3-8-2025

Brandon MejiaJARAMILLO MARTINEZ DONOVAN ALEXIS

MEJIA VILLAFRANCA BRANDON URIEL

SANCHEZ GOMEZ JOSE DANIEL

SERRANO VAZQUEZ JONATAN MANUEL

VALERIO SANTOS ANGEL GAE

Proyecto Final. 3er. Parcial.

Base de Datos para Computo en la Nube

**📄 2. Descripción del Proyecto**

Este proyecto tiene como objetivo analizar datos sobre la calidad del aire y emisiones contaminantes en México. Utilizamos una base de datos abierta proveniente de datos.gob.mx, y trabajamos con herramientas como **Pandas**, **Seaborn**, **Plotly**, **Google Colab** y **Firebase Firestore** para limpiar, analizar y visualizar los datos, así como para almacenarlos en la nube.

**🎯 3. Objetivos**

**3.1 Objetivo General**

Analizar y visualizar los datos de emisiones contaminantes en municipios mexicanos utilizando herramientas de ciencia de datos y bases de datos NoSQL en la nube.

**3.2 Objetivos Específicos**

* Extraer datos abiertos del portal de datos del gobierno.
* Limpiar y transformar datos con Pandas.
* Subir datos relevantes a Firebase Firestore.
* Crear visualizaciones claras con Seaborn y Plotly.
* Analizar resultados y tendencias significativas.

**❗ 4. Justificación del Problema**

La contaminación del aire representa un riesgo importante para la salud y el medio ambiente en México. Este proyecto permite identificar las principales fuentes de contaminación y los municipios más afectados, con el fin de generar conciencia y facilitar decisiones basadas en datos.

**🧩 5. Marco Técnico**

**5.1 Bases de Datos NoSQL**

Firebase Firestore es una base de datos en tiempo real y basada en documentos. Es ideal para almacenar datos estructurados sin necesidad de esquemas estrictos, y permite consultar y visualizar datos de forma flexible.

**5.2 ETL (Extracción, Transformación, Carga)**

* **Extracción:** Se usó un archivo .csv del sitio datos.gob.mx.
* **Transformación:** Se limpiaron valores nulos y se renombraron columnas.
* **Carga:** Se subieron los primeros 10 registros al Firestore.

**5.3 Librerías de Visualización**

Se usaron **Seaborn** para gráficas estáticas y **Plotly Express** para gráficas interactivas.

**🛠️ 6. Desarrollo Técnico**

**6.1 Lectura del CSV en Colab**

python

CopiarEditar

import pandas as pd

df = pd.read\_csv('/d3\_aire01\_49\_1.csv', encoding='latin-1')

**6.2 Limpieza de datos**

python

CopiarEditar

df\_clean = df.dropna(subset=['SO\_2', 'CO', 'NOx'])

df\_clean.columns = df\_clean.columns.str.replace(' ', '\_')

**6.3 Instalación y conexión con Firebase**

python

CopiarEditar

!pip install firebase-admin

import firebase\_admin

from firebase\_admin import credentials, firestore

cred = credentials.Certificate("nombre\_de\_tu\_archivo.json")

firebase\_admin.initialize\_app(cred)

db = firestore.client()

