NOMBRE COMPLETO: Ing. Hilaría Adima Vásquez Durán

NAO ID: 3033

FECHA: 17de octubre de 2024

NOMBRE DE LA TRAYECTORIA

EN LA QUE ESTÁS ENROLADO: DATA ANALYST CORE

Título del Reto: MySQL/SQL Server en la gestión

de bases de datos relacionales

Sprint 2 - MySQL/SQL Server en la gestión de bases de datos relacionales

- 1. Crea una base de datos, utilizando código SQL, diseñada para permitir las siguientes acciones con la información extraída de Python:
 - Lectura
 - Procesamiento
 - Análisis
 - Almacenamiento
 - -- 1. Eliminar la base de datos si ya existe

```
DROP DATABASE IF EXISTS tweets_db;
```

-- 2. Crear la base de datos llamada "tweets_db"

```
CREATE DATABASE IF NOT EXISTS tweets_db

CHARACTER SET utf8mb4

COLLATE utf8mb4_unicode_ci;
```

-- 3. Seleccionar la base de datos "tweets_db"

```
USE tweets_db;
```

-- 4. Crear la tabla "tweets" con la estructura adecuada

```
CREATE TABLE IF NOT EXISTS tweets (
```

id INT AUTO_INCREMENT PRIMARY KEY, -- Clave primaria id_tweet BIGINT NOT NULL, -- ID original del tweet usuario VARCHAR (255) NOT NULL, -- Nombre de usuario que publicó el tweet

texto TEXT, --Contenido del tweet (puede ser NULL) fecha DATETIME NOT NULL, - Fecha y hora del tweet retweets INT DEFAULT 0, -- Número d e retweets (por defecto 0).

favoritos INT DEFAULT 0, -- Número de favoritos (por defecto 0)

hashtags TEXT, -- Hashtags asociados al tweet (puede ser NULL)

INDEX idx_usuario (usuario), -- Índice p ara búsquedas por usuario

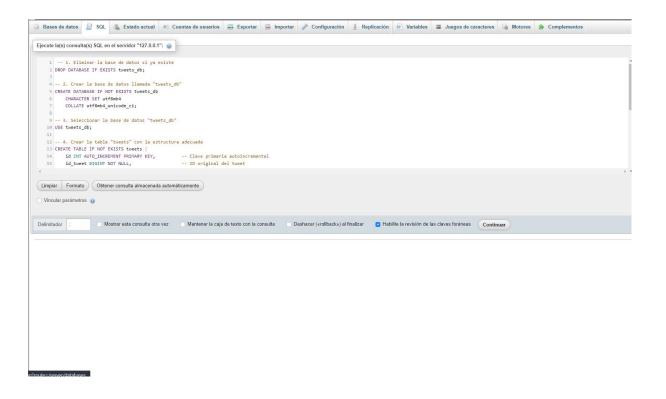
INDEX idx_fecha (fecha), -- Índice p ara búsquedas por fecha

INDEX idx_hashtags (hashtags(255)) -- Índice para búsquedas
por hashtags (limitado a los primeros 255 car acteres))
ENGINE=InnoDB;

-- 5. Insertar datos de ejemplo para verificar la estructura

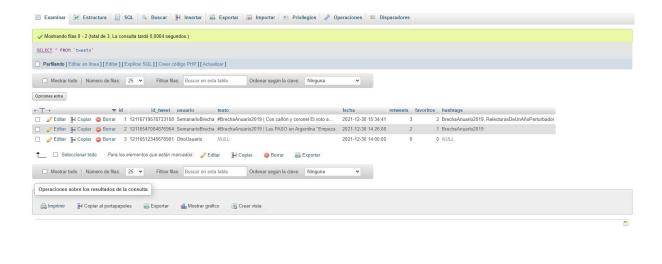
INSERT INTO tweets (id_tweet, usuario, texto, fecha,
retwee ts, favoritos, hashtags)
VALUES

