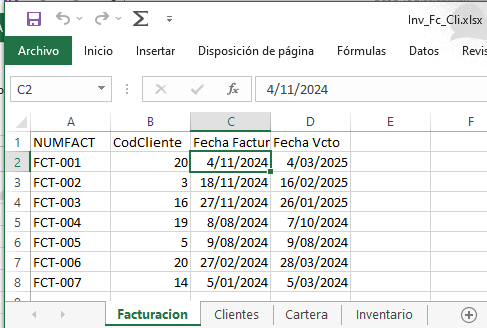
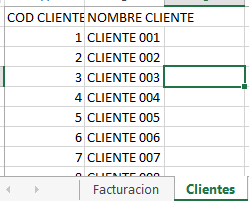
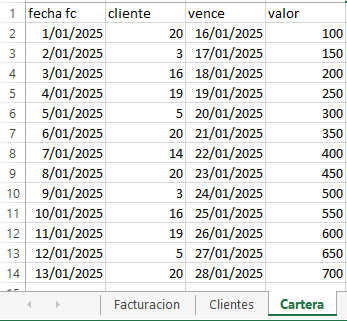
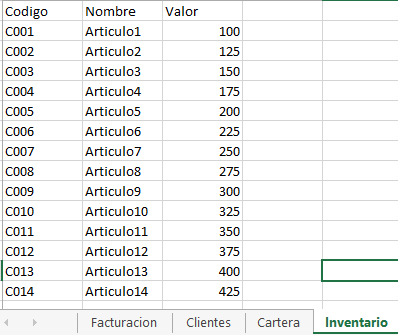
Inventario-Facturacion-Cartera\_Ventas con chatGPT

**mi\_proyecto\_python/ Fase 1:**

**el archivo de excel :**

**Lectura de excel con hojas** ["Facturacion", "Clientes", "Cartera", "Inventario"]

Estructura de datos:

│└── data/

│ └── Inv\_Fc\_Cli.xlsx

├── src/

│ ├── main.py

│ ├── archivos/

│ │ └── procesador\_excel.py

**main.py**

from archivos.procesador\_excel import leer\_archivo\_excel

ruta = "data/Inv\_Fc\_Cli.xlsx"

datos = leer\_archivo\_excel(ruta)

for hoja, df in datos.items():

print(f"📊 {hoja}: {df.shape[0]} filas y {df.shape[1]} columnas")

**procesador\_excel.py**

import pandas as pd

import os

HOJAS\_REQUERIDAS = ["Facturacion", "Clientes", "Cartera", "Inventario"]

def leer\_archivo\_excel(ruta\_archivo):

if not os.path.exists(ruta\_archivo):

raise FileNotFoundError(f"No se encontró el archivo: {ruta\_archivo}")

print(f"📂 Leyendo archivo: {ruta\_archivo}")

try:

hojas = pd.read\_excel(ruta\_archivo, sheet\_name=None)

except Exception as e:

raise ValueError(f"No se pudo leer el archivo Excel: {e}")

datos = {}

for hoja in HOJAS\_REQUERIDAS:

if hoja not in hojas:

raise ValueError(f"Falta la hoja requerida: {hoja}")

df = hojas[hoja].copy()

df.dropna(how='all', inplace=True)

df.columns = [str(col).strip() for col in df.columns]

datos[hoja] = df

print(f"✅ Hoja '{hoja}' cargada con {len(df)} registros.")

return datos

**Se ejecuta en la consola <Ctrl> +>ñ> con el comando :**

python src/main.py

**Salida del modulo 1:**

📂 Leyendo archivo: data/Inv\_Fc\_Cli.xlsx

✅ Hoja 'Facturacion' cargada con 7 registros.

✅ Hoja 'Clientes' cargada con 20 registros.

✅ Hoja 'Cartera' cargada con 14 registros.

✅ Hoja 'Inventario' cargada con 14 registros.

📊 Facturacion: 7 filas, 4 columnas

📊 Clientes: 20 filas, 2 columnas

📊 Cartera: 14 filas, 4 columnas

📊 Inventario: 14 filas, 3 columnas