Standardizing DataFrame Management with SQLite in Python: A Student's Guide

Introduction

This tutorial aims to help students learn how to use Python to create a schema in SQLite, load DataFrames into an SQLite table, and verify their data. We will use a practical example involving Pokémon data.

Prerequisites

- Python installed on your system.
- Basic understanding of Python, pandas, and SQL.

Step 1: Setting Up Your Python Environment

First, import the required libraries:

```
import sqlite3
import pandas as pd
```

Step 2: Creating a Database Connection

Establish a connection to your SQLite database:

```
conn = sqlite3.connect('Pokemon_data.sqlite')
```

Step 3: Creating a Table in SQLite

Define a schema for your table:

```
query = '''
CREATE TABLE IF NOT EXISTS Pokemon (
    Pokemon_name TEXT,
    Fighting_type TEXT,
```

```
    Growth_Rate TEXT,
    Main_ability TEXT,
    Height REAL,
    Weight REAL,
    Attack INT,
    Defense INT,
    Type_of_Move TEXT,
    Region TEXT,
    Latitude REAL,
    Longitude REAL,
    Gender TEXT
)
'''
conn.execute(query)
```

Step 4: Preparing a DataFrame for Uploading

Suppose you have a DataFrame named pokemon_df that contains Pokémon-related data.

Step 5: Loading DataFrame into SQLite

Use the to_sql method to load pokemon_df into the SQLite table:

```
pokemon_df.to_sql('Pokemon', conn, if_exists='replace', index=False)
```

Step 6: Committing and Closing the Database Connection

Always remember to commit your changes and close the connection:

```
conn.commit()
conn.close()
```

Step 7: Verifying Data Upload

Reconnect to the database and run a SELECT query to verify the data:

```
conn = sqlite3.connect('Pokemon_data.sqlite')
df = pd.read_sql_query("SELECT * FROM Pokemon", conn)
print(df.head())
conn.close()
```

Conclusion

This tutorial demonstrates the process of managing and verifying data using pandas and SQLite in Python. It's a fundamental skill for students in data science, allowing efficient data storage and retrieval.

Tips for Further Learning

- Experiment with different SQL queries.
- Explore pandas documentation for advanced DataFrame manipulation.
- Practice with different datasets to get a better understanding.