Taller: Flujo de datos de datos SQL (Oracle) a NoSQL (MongoDB)

Objetivo:

Al finalizar este taller, los participantes serán capaces de realizar un flujo de datos de Oracle a MongoDB utilizando un programa de Google Colab.

Requisitos:

- Cuenta de Google
- Conocimientos básicos de Python
- Una base de datos de Oracle
- Una base de datos de MongoDB Atlas

Metodología:

- En grupos de trabajo, realicen uno de los cuatro ejercicios asignados.
- Construya la solución basado en el código entregado.

Ejercicios:

Ejercicio 1: CSV a MongoDB

Desde Python con el siguiente código se sube la base seleccionada Noticias:

Se genera la conexión:

```
!pip install pymongo
!pip install --upgrade pymongo

Requirement already satisfied: pymongo in /usr/local/lib/python3.10/dist-packages (4.6.0)
Requirement already satisfied: dnspython<3.0.0,>=1.16.0 in /usr/local/lib/python3.10/dist-packages (from pymongo) (2.4.2)
Requirement already satisfied: pymongo in /usr/local/lib/python3.10/dist-packages (4.6.0)
Requirement already satisfied: dnspython<3.0.0,>=1.16.0 in /usr/local/lib/python3.10/dist-packages (from pymongo) (2.4.2)
```

```
from pymongo.mongo_client import MongoClient
from pymongo.server_api import ServerApi

uri = "mongodb+srv://Adri_30:Melek@cluster0.etcfqhm.mongodb.net/?retryWrites=true&w=majority"

# Create a new client and connect to the server
client = MongoClient(uri, server_api=ServerApi('1'))

# Send a ping to confirm a successful connection
try:
    client.admin.command('ping')
    print("Pinged your deployment. You successfully connected to MongoDB!")
except Exception as e:
    print(e)

Pinged your deployment. You successfully connected to MongoDB!
```

Se genera la base de datos.csv y se sube a python:

```
[47] # Seleccionar una base de datos específica
     db = client["Noticias_"]

     # Ahora puedes usar 'db' para referirte a tu base de datos
     coleccion = db["Noticias_work"]
```

```
# Intentando determinar el delimitador correcto del archivo CSV
import csv
import pandas as pd

# Ruta al archivo CSV
ruta_archivo_csv = '/content/Noticias_bases.csv'

# Cargar el archivo CSV con el delimitador detectado
df = pd.read_csv(ruta_archivo_csv, delimiter=';')

# Mostrar las primeras filas del DataFrame
df.head()
```

	Unnamed: 0	Unnamed: 1	Unnamed: 2	Unnamed: 3	Unnamed: 4	Unnamed: 5	Unnamed: 6	Unnamed: 7	Unnamed: 8	Unnamed: 9	...	Unnamed: 16	Unnamed: 17	Unnamed: 18	Unnamed: 19	Unn
0	NaN	Observatorio de Noticias en Tecnología	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	
1	NaN	#	Fecha Actualización	Fecha de Noticia	Nombre Estudiante	Link Noticia	Titulo	Resumen (No mayor a 50 palabras)	Categorias (Maximo 4 palabras claves que descr...	NaN	...	NaN	NaN	NaN	NaN	
2	NaN	1	17/11/2023	10/5/2023	Maria Camila Prada	https://news.un.org/es/story/2023/05/1520892	Las tecnologías digitales son una herramienta ...	El artículo de la ONU informa sobre la crisis ...	Crisis humanitaria en Yemen\n Falta de fondos\...	NaN	...	NaN	NaN	NaN	NaN	

Este código delimita el archivo CSV, se genera automáticamente antes de cargar los datos en un DataFrame.

```
# Leer las primeras líneas del archivo para determinar el delimitador
with open(ruta_archivo_csv, 'r', encoding='utf-8') as archivo:
    muestra = archivo.read(1024)

delimitador = csv.Sniffer().sniff(muestra).delimiter

# Cargar el archivo CSV con el delimitador detectado
df = pd.read_csv(ruta_archivo_csv, delimiter=delimitador)

# Mostrar las primeras filas del DataFrame
df.head()
```

