**EJERCICIO EN CLASE**

1. Análisis de Transacciones:

import pandas as pd

import matplotlib.pyplot as plt

# Ejemplo de datos de transacciones

transacciones = {

 "Remitente": ["A", "B", "C", "D", "E"],

 "Destinatario": ["X", "Y", "Z", "X", "Y"],

 "Cantidad": [100, 200, 150, 300, 250],

 "Marca de Tiempo": ["2022-01-01", "2022-01-02", "2022-01-03", "2022-01-04", "2022-01-05"]

}

# Crear un DataFrame de Pandas

df\_transacciones = pd.DataFrame(transacciones)

# Graficar la cantidad de transacciones a lo largo del tiempo

df\_transacciones["Marca de Tiempo"] = pd.to\_datetime(df\_transacciones["Marca de Tiempo"])

df\_transacciones.set\_index("Marca de Tiempo", inplace=True)

df\_transacciones.resample("D").size().plot(kind="bar")

plt.title("Cantidad de Transacciones por Día")

plt.xlabel("Fecha")

plt.ylabel("Cantidad de Transacciones")

plt.show()

2. Verificación de la Coherencia de la Cadena:

import networkx as nx

import matplotlib.pyplot as plt

# Ejemplo de datos de una cadena de bloques

cadena\_de\_bloques = {

 "Número de Bloque": [1, 2, 3],

 "Bloque Anterior": [None, 1, 2],

 "Transacciones": [[1, 2], [3, 4], [5, 6]],

 "Sello de Tiempo": ["2022-01-01", "2022-01-02", "2022-01-03"]

}

# Crear un grafo dirigido para representar la cadena de bloques

G = nx.DiGraph()

# Agregar nodos y aristas al grafo

for i, bloque in enumerate(cadena\_de\_bloques["Número de Bloque"]):

 G.add\_node(bloque)

 if cadena\_de\_bloques["Bloque Anterior"][i] is not None:

 G.add\_edge(cadena\_de\_bloques["Bloque Anterior"][i], bloque)

# Dibujar el grafo

nx.draw(G, with\_labels=True, node\_size=2000, node\_color="lightblue", font\_size=10, font\_weight="bold")

plt.title("Estructura de la Cadena de Bloques")

plt.show()