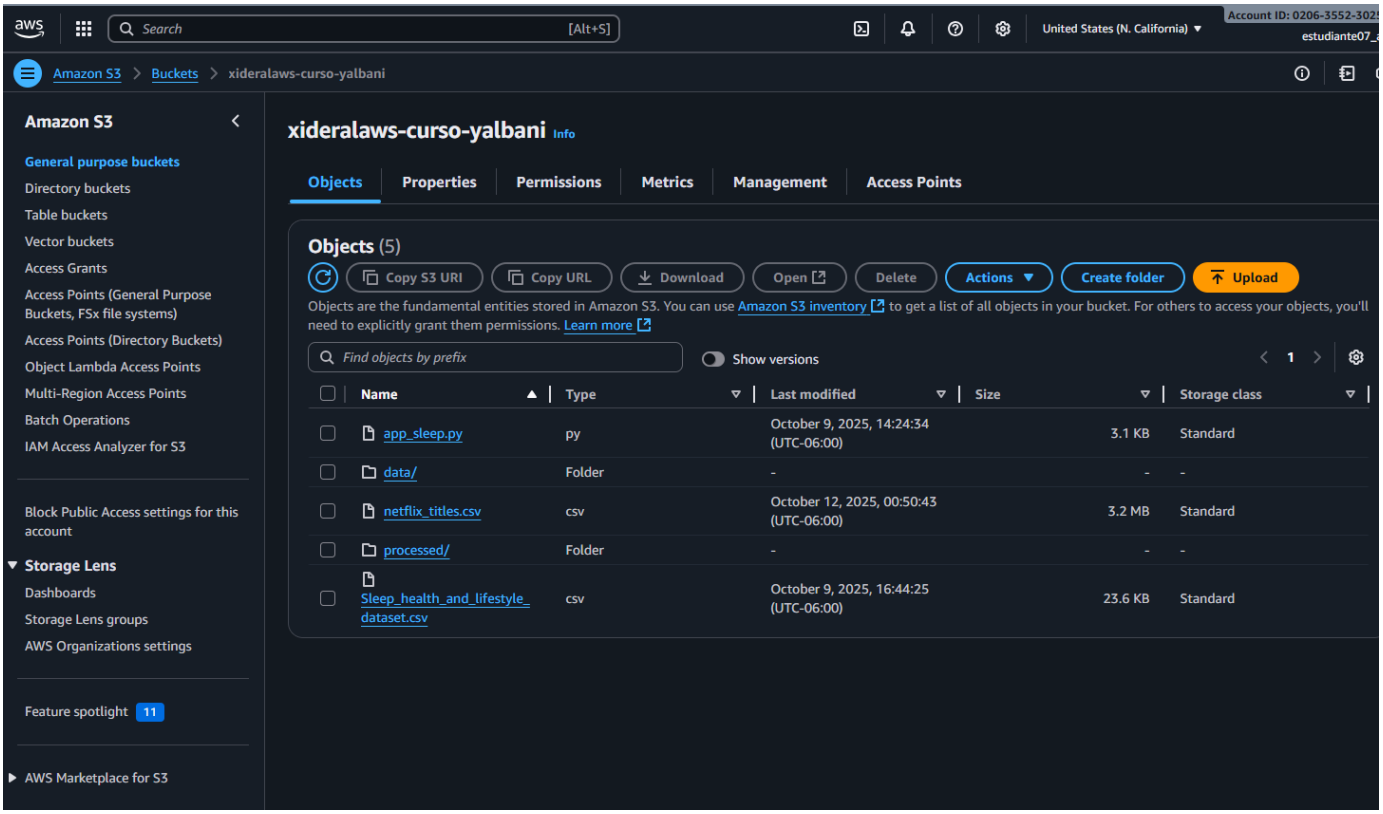