	Unnamed: 0	Unnamed: 1	Unnamed: 2	Unnamed: 3	Unnamed: 4	Unnamed: 5	Unnamed: 6	Unnamed: 7	Unnamed: 8	Unnamed: 9	...	Unnamed: 16	Unnamed: 17	Unnamed: 18	Unnamed: 19
0	NaN	Observatorio de Noticias en Tecnología	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN
1	NaN	#	Fecha Actualización	Fecha de Noticia	Nombre Estudiante	Link Noticia	Titulo	Resumen (No mayor a 50 palabras)	Categorias (Maximo 4 palabras claves que descr...	NaN	...	NaN	NaN	NaN	NaN
2	NaN	1	17/11/2023	10/5/2023	Maria Camila Prada	https://news.un.org/es/story/2023/05/1520892	Las tecnologías digitales son una herramienta ...	El artículo de la ONU informa sobre la crisis ...	Crisis humanitaria en Yemen\n Falta de fondos\...	NaN	...	NaN	NaN	NaN	NaN
3	NaN	2	24/08/2023	05-06-23	Alexander Victoria Garcia	https://elpais.com/tecnologia/2023-06-06/ibm-a...	IBM anuncia su primer centro de datos cuántico...	La computación cuántica encuentra viabilidad a...	IBM, Computacion Cuantica, red	NaN	...	NaN	NaN	NaN	NaN
4	NaN	3	05/09/2023	28/06/2023	Ana María Cruz Pacheco	https://www.bbc.com/news/science-environment-5...	What is net zero and how are the UK and other ...	\nEl Reino Unido podría quedarse rezagado en	Captura y almacenamiento de carbono, gases de ...	NaN	...	NaN	NaN	NaN	NaN

