import pandas as pd

# Sample data

data = {

'account\_id': ['A1', 'A1', 'A1', 'A2', 'A2', 'A3'],

'channel\_visit\_id': [1, 1, 1, 2, 2, 3],

'page\_number': [1, 2, 3, 1, 2, 1],

'page': ['Springboard', 'Manage', 'Check Allowances', 'Springboard', 'Manage', 'Check Allowances']

}

df = pd.DataFrame(data)

print("Original DataFrame:")

print(df)

# Step 1: Sort the DataFrame by 'account\_id', 'channel\_visit\_id', and 'page\_number'

sorted\_df = df.sort\_values(by=['account\_id', 'channel\_visit\_id', 'page\_number'])

print("\nStep 1: Sorted DataFrame:")

print(sorted\_df)

# Step 2: Group by 'account\_id' and 'channel\_visit\_id' and aggregate 'page' into a list

grouped\_df = (

sorted\_df.groupby(['account\_id', 'channel\_visit\_id'])['page']

.apply(list)

.reset\_index()

)

print("\nStep 2: Grouped DataFrame with paths for each account:")

print(grouped\_df)

# Step 3: Group paths by 'account\_id' to get all paths under each account

account\_paths\_df = (

grouped\_df.groupby('account\_id')['page']

.apply(list)

.reset\_index()

)

account\_paths\_df.rename(columns={'page': 'paths'}, inplace=True)

print("\nStep 3: Account-wise grouped paths:")

print(account\_paths\_df)

# Optional: Save the result to a CSV file

account\_paths\_df.to\_csv("account\_wise\_paths.csv", index=False)

print("\nResults saved to 'account\_wise\_paths.csv'.")

import networkx as nx

import matplotlib.pyplot as plt

# Create a directed graph for visualization

G = nx.DiGraph()

# Iterate through account\_paths\_df to add edges for each account

for \_, row in account\_paths\_df.iterrows():

account\_id = row['account\_id']

paths = row['paths']

for path in paths:

for i in range(len(path) - 1):

G.add\_edge(path[i], path[i+1], account=account\_id)

# Plot the graph

plt.figure(figsize=(12, 8))

pos = nx.spring\_layout(G, seed=42) # Set layout for the graph

nx.draw\_networkx\_nodes(G, pos, node\_size=700, node\_color="lightblue")

nx.draw\_networkx\_edges(G, pos, edgelist=G.edges(), edge\_color="gray", arrows=True)

nx.draw\_networkx\_labels(G, pos, font\_size=10, font\_color="black")

# Add a title

plt.title("Account-Wise Path Network")

plt.show()

pip install plotly

import plotly.graph\_objects as go

# Prepare data for Sankey Diagram

source = [] # Source nodes

target = [] # Target nodes

value = [] # Weights (counts of transitions)

# Journey-wise transitions (from grouped\_df)

for \_, row in grouped\_df.iterrows():

path = row['page']

for i in range(len(path) - 1):

source.append(path[i])

target.append(path[i + 1])

value.append(1) # Each transition counts as 1

# Convert source/target nodes into numeric indices for Sankey

unique\_nodes = list(set(source + target))

node\_indices = {node: i for i, node in enumerate(unique\_nodes)}

source\_indices = [node\_indices[s] for s in source]

target\_indices = [node\_indices[t] for t in target]

# Create Sankey Diagram

fig = go.Figure(data=[go.Sankey(

node=dict(

pad=15,

thickness=20,

line=dict(color="black", width=0.5),

label=unique\_nodes

),

link=dict(

source=source\_indices,

target=target\_indices,

value=value

)

)])

# Add title and show plot

fig.update\_layout(title\_text="Journey-Wise Paths (Sankey Diagram)", font\_size=10)

fig.show()