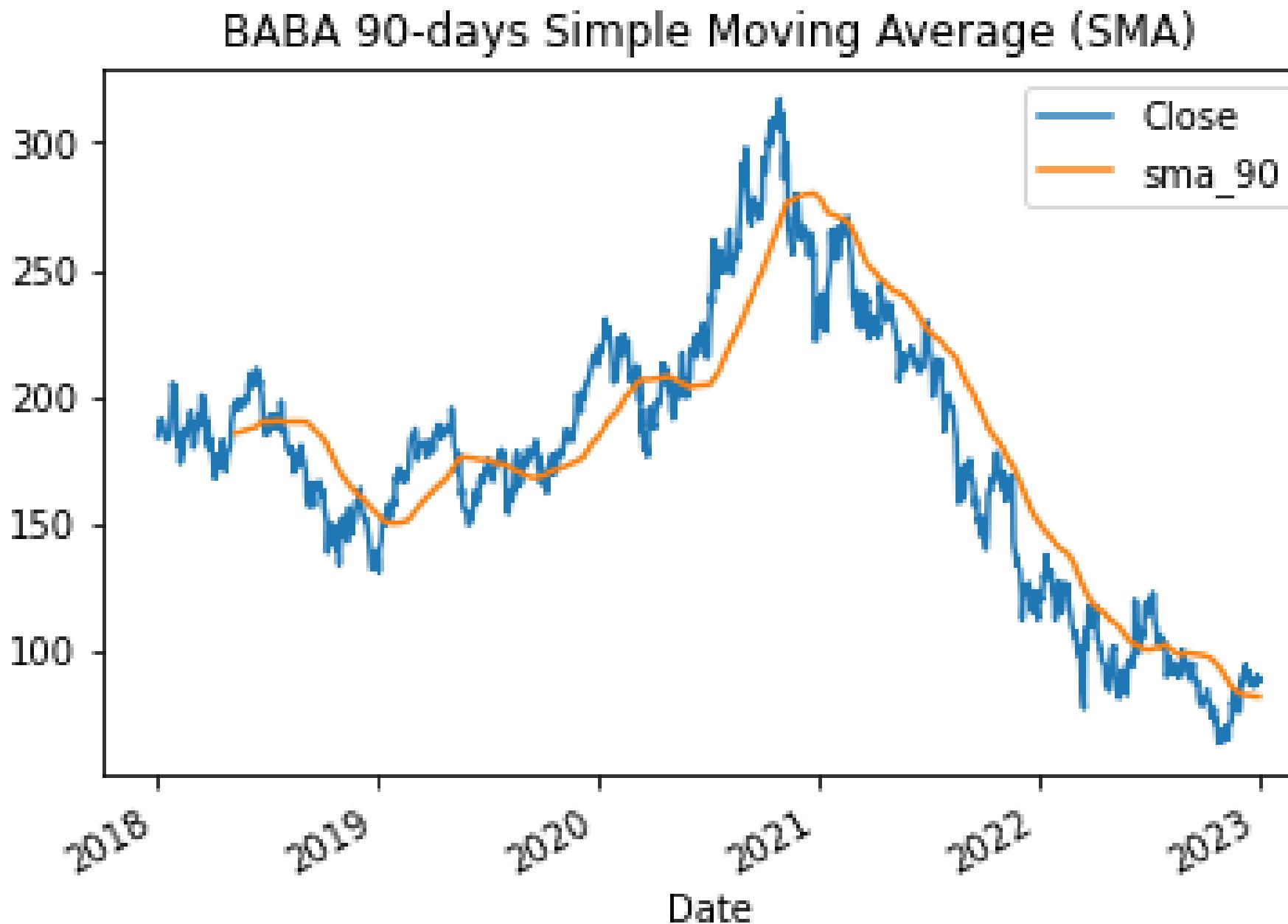




Technical Indicator Analysis

