[question-6-time-check](https://github.com/Harshalns/MapUp-Data-Assessment-F#question-6-time-check)

import pandas as pd

def verify\_time\_completeness(df):

# Assuming df is a DataFrame with columns id, id\_2, startDay, startTime, endDay, endTime

df['start\_datetime'] = pd.to\_datetime(df['startDay'] + ' ' + df['startTime'])

df['end\_datetime'] = pd.to\_datetime(df['endDay'] + ' ' + df['endTime'])

# Create a full 24-hour period for each day of the week

full\_24\_hours = pd.date\_range(start='00:00:00', end='23:59:59', freq='1s')

# Create a DataFrame with all combinations of id, id\_2, and time

all\_combinations = pd.MultiIndex.from\_product([df['id'].unique(), df['id\_2'].unique(), full\_24\_hours],

names=['id', 'id\_2', 'timestamp'])

# Merge the original DataFrame with all combinations to fill missing values

df\_all\_combinations = pd.DataFrame(index=all\_combinations).reset\_index()

df\_merged = pd.merge(df\_all\_combinations, df, on=['id', 'id\_2', 'timestamp'], how='left')

# Check for missing values, indicating incorrect timestamps

incomplete\_time\_data = df\_merged['start\_datetime'].isnull() | df\_merged['end\_datetime'].isnull()

# Create a boolean series with multi-index (id, id\_2)

result\_series = incomplete\_time\_data.groupby(['id', 'id\_2']).any()

return result\_series

# Example usage:

# Assuming dataset-2.csv is your CSV file

file\_path = 'path\_to\_your\_file/dataset-2.csv'

data = pd.read\_csv(file\_path)

result = verify\_time\_completeness(data)

print(result)