

```

import pandas as pd

import sqlite3

# Function to create database connection
def create_connection(db_file):
    conn = None
    try:
        conn = sqlite3.connect(db_file)
        return conn
    except sqlite3.Error as e:
        print(e)
    return conn

# Function to create table in SQLite database
def create_table(conn, create_table_sql):
    try:
        c = conn.cursor()
        c.execute(create_table_sql)
    except sqlite3.Error as e:
        print(e)

# Function to insert data into SQLite database
def insert_data(conn, insert_statement, data):
    try:
        c = conn.cursor()
        c.execute(insert_statement, data)
    except sqlite3.Error as e:
        print(e)

# Function to read data from spreadsheet and populate database
def populate_database(db_file, spreadsheet_files):

```

```

# Establish database connection

conn = create_connection(db_file)


# If connection is established, proceed
if conn is not None:

    # Read Spreadsheet 0
    df0 = pd.read_excel(spreadsheet_files[0])


    # Create Product table
    create_product_table_sql = """
CREATE TABLE IF NOT EXISTS Product (
    ProductID INTEGER PRIMARY KEY AUTOINCREMENT,
    Name TEXT NOT NULL,
    Type TEXT,
    ManufacturerID INTEGER,
    Weight REAL,
    Flavor TEXT,
    HealthCondition TEXT,
    Material TEXT,
    Durability TEXT,
    Color TEXT,
    Size TEXT,
    CareInstructions TEXT,
    UNIQUE(Name, ManufacturerID)
);
"""

    create_table(conn, create_product_table_sql)


# Insert data into Product table from Spreadsheet 0
for index, row in df0.iterrows():

    product_data = (

```

```

        row['Name'], row['Type'], row['ManufacturerID'],
        row['Weight'], row['Flavor'], row['HealthCondition'],
        row['Material'], row['Durability'], row['Color'],
        row['Size'], row['CareInstructions']
    )

    insert_product_sql = """
    INSERT OR IGNORE INTO Product (Name, Type, ManufacturerID, Weight, Flavor,
                                   HealthCondition, Material, Durability,
                                   Color, Size, CareInstructions)
    VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?);
    """

    insert_data(conn, insert_product_sql, product_data)


# Read Spreadsheet 1
df1 = pd.read_excel(spreadsheet_files[1])


# Read Spreadsheet 2
df2 = pd.read_excel(spreadsheet_files[2])


# Create Shipment table
create_shipment_table_sql = """
CREATE TABLE IF NOT EXISTS Shipment (
    ShipmentID INTEGER PRIMARY KEY AUTOINCREMENT,
    OriginID INTEGER,
    DestinationID INTEGER,
    Date TEXT,
    FOREIGN KEY (OriginID) REFERENCES Location(LocationID),
    FOREIGN KