import pandas as pd

import numpy as np

# 1) Load the dataset from your local machine

df = pd.read\_csv(r'C:\Users\abhis\Desktop\dataset for cleaning\data sets\Uncleaned\_DS\_jobs.csv')

# 2) Display the first 5 rows to confirm it's loaded correctly

print("Original Data Head:")

print(df.head())

# 3) Get an initial overview of the data

print("\nOriginal Data Info:")

df.info()

# 4) Remove rows where salary is '-1' (or not specified)

df = df[df['Salary Estimate'] != '-1']

# 5) Clean the 'Salary Estimate' column text

# Remove '(Glassdoor est.)' and 'K' and '$' symbols

salary = df['Salary Estimate'].str.split('(').str[0]

salary = salary.str.replace('K', '').str.replace('$', '')

# 6) Split the range into 'min\_salary' and 'max\_salary'

df['min\_salary'] = salary.apply(lambda x: int(x.split('-')[0]))

df['max\_salary'] = salary.apply(lambda x: int(x.split('-')[1]))

# 7) Create an 'avg\_salary' column (in thousands)

df['avg\_salary'] = (df['min\_salary'] + df['max\_salary']) / 2

print("\nSalary columns cleaned and created.")

# 8) Remove the rating and newline characters from the Company Name

df['company\_cleaned'] = df['Company Name'].apply(lambda x: x.split('\n')[0])

print("Company Name cleaned.")

# 9) Create a 'job\_state' column by splitting the location string

df['job\_state'] = df['Location'].apply(lambda x: x.split(',')[1].strip() if ',' in x else x.strip())

# 10) Check the count of jobs per state

print("\nJobs per state:")

print(df.job\_state.value\_counts().head(10))

# 11) Replace invalid 'Founded' year (-1) with NaN (Not a Number)

df['Founded'] = df['Founded'].replace(-1, np.nan)

# 12) Simplify the 'Size' column by removing " employees"

df['Size'] = df['Size'].str.replace(' employees', '')

print("\nFounded and Size columns cleaned.")

# 13) Replace invalid 'Founded' year (-1) with NaN (Not a Number)

df['Founded'] = df['Founded'].replace(-1, np.nan)

# 14) Simplify the 'Size' column by removing " employees"

df['Size'] = df['Size'].str.replace(' employees', '')

print("\nFounded and Size columns cleaned.")

# 15) Simplify the 'Revenue' column

df['Revenue'] = df['Revenue'].str.replace(' (USD)', '', regex=False)

df['Revenue'] = df['Revenue'].replace('Unknown / Non-Applicable', np.nan)

print("Revenue column cleaned.")

# 16) Select and reorder the most important columns for a cleaner view

df\_cleaned = df[['Job Title', 'company\_cleaned', 'Location', 'job\_state',

'min\_salary', 'max\_salary', 'avg\_salary',

'Size', 'Founded', 'Type of ownership', 'Industry',

'Sector', 'Revenue', 'Job Description']]

print("\nHead of the Final Cleaned DataFrame:")

print(df\_cleaned.head())

print("\nInfo of the Final Cleaned DataFrame:")

df\_cleaned.info()

# 17) Save the cleaned dataframe to a new CSV file

df\_cleaned.to\_csv('Cleaned\_DS\_jobs.csv', index=False)

print("\nSuccessfully cleaned the data and saved it to 'Cleaned\_DS\_jobs.csv'")

# 18) Define the full path for the new, cleaned file

cleaned\_file\_path = r'C:\Users\abhis\Desktop\dataset for cleaning\data sets\Cleaned\_DS\_jobs.csv'

# 19) Save the cleaned DataFrame to this specific location

df\_cleaned.to\_csv(cleaned\_file\_path, index=False)

print(f"Cleaned data successfully saved to: {cleaned\_file\_path}")