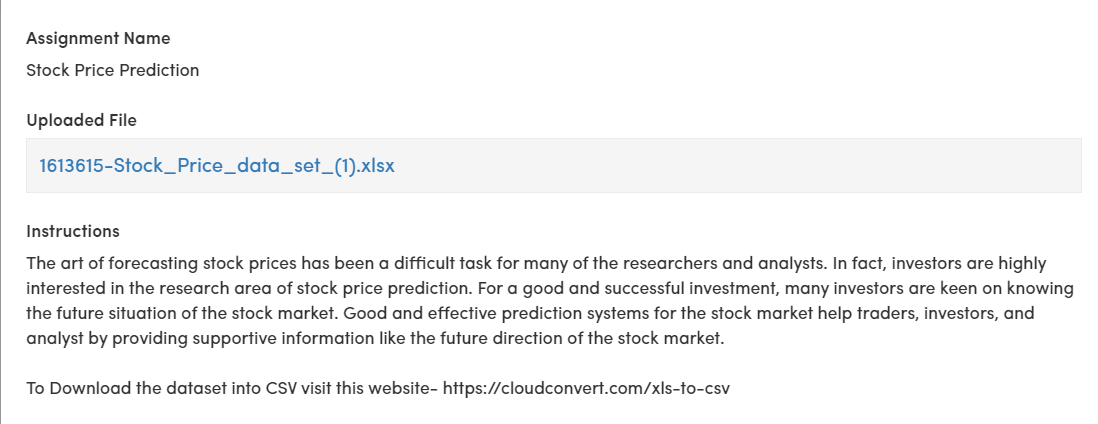
MINOR PROJECT

STOCK PRICE PREDICTION



CODING:

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

from sklearn.model\_selection import train\_test\_split

from sklearn.linear\_model import LinearRegression

from sklearn.metrics import mean\_squared\_error, r2\_score

file\_path = "Stock\_Price.csv"

df = pd.read\_excel(file\_path)

print(df.head())

# Assuming the dataset has columns like 'Date', 'Open', 'High', 'Low', 'Close', 'Volume'

print("Missing values:")

print(df.isnull().sum())

df['Date'] = pd.to\_datetime(df['Date'])

df.sort\_values('Date', inplace=True)

df.set\_index('Date', inplace=True)

df['DayofWeek'] = df.index.dayofweek

features = ['Open', 'High', 'Low', 'Volume', 'DayofWeek']

target = ['Close']

X = df[features]

y = df[target]

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)

model = LinearRegression()

model.fit(X\_train, y\_train)

y\_pred = model.predict(X\_test)

mse = mean\_squared\_error(y\_test, y\_pred)

r2 = r2\_score(y\_test, y\_pred)

print("Mean Squared Error:", mse)

print("R-squared:", r2)

plt.figure(figsize=(12, 6))

plt.plot(df.index[-len(y\_test):], y\_test, label='Actual Prices', color='blue')

plt.plot(df.index[-len(y\_test):], y\_pred, label='Predicted Prices', color='red')

plt.xlabel("Date")

plt.ylabel("Stock Price")

plt.title("Actual vs. Predicted Stock Prices")

plt.legend()

plt.show()