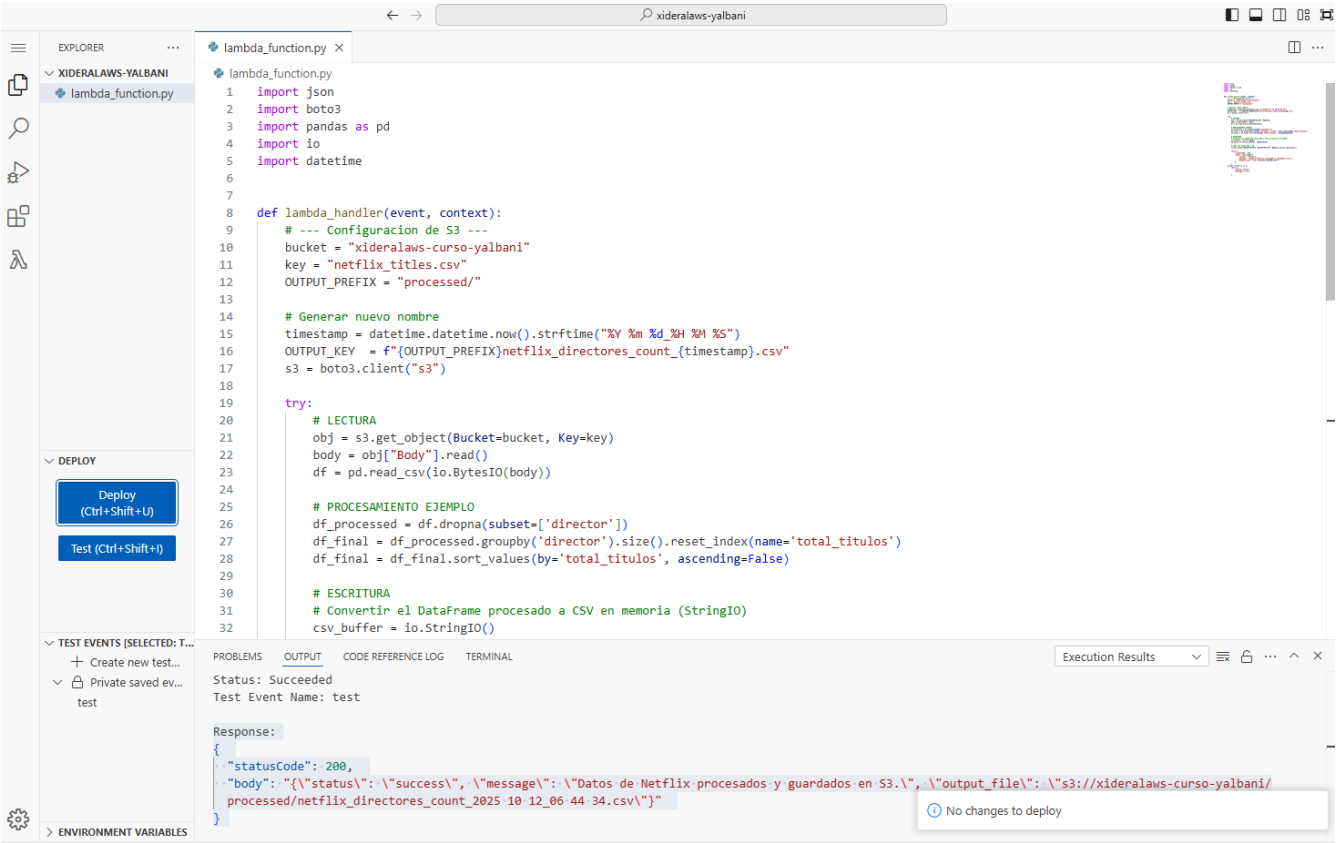
LAMBDA A STRAMLIT