Se genera visualización del data frame:

```
[50]  # Convertir el DataFrame de pandas a una lista de diccionarios
      datos_para_insertar = df.to_dict(orient='records')

      # Ahora puedes usar 'insert_many' con esta lista de diccionarios
      coleccion.insert_many(datos_para_insertar)
```

```
InsertManyResult([ObjectId('65637afc377dc27ebee2cbe9'), ObjectId('65637afc377dc27ebee2cbea'), ObjectId('65637afc377dc27ebee2cbeb'), ObjectId('65637afc377dc27ebee2cbec'),
ObjectId('65637afc377dc27ebee2cbed'), ObjectId('65637afc377dc27ebee2cbee'), ObjectId('65637afc377dc27ebee2cbef'), ObjectId('65637afc377dc27ebee2cbf0'), ObjectId('65637afc377dc27ebee2cbf1'),
ObjectId('65637afc377dc27ebee2cbf2'), ObjectId('65637afc377dc27ebee2cbf3'), ObjectId('65637afc377dc27ebee2cbf4'), ObjectId('65637afc377dc27ebee2cbf5'), ObjectId('65637afc377dc27ebee2cbf6'),
ObjectId('65637afc377dc27ebee2cbf7'), ObjectId('65637afc377dc27ebee2cbf8'), ObjectId('65637afc377dc27ebee2cbf9'), ObjectId('65637afc377dc27ebee2cbfa'), ObjectId('65637afc377dc27ebee2cbfb'),
ObjectId('65637afc377dc27ebee2cbfc'), ObjectId('65637afc377dc27ebee2cbfd'), ObjectId('65637afc377dc27ebee2cbfe'), ObjectId('65637afc377dc27ebee2cbff'), ObjectId('65637afc377dc27ebee2cc00'),
ObjectId('65637afc377dc27ebee2cc01'), ObjectId('65637afc377dc27ebee2cc02'), ObjectId('65637afc377dc27ebee2cc03'), ObjectId('65637afc377dc27ebee2cc04'), ObjectId('65637afc377dc27ebee2cc05'),
ObjectId('65637afc377dc27ebee2cc06'), ObjectId('65637afc377dc27ebee2cc07'), ObjectId('65637afc377dc27ebee2cc08'), ObjectId('65637afc377dc27ebee2cc09'), ObjectId('65637afc377dc27ebee2cc0a'),
ObjectId('65637afc377dc27ebee2cc0b'), ObjectId('65637afc377dc27ebee2cc0c'), ObjectId('65637afc377dc27ebee2cc0d'), ObjectId('65637afc377dc27ebee2cc0e'), ObjectId('65637afc377dc27ebee2cc0f'),
ObjectId('65637afc377dc27ebee2cc10'), ObjectId('65637afc377dc27ebee2cc11'), ObjectId('65637afc377dc27ebee2cc12'), ObjectId('65637afc377dc27ebee2cc13'), ObjectId('65637afc377dc27ebee2cc14'),
ObjectId('65637afc377dc27ebee2cc15'), ObjectId('65637afc377dc27ebee2cc16'), ObjectId('65637afc377dc27ebee2cc17'), ObjectId('65637afc377dc27ebee2cc18'), ObjectId('65637afc377dc27ebee2cc19'),
ObjectId('65637afc377dc27ebee2cc1a'), ObjectId('65637afc377dc27ebee2cc1b'), ObjectId('65637afc377dc27ebee2cc1c'), ObjectId('65637afc377dc27ebee2cc1d'), ObjectId('65637afc377dc27ebee2cc1e'),
ObjectId('65637afc377dc27ebee2cc1f'), ObjectId('65637afc377dc27ebee2cc20'), ObjectId('65637afc377dc27ebee2cc21'), ObjectId('65637afc377dc27ebee2cc22'), ObjectId('65637afc377dc27ebee2cc23'),
ObjectId('65637afc377dc27ebee2cc24'), ObjectId('65637afc377dc27ebee2cc25'), ObjectId('65637afc377dc27ebee2cc26'), ObjectId('65637afc377dc27ebee2cc27'), ObjectId('65637afc377dc27ebee2cc28'),
ObjectId('65637afc377dc27ebee2cc29'), ObjectId('65637afc377dc27ebee2cc2a'), ObjectId('65637afc377dc27ebee2cc2b'), ObjectId('65637afc377dc27ebee2cc2c'), ObjectId('65637afc377dc27ebee2cc2d'),
ObjectId('65637afc377dc27ebee2cc2e'), ObjectId('65637afc377dc27ebee2cc2f'), ObjectId('65637afc377dc27ebee2cc30'), ObjectId('65637afc377dc27ebee2cc31'), ObjectId('65637afc377dc27ebee2cc32'),
ObjectId('65637afc377dc27ebee2cc33'), ObjectId('65637afc377dc27ebee2cc34'), ObjectId('65637afc377dc27ebee2cc35'), ObjectId('65637afc377dc27ebee2cc36'), ObjectId('65637afc377dc27ebee2cc37'),