('12116719578733158', 'SemanarioBrecha', '#BrechaAnuario201 9 | Con cañón y coronel El voto a Cabildo Abierto en la periferia de Montevideo. por Venancio Acosta https://brecha.co m.uy/con-canon-ycoronel-2/... #RelecturasDeUnAñoPerturbado r', '2021-12-30 15:34:41', 3, 2, 'BrechaAnuario2019, Relect arasDeUnAñoPerturbador'), ('12116547084876964', 'SemanarioBrecha', '#BrechaAnuario201 9 | Las PASO en Argentina "Empezamos a ver, después de unos años catastróficos para la izquierda sudamericana, que es posible montar defensas, contraataques, replanteos estratégicos, nuevas alianzas, actores y lenguajes." por @dedelacost e https://brecha.com.uy/vamos-a-volver-2/...', '2021-12-30 1 4:26:08', 2, 1, 'BrechaAnuario2019'), ('12116512345678901', 'OtroUsuario', NULL, '2021-12-30 14:0 0:00', 0, 0, NULL); -- Ejemplo con valores NULL









■ Consola

#Subir los datos con python

```
import pandas as pd
import pymysql
```

Ruta del archivo CSV

file path = r'C:\Users\myvlad\Desktop\Udemy courses\FreeLan
ce\tweets extraction.csv'

def leer csv(file path):

11 11 11

#Función para leer el archivo CSV y devolverlo como un Maneja excepciones en caso de que haya problemas con el

Parámetros:

file path (str): Ruta del archivo CSV.

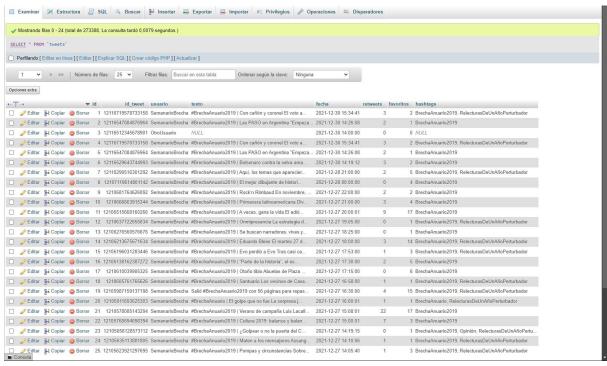
```
Retorna:
    DataFrame: Datos leídos del archivo CSV.
    try:
     df = pd.read csv(file path)
     except FileNotFoundError as e:
        print (f"Error: No se encontró el archivo en la
ruta
        especificada: {e}")
        exit()
     except pd.errors.EmptyDataError as e:
        print(f"Error: El archivo está vacío: {e}")
        exit()
     except pd.errors.ParserError as e:
        print(f"Error al analizar el archivo CSV: {e}")
        exit()
         return df
def insertar datos(df, connection):
  11 11 11
Función para insertar los datos del DataFrame en la bas
e de datos MySQL.
    Parámetros:
    df (DataFrame): DataFrame que contiene los datos del
CSV.
    connection: //Conexión activa a la base de datos
MySQL.
    ** ** **
    try:
         with connection.cursor() as cursor:
  # Insertar los datos de CSV en la tabla 'tweet s'
        insert query = """
        INSERT INTO tweets (id tweet, usuario, texto,
        fecha, retweets, favoritos, hashtags)
        VALUES (%s, %s, %s, %s, %s, %s)
        ** ** **
```

```
for , row in df.iterrows():
              cursor.execute(insert query, (
            row['ID'], # id tweet del tweet
            row['Usuario'], row['Texto'],
            row['Fecha'], row['Retweets'],
            row['Favoritos'], row['Hashtags']
                 ) )
        # Confirmar los cambios en la base de datos
          connection.commit()
          print ("Datos insertados correctamente en la
     base de datos.")
      except pymysql.MySQLError as e:
        print(f"Error al insertar los datos: {e}")
        connection.rollback()
   def main():
   11 11 11
   #Función principal que ejecuta la lectura del archivo
CSV
   y la inserción de datos en la base de datos MySQL.
     # Leer el archivo CSV
     df = leer csv(file path)
    # Reemplazar NaN por None
    df = df.where(pd.notnull(df), None)
    # Conectar a la base de datos MySQL
      connection = None
                           try:
```

```
connection = pymysql.connect(
            host="127.0.0.1",
            user='root',
            password="", # Asegúrate de ingresar tu contra
                           seña si la tienes
            database='tweets db',
            charset='utf8mb4',
            cursorclass=pymysql.cursors.DictCursor
        )
        # Insertar los datos en la base de datos
        insertar datos(df, connection)
    finally:
        # Cerrar la conexión a la base de datos
        if connection:
            connection.close()
         print("Conexión a la base de datos cerrada.")
if __name__ == "__main__":
   main()
```

```
## upinch benefity 3 @ ben_cov
| Import prints as pf
| Import prin
```





Consultas básicas