- Contenido S3



- Funcion LAMBDA. Guarda procesamiento en carpeta processed



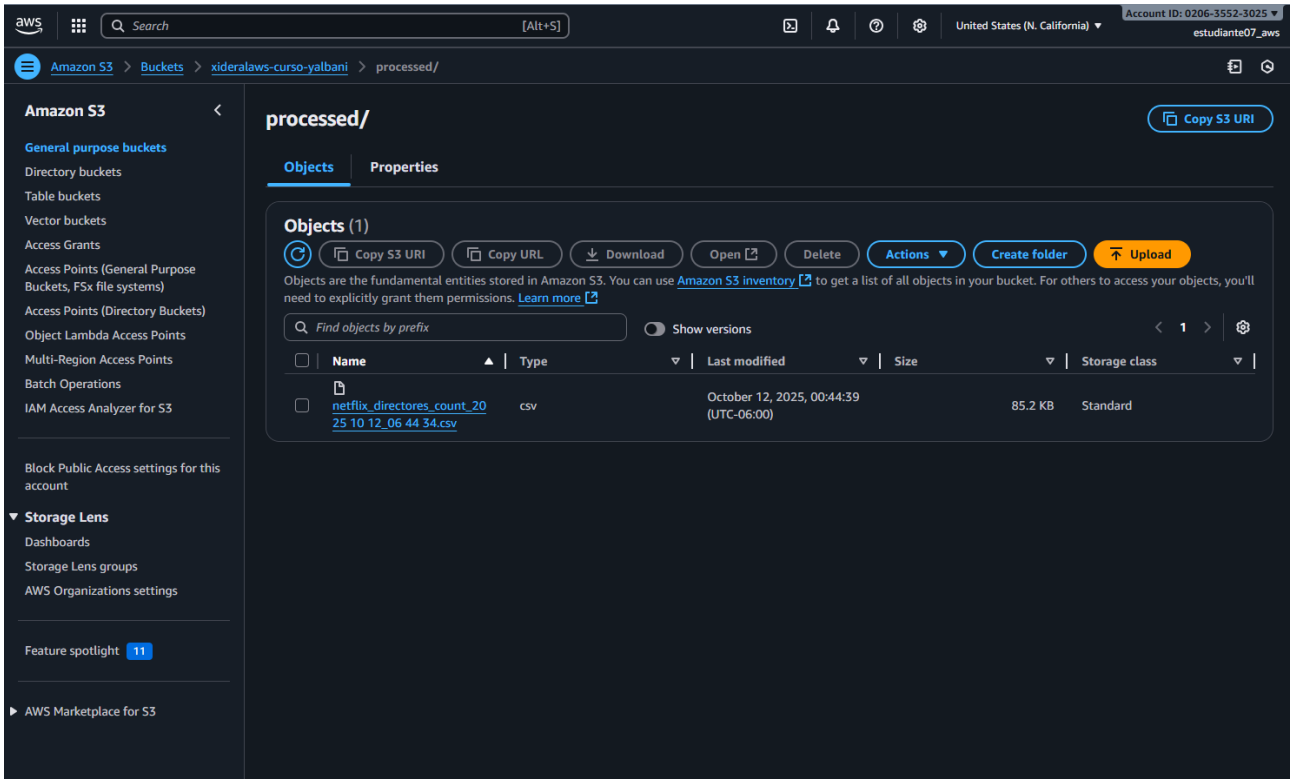
- Con el csv guardado, se activa el streamlit

jupyter app_lambda.py Last Checkpoint: 6 minutes ago

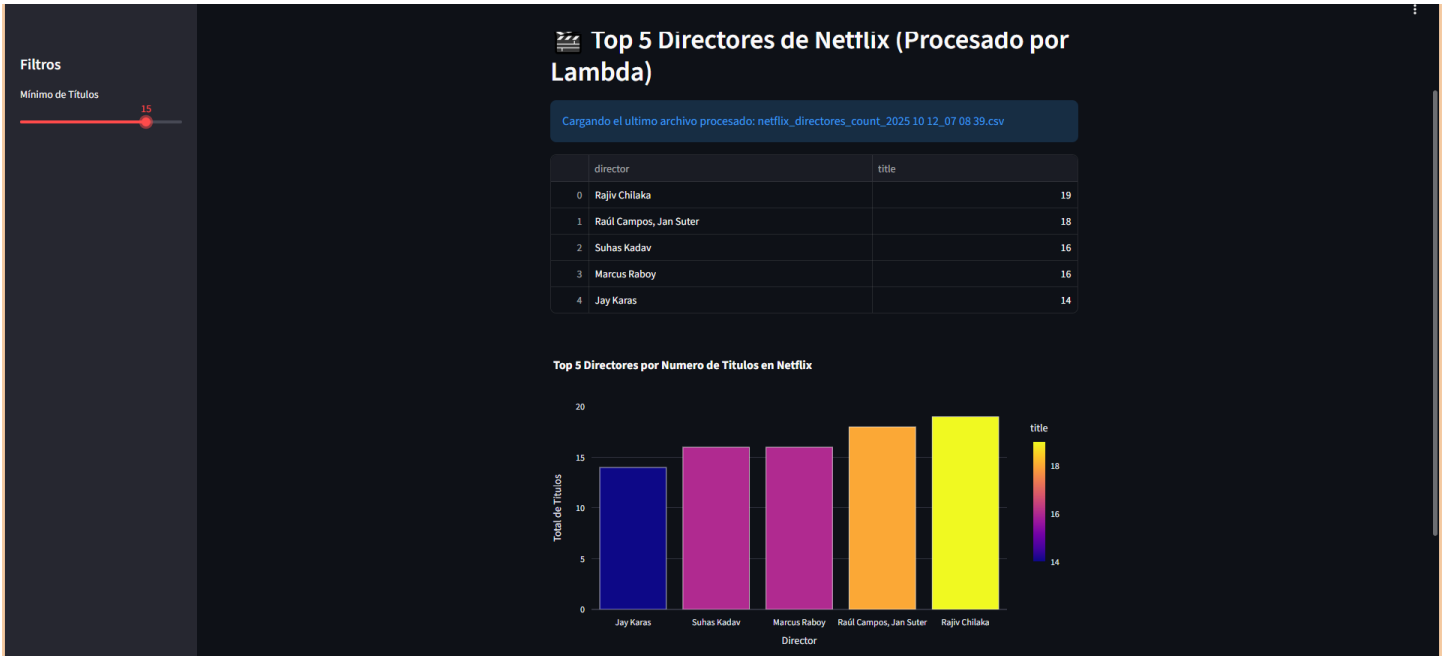
File Edit View Settings Help

```python
import pandas as pd
import streamlit as st
import boto3
import io
import requests
import json
import plotly.express as px

# --- Carga de Datos Procesados desde S3 ---
@st.cache_data
def cargar_datos_procesados():
    s3 = boto3.client("s3")
    bucket = "xideralaws-curso-yalbani"
    OUTPUT_PREFIX = "processed/"

    try:
        # Listar todos los archivos en la carpeta procesada
        response = s3.list_objects_v2(Bucket=bucket, Prefix=OUTPUT_PREFIX)

        if "Contents" not in response:
            st.warning("Lambda no ha ejecutado el procesamiento. No se encontraron archivos procesados.")
            return pd.DataFrame()

        # Encontrar el archivo mas reciente (timestamp mas alto)
        all_files = response["Contents"]
        # Filtrar solo archivos CSV si fuera necesario, y ordenar por la ultima modificacion
        all_files = sorted(all_files, key=lambda x: x['LastModified'], reverse=True)

        # El archivo mas reciente es el primero
        latest_key = all_files[0]['Key']

        # Leer el archivo mas reciente
        st.info(f"Cargando el ultimo archivo procesado: {latest_key.split('/')[-1]}")
        obj = s3.get_object(Bucket=bucket, Key=latest_key)
        body = obj["Body"].read()

        df_final = pd.read_csv(io.BytesIO(body))
        return df_final

    except Exception as e:
        st.error(f"Error al cargar el CSV procesado desde S3: {e}")
        return pd.DataFrame()

# --- Streamlit ---
```

- Aquí se guarda el csv procesado



- Resultado Streamlit



- Crear un CRON que ejecute diario de Lunes a Viernes a las 13 horas y agregarlo a la lambda

cron(0 13 ? * MON-FRI *)