**6.4 Subida de datos a Firestore**

python

CopiarEditar

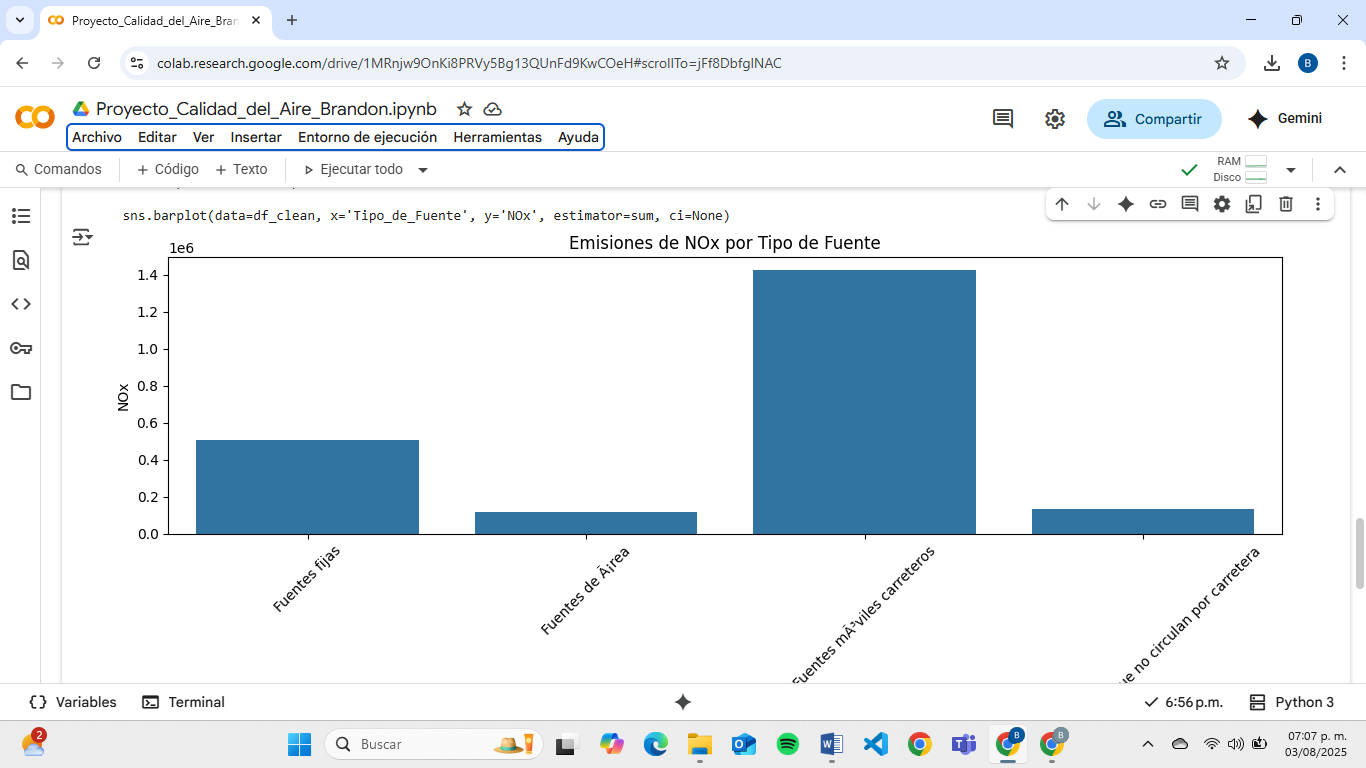
data\_list = df\_clean.head(10).to\_dict(orient='records')

for i, doc in enumerate(data\_list):