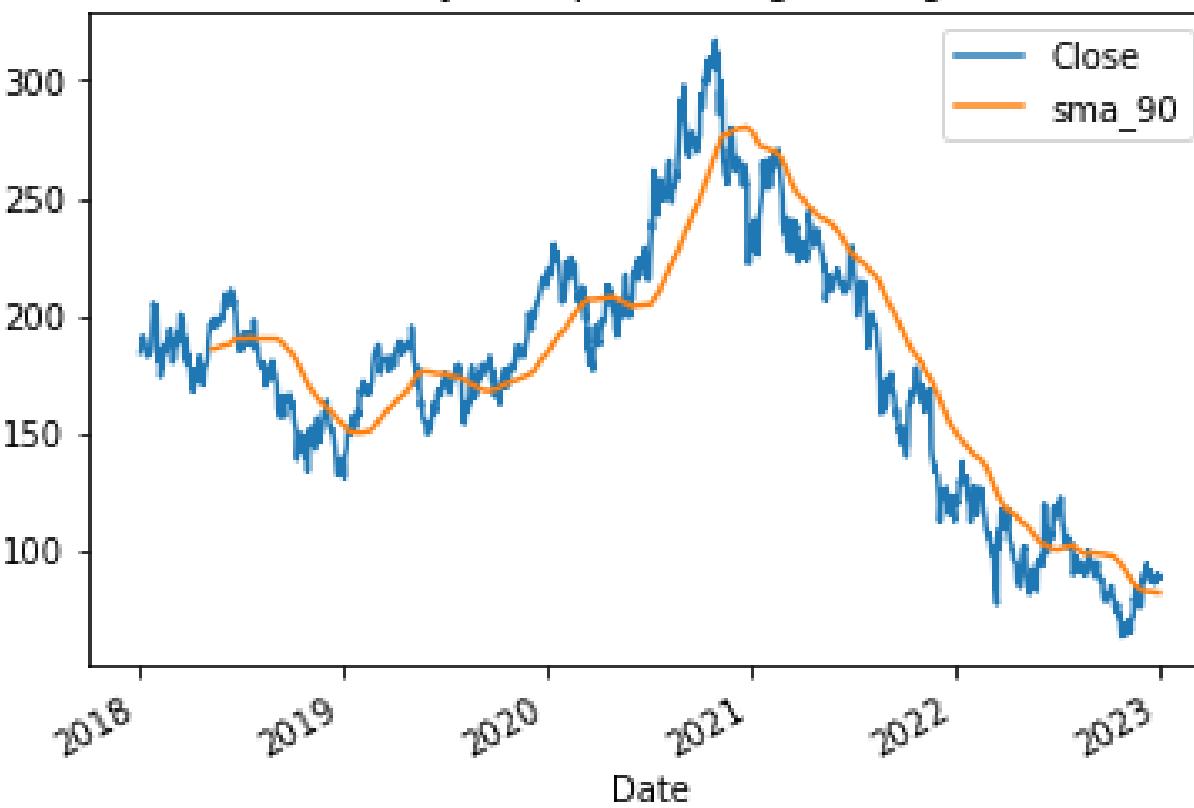
- 90-Days Simple Moving Average (SMA)

90-Days Simple Moving Average

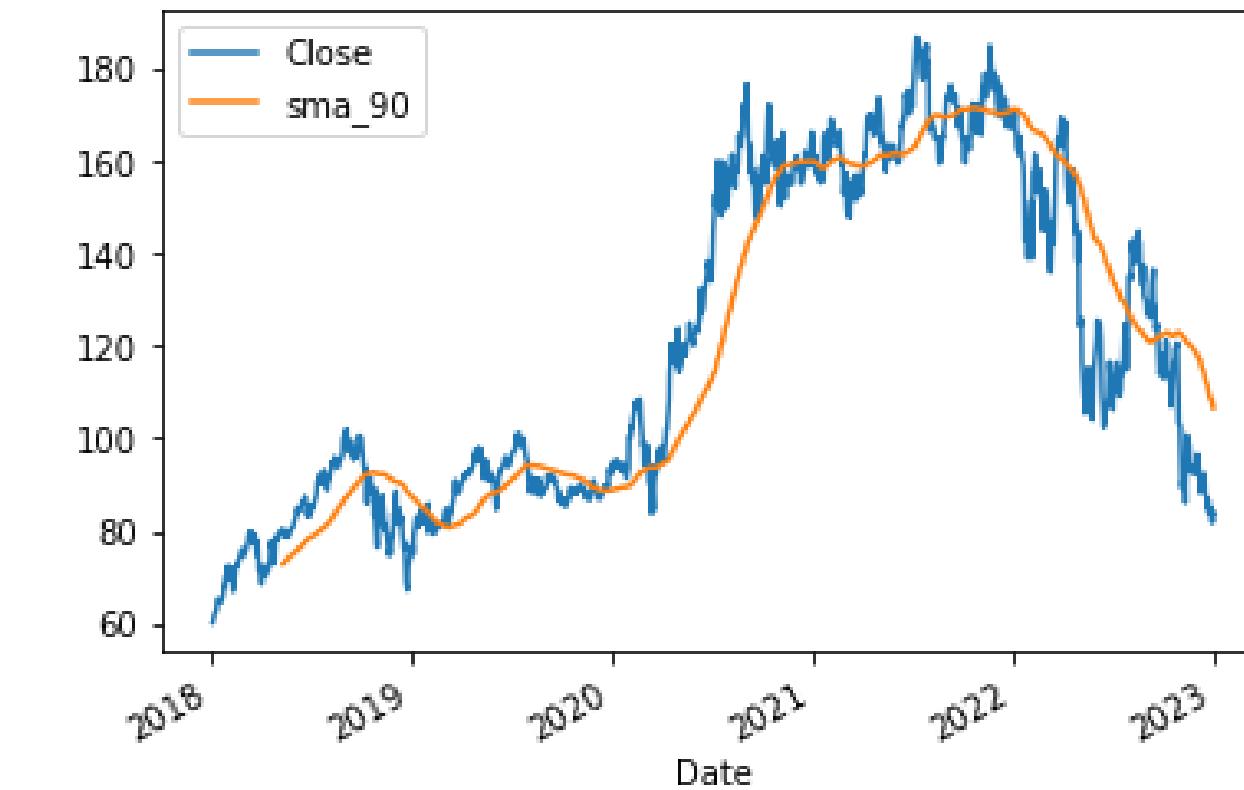


```
import talib  
  
data_1["sma_90"] = talib.SMA(data_1["Close"], timeperiod = 90)  
(  
    data_1[["Close", "sma_90"]]  
    .plot(title = "BABA 90-days Simple Moving Average (SMA)")  
)
```

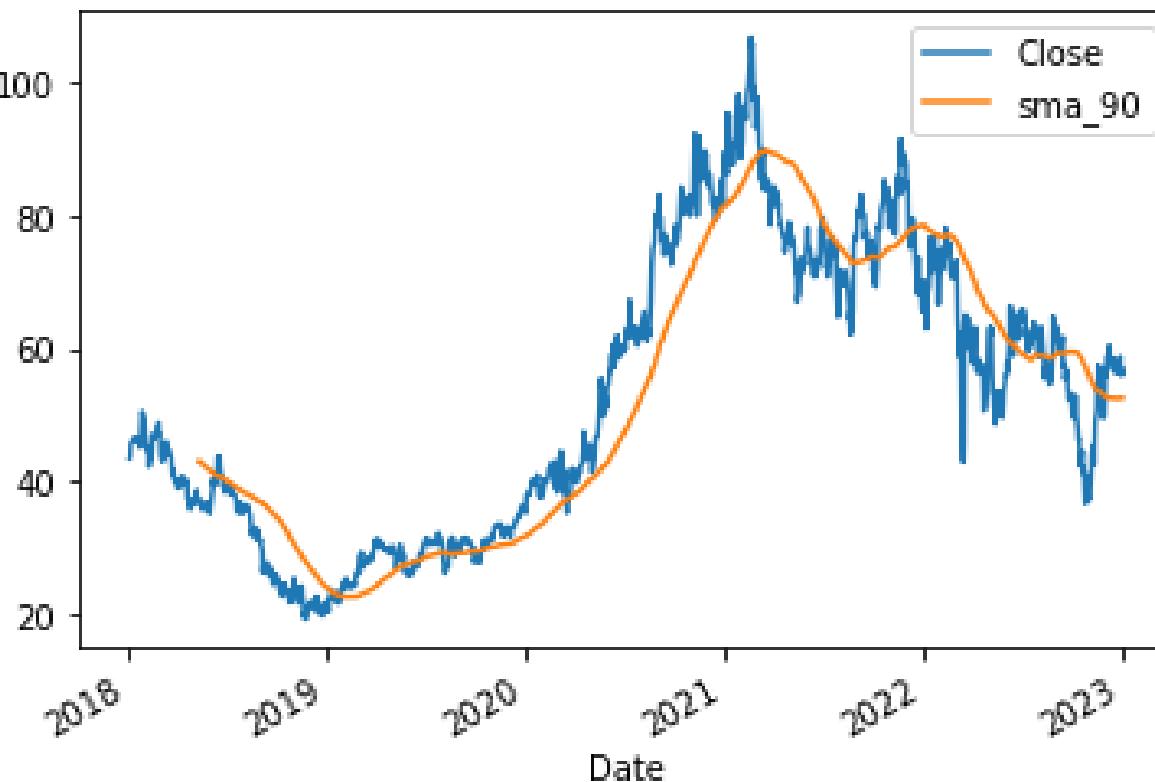
BABA 90-days Simple Moving Average (SMA)



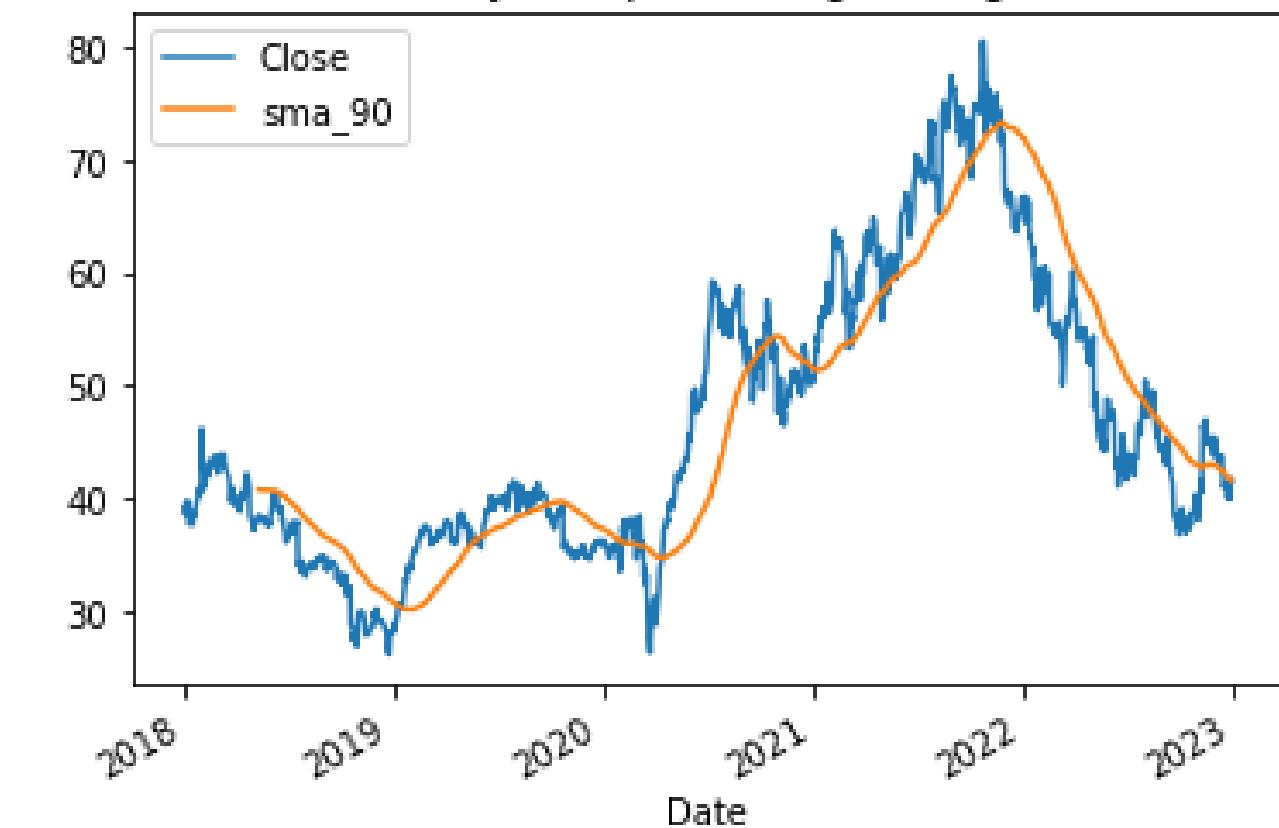
AMZN 90-days Simple Moving Average (SMA)



JD 90-days Simple Moving Average (SMA)



EBAY 90-days Simple Moving Average (SMA)



Conclusion

Data source :
stockanalysis.com / [Yahoofinance.com](https://www.yahoo.com/finance)

Library :
`pandas` / `yfinance` / `numpy` / `matplotlib` / `talib`

THANK YOU