```
import yfinance as yf
import pandas as pd
from sklearn.model_selection import train_test_split
# Fetch Coca-Cola stock data
ticker = "AAPL" # Coca-Cola stock ticker
data = yf.download(ticker, start='2015-01-01',
end='2023-12-31')
data = data.ffill()
# Reset index for easier handling
data.reset_index(inplace=True)
# Display data structure

#print(data.isnull().sum())

# Confirm no missing values remain
#print(data.isnull().sum())
#print(data.isnull().sum())
data['MA_20'] = data['Close'].rolling(window=20).mean()
data['MA_50'] = data['Close'].rolling(window=50).mean()
data['Daily_Return'] = data['Close'].pct_change()
data['Volatility'] = data['Daily_Return'].rolling(window=20).std()

# Drop rows with NA due to rolling calculations
data.dropna(inplace=True)
#print(data.head())
#print(data.describe())

import matplotlib.pyplot as plt
import seaborn as sns
# Line plot for stock prices
#plt.figure(figsize=(12, 6))
```

```
#plt.plot(data['Date'], data['Close'], label='Close Price')
#plt.plot(data['Date'], data['MA_20'], label='MA 20',
#linestyle='--')
#plt.plot(data['Date'], data['MA_50'], label='MA 50',
#linestyle='--')
#plt.title('Coca-Cola Stock Prices with Moving Averages')
#plt.xlabel('Date')
#plt.ylabel('Price')
#plt.legend()
#plt.show()

#plt.figure(figsize=(10, 8))
#sns.heatmap(data.corr(), annot=True, cmap='coolwarm')
#plt.title('Correlation Heatmap')
#plt.show()

features = ["Open", "High", "Low", "Volume", "MA_20", "MA_50", "Daily_Return", "Volatility"]
target = "Close"
X = data[features]
y = data[target]
X_train, X_test, y_train, y_test = train_test_split(
   X, y, test_size=0.2, random_state=42, shuffle=False
)

#print(X_train.head())
#print(y_train.head())
from sklearn.ensemble import RandomForestRegressor
from sklearn.metrics import mean_absolute_error, mean_squared_error

model = RandomForestRegressor(n_estimators=100, random_state=42)
model.fit(X_train, y_train)
```

```
# Predict
y_pred = model.predict(X_test)

# Evaluate
mae = mean_absolute_error(y_test, y_pred)
mse = mean_squared_error(y_test, y_pred)

#print("MAE:", mae)
#print("MSE:", mse)
import streamlit as st
import yfinance as yf
st.title("Coca-Cola Stock Price Prediction")

data = yf.download("KO", start="2015-01-01")
data["MA_20"] = data["Close"].rolling(20).mean()
data["MA_50"] = data["Close"].rolling(50).mean()
data = data.dropna()
st.line_chart(data[["Close", "MA_20", "MA_50"]])
st.write(f"Predicted Closing Price: {live_prediction[0]}")
live_data = yf.download(ticker, period='1d', interval='1m')
# Prepare live data for prediction
live_data['MA_20'] = live_data['Close'].rolling(window=20).mean()
live_data['MA_50'] = live_data['Close'].rolling(window=50).mean()
live_data['Daily_Return'] = live_data['Close'].pct_change()
live_data['Volatility'] = live_data['Daily_Return'].rolling(window=20).std()
# Ensure no missing values
live_data.fillna(0, inplace=True)
# Use the latest data point for prediction
latest_features = live_data[features].iloc[-1:].dropna()
live_prediction = model.predict(latest_features)
```

```
print(f"Predicted Closing Price: {live_prediction[0]}")
```