ObjectId('65637afc377dc27ebee2cc38'), ObjectId('65637afc377dc27ebee2cc39'), ObjectId('65637afc377dc27ebee2cc3a'), ObjectId('65637afc377dc27ebee2cc3b'), ObjectId('65637afc377dc27ebee2cc3c'),
ObjectId('65637afc377dc27ebee2cc3d'), ObjectId('65637afc377dc27ebee2cc3e'), ObjectId('65637afc377dc27ebee2cc3f'), ObjectId('65637afc377dc27ebee2cc40'), ObjectId('65637afc377dc27ebee2cc41'),
ObjectId('65637afc377dc27ebee2cc42'), ObjectId('65637afc377dc27ebee2cc43'), ObjectId('65637afc377dc27ebee2cc44'), ObjectId('65637afc377dc27ebee2cc45'), ObjectId('65637afc377dc27ebee2cc46'),
ObjectId('65637afc377dc27ebee2cc47'), ObjectId('65637afc377dc27ebee2cc48'), ObjectId('65637afc377dc27ebee2cc49'), ObjectId('65637afc377dc27ebee2cc4a'), ObjectId('65637afc377dc27ebee2cc4b'),
ObjectId('65637afc377dc27ebee2cc4c'), ObjectId('65637afc377dc27ebee2cc4d'), ObjectId('65637afc377dc27ebee2cc4e'), ObjectId('65637afc377dc27ebee2cc4f'), ObjectId('65637afc377dc27ebee2cc50'),
ObjectId('65637afc377dc27ebee2cc51'), ObjectId('65637afc377dc27ebee2cc52'), ObjectId('65637afc377dc27ebee2cc53'), ObjectId('65637afc377dc27ebee2cc54'), ObjectId('65637afc377dc27ebee2cc55'),
ObjectId('65637afc377dc27ebee2cc56'), ObjectId('65637afc377dc27ebee2cc57'), ObjectId('65637afc377dc27ebee2cc58'), ObjectId('65637afc377dc27ebee2cc59'), ObjectId('65637afc377dc27ebee2cc5a'),
ObjectId('65637afc377dc27ebee2cc5b'), ObjectId('65637afc377dc27ebee2cc5c'), ObjectId('65637afc377dc27ebee2cc5d'), ObjectId('65637afc377dc27ebee2cc5e'), ObjectId('65637afc377dc27ebee2cc5f'),
ObjectId('65637afc377dc27ebee2cc60'), ObjectId('65637afc377dc27ebee2cc61'), ObjectId('65637afc377dc27ebee2cc62'), ObjectId('65637afc377dc27ebee2cc63'), ObjectId('65637afc377dc27ebee2cc64'),
ObjectId('65637afc377dc27ebee2cc65'), ObjectId('65637afc377dc27ebee2cc66'), ObjectId('65637afc377dc27ebee2cc67'), ObjectId('65637afc377dc27ebee2cc68'), ObjectId('65637afc377dc27ebee2cc69'),
ObjectId('65637afc377dc27ebee2cc6a'), ObjectId('65637afc377dc27ebee2cc6b'), ObjectId('65637afc377dc27ebee2cc6c'), ObjectId('65637afc377dc27ebee2cc6d'), ObjectId('65637afc377dc27ebee2cc6e'),
ObjectId('65637afc377dc27ebee2cc6f'), ObjectId('65637afc377dc27ebee2cc70'), ObjectId('65637afc377dc27ebee2cc71'), ObjectId('65637afc377dc27ebee2cc72'), ObjectId('65637afc377dc27ebee2cc73'),
ObjectId('65637afc377dc27ebee2cc74'), ObjectId('65637afc377dc27ebee2cc75'), ObjectId('65637afc377dc27ebee2cc76'), ObjectId('65637afc377dc27ebee2cc77'), ObjectId('65637afc377dc27ebee2cc78'),
ObjectId('65637afc377dc27ebee2cc79'), ObjectId('65637afc377dc27ebee2cc7a'), ObjectId('65637afc377dc27ebee2cc7b'), ObjectId('65637afc377dc27ebee2cc7c'), ObjectId('65637afc377dc27ebee2cc7d'),
ObjectId('65637afc377dc27ebee2cc7e'), ObjectId('65637afc377dc27ebee2cc7f'), ObjectId('65637afc377dc27ebee2cc80'), ObjectId('65637afc377dc27ebee2cc81'), ObjectId('65637afc377dc27ebee2cc82'),
ObjectId('65637afc377dc27ebee2cc83'), ObjectId('65637afc377dc27ebee2cc84'), ObjectId('65637afc377dc27ebee2cc85'), ObjectId('65637afc377dc27ebee2cc86'), ObjectId('65637afc377dc27ebee2cc87'),
ObjectId('65637afc377dc27ebee2cc88'), ObjectId('65637afc377dc27ebee2cc89'), ObjectId('65637afc377dc27ebee2cc8a'), ObjectId('65637afc377dc27ebee2cc8b'), ObjectId('65637afc377dc27ebee2cc8c'),
```

Se verifica en Mongo, y se muestra la data cargada: