**Report: Data Migration Between MySQL and PostgreSQL**

**Project Overview**

This project involves migrating data from a MySQL database to a PostgreSQL database while ensuring data integrity. The goal is to demonstrate how to handle schema migration, data transfer, and validation programmatically using Python.

**Technologies Used**

* **Databases**: MySQL and PostgreSQL
* **Programming Language**: Python
* **Libraries**:
  + pymysql for connecting to MySQL
  + psycopg2 for connecting to PostgreSQL
  + pandas for handling dataframes
  + sqlalchemy for database engines

**Database Details**

**MySQL Database**

* **Host**: localhost
* **Username**: …..
* **Password** ……
* **Database Name**: source\_db
* **Table**: employees

**Sample Schema and Data:**

CREATE TABLE employees (

id INT PRIMARY KEY AUTO\_INCREMENT,

name VARCHAR(100),

age INT,

department VARCHAR(50)

);

INSERT INTO employees (name, age, department)

VALUES

('Alice', 30, 'HR'),

('Bob', 25, 'Engineering'),

('Charlie', 28, 'Marketing');

**PostgreSQL Database**

* **Host**: localhost
* **Username**:……..
* **Password**: ………
* **Database Name**: target\_db
* **Table**: employees

**Sample Schema:**

CREATE TABLE employees (

id SERIAL PRIMARY KEY,

name VARCHAR(100),

age INT,

department VARCHAR(50)

);

**Steps Followed**

**Step 1: Install Required Libraries**

Installed the necessary Python libraries using the command:

pip install pymysql psycopg2 pandas sqlalchemy

**Step 2: Create SQLAlchemy Engines**

Used the following connection strings for MySQL and PostgreSQL:

mysql\_engine = create\_engine("mysql+pymysql://root:shakira@localhost/source\_db")

postgres\_engine = create\_engine("postgresql+psycopg2://postgres:shakisiru@localhost/target\_db")

**Step 3: Migrate Data Using Python**

The Python script performed the following actions:

1. Connected to both MySQL and PostgreSQL databases.
2. Retrieved data from the MySQL employees table.
3. Inserted the data into the PostgreSQL employees table.
4. Verified the integrity by comparing row counts.

**Python Code:**

import pymysql

import psycopg2

import pandas as pd

from sqlalchemy import create\_engine

# Create database connections

mysql\_engine = create\_engine("mysql+pymysql://root:shakira@localhost/Hotelbooking")

postgres\_engine = create\_engine("postgresql+psycopg2://postgres:shakisiru@localhost/testdb")

# Read data from MySQL

df = pd.read\_sql("SELECT \* FROM employees", mysql\_engine)

# Write data to PostgreSQL

df.to\_sql("employees", postgres\_engine, if\_exists="append", index=False)

# Verify data integrity

mysql\_count = len(df)

with postgres\_engine.connect() as conn:

postgres\_count = conn.execute("SELECT COUNT(\*) FROM employees").scalar()

if mysql\_count == postgres\_count:

print("Data migrated successfully with integrity!")

else:

print("Data mismatch detected!")

**Step 4: Validate Migration**

The migrated data was validated using SQL queries in both databases:

**MySQL:**

SELECT \* FROM employees;

**PostgreSQL:**

SELECT \* FROM employees;

Expected Output:

| **id** | **name** | **age** | **department** |
| --- | --- | --- | --- |
| 1 | Alice | 30 | HR |
| 2 | Bob | 25 | Engineering |
| 3 | Charlie | 28 | Marketing |

**Results**

* Data was successfully migrated from MySQL to PostgreSQL.
* Data integrity was confirmed with identical row counts in both databases.
* The employees table in PostgreSQL now contains the following data:

| **id** | **name** | **age** | **department** |
| --- | --- | --- | --- |
| 1 | Alice | 30 | HR |
| 2 | Bob | 25 | Engineering |
| 3 | Charlie | 28 | Marketing |

**Conclusion**

This project demonstrates an effective approach to migrating data between databases. The Python-based solution is scalable and can be adapted for more complex migrations involving larger datasets or additional transformations.

**Future Improvements**

1. Automate the process with a scheduled task or job.
2. Implement more robust logging and monitoring for large-scale migrations.
3. Extend the script to handle schema creation dynamically.

**Prepared by:** shakira