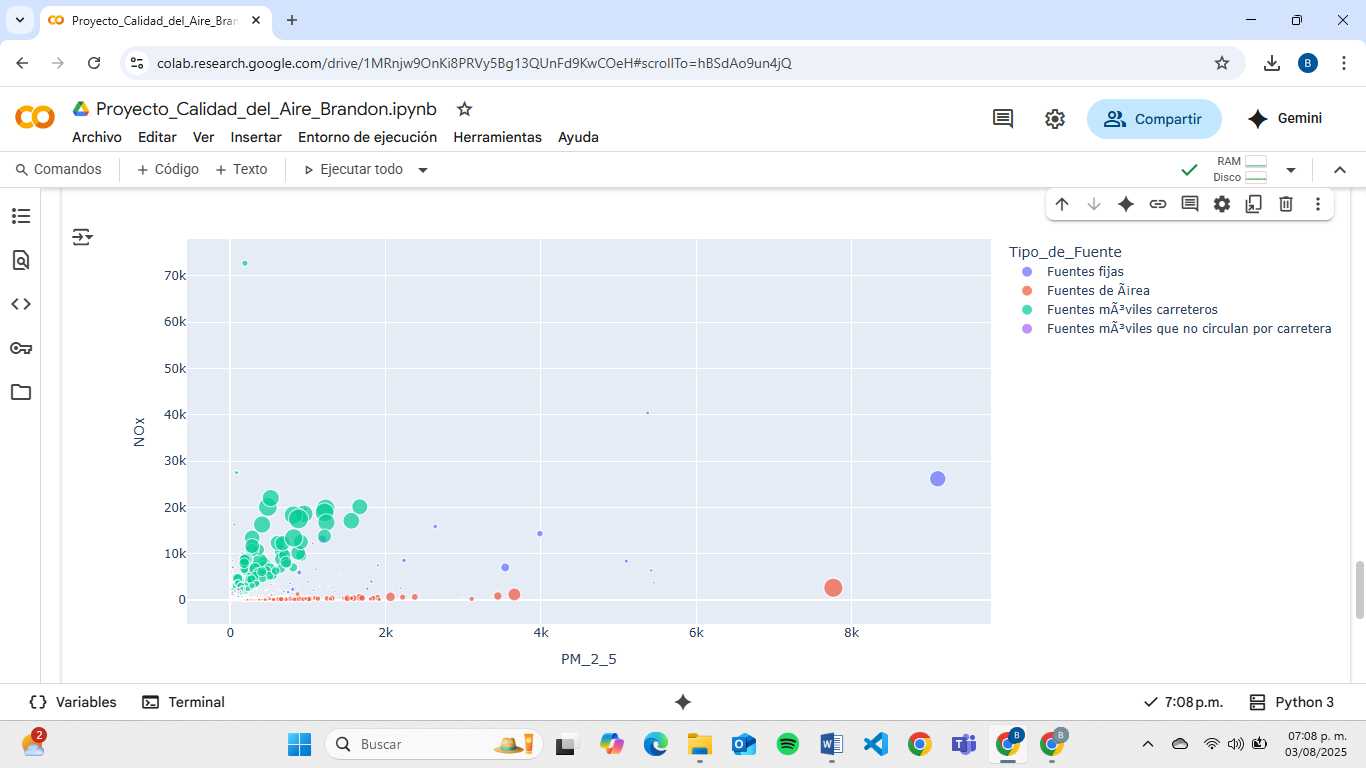
db.collection("emisiones").document(f"doc\_{i}").set(doc)

**📊 7. Visualizaciones**

**7.1 Emisiones de NOx por tipo de fuente**

****

**7.2 Comparación de contaminantes PM\_2\_5 vs NOx**



**📍 8. Resultados y Hallazgos**

* Las **fuentes móviles** (vehículos) son responsables de la mayor cantidad de emisiones de NOx y CO.
* Estados urbanos presentan los niveles más altos de contaminantes.
* Se observó una clara diferencia entre tipos de fuentes fijas y móviles.

**🧠 9. Conclusiones**

* Se logró integrar una base de datos abierta en una plataforma cloud (Firebase) y visualizarla.
* Aprendimos a usar herramientas modernas para análisis de datos como Seaborn, Plotly y Colab.
* El uso de Firestore fue útil para almacenar y consultar datos estructurados fácilmente.

**📦 10. Repositorio**

* [Enlace a GitHub](https://github.com/usuario/proyecto-calidad-aire) https://github.com/Brandon040504-art/proyecto-calidad-aire

**📚 11. Referencias**

* Gobierno de México, Datos Abiertos: https://datos.gob.mx
* Documentación oficial de Firebase: https://firebase.google.com/docs
* Pandas Documentation: https://pandas.pydata.org/
* Seaborn Documentation: https://seaborn.pydata.org/
* Plotly Express: https://plotly.com/python/plotly-express/