

```

import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
warnings.filterwarnings('ignore')

df = pd.read_csv("C:\Users\User\OneDrive\Documents\Visual Studio Code\Scientific Computing\AAPL.csv")
df.head()

# cleans and reformats the Date column, filtering the data with dates between 2020-2024
def clean_date_column(df):
    df["Date"] = pd.to_datetime(df["Date"])
    start_date = "2020-01-01"
    end_date = "2024-12-31"
    df = df[(df["Date"] >= start_date) & (df["Date"] <= end_date)]
    df["Date"] = pd.to_datetime(df["Date"]).dt.date.astype(str)
    df = df.reset_index(drop=True)
    return df

# applying the clean_date_column function to the dataframe
df = clean_date_column(df)
df.head()

df["daily_pct_return"] = df["Close"].pct_change() * 100
df.head()

df.info()

def clean_stock_data(df):
    df = df.rename({"Close": "Price"}, axis=1)

    df.columns = df.columns.str.lower()

    for column in df.columns:
        if df[column].dtype == "float64":
            df[column] = df[column].round(2)
    return df

df = clean_stock_data(df)
df.head()

def get_year_month(df):
    df["year"] = pd.to_datetime(df["date"]).dt.year.astype(str)
    df["month"] = pd.to_datetime(df["date"]).dt.month.astype(str)

```

```

df["month_year"] = df["month"] + "-" + df["year"]
return df

df = get_year_month(df)
df.head()

plt.figure(figsize=(15,5),dpi=180)
sns.lineplot(data=df,x="month_year",y="price",ci=None,color='#E50914')
plt.tick_params(axis="x",labelsize="small",labelrotation=-70)
plt.title("Apple Stock Price (2020 - 2024)")
plt.xlabel("Date")
plt.ylabel("Price (USD)")
plt.show()

plt.figure(figsize=(12,5),dpi=150)

axplot = sns.histplot(data=df,x="volume",color='#40B0A6')
axplot.set(title="Apple Trading Volume Distribution (2020 - 2024)",xlabel="Trading Volume",ylabel="Count")

mean_volume = df["volume"].mean()
plt.axvline(mean_volume,color="orange",ls="--",lw=2,dashes=(1,2),label="Mean Trading Volume")

std_volume = df["volume"].std()

# Mean ± 1 STD
plt.axvline(mean_volume - std_volume, color="salmon", ls="--", lw=2,label="Mean +/- 1 Std")
plt.axvline(mean_volume + std_volume, color="salmon", ls="--", lw=2)

# Mean ± 2 STD
plt.axvline(mean_volume - 2*std_volume, color="red", ls="--", lw=2,label="Mean +/- 2 Std")
plt.axvline(mean_volume + 2*std_volume, color="red", ls="--", lw=2)

# Mean ± 3 STD
plt.axvline(mean_volume - 3*std_volume, color="darkred", ls="--", lw=2,label="Mean +/- 3 Std")
plt.axvline(mean_volume + 3*std_volume, color="darkred", ls="--", lw=2)

plt.legend(loc="upper right")
plt.show()

```