1. Introduction

This report details the process of importing data from a CSV file into a PostgreSQL database using Python. The process involves reading the CSV file, establishing a connection with PostgreSQL, and inserting the data into a pre-existing table using the Pandas and SQLAlchemy libraries. The data import was validated using DBeaver.

2. Environment Details

- **Operating System:** Windows/Linux (User-specific)
- **Python Version:** 3.x
- **Required Libraries:** pandas, psycopg2, sqlalchemy
- **Database Management Tool:** DBeaver
- **Database Engine:** PostgreSQL
- **Server IP:** 18.132.73.146
- **Database Name:** testdb
- **User:** consultants

3. Table Structure in PostgreSQL

The target table `emp_data` was created in PostgreSQL using the following SQL command:

```
CREATE TABLE IF NOT EXISTS emp_data (
   id INTEGER PRIMARY KEY,
   name TEXT,
   dept_id INTEGER
);
```

4. Python Script to Import Data

The following Python script was used to import CSV data into PostgreSQL:

```
import pandas as pd
import psycopg2
from sqlalchemy import create_engine

# Load the CSV file
df = pd.read_csv("D:/Demo files/managedfile.csv")
print(df.head())

# PostgreSQL connection details
PUBLIC_IP = "18.132.73.146"
USERNAME = "consultants"
PASSWORD = "WelcomeItc@2022"
DB_NAME = "testdb"
PORT = "5432"
```

```
# Establish connection using psycopg2
try:
    connection = psycopg2.connect(
        host=PUBLIC_IP,
        database=DB_NAME,
        user=USERNAME,
        password=PASSWORD,
        port=PORT
    )
    print("Connected to the PostgreSQL database successfully!")
except Exception as e:
    print("Failed to connect to the PostgreSQL database!")
    print(e)

# Establish connection using SQLAlchemy
engine = create_engine('postgresql://consultants:WelcomeItc%402022@18.132.73.146:5432/testdb')
print("Database connection established.")

# Import data into PostgreSQL
df.to_sql('emp_data', engine, index=False, if_exists='replace')
print("Data import completed.")
```

5. Validation Steps in DBeaver

To validate the data import, the following SQL command was executed in DBeaver:

```
SELECT * FROM emp_data;
```

6. Observations and Issues Encountered

- The database connection was successfully established.
- The CSV file was loaded correctly.
- The data was successfully inserted into the `emp_data` table.
- The `if_exists='replace'` parameter in `to_sql()` replaced the table, potentially deleting existing records.
- If the table should be preserved, consider using `if_exists='append'` instead.

7. Recommendations

- Use environment variables or a configuration file to store database credentials instead of hardcoding them.

- Ensure that the CSV file contains unique IDs to avoid primary key violations.
- If replacing the table is not intended, change `if_exists='replace'` to `if_exists='append'`.

8. Conclusion

The CSV file was successfully imported into the PostgreSQL database, and the data was verified using DBeaver. The process was automated using Python, ensuring efficiency in future data imports.

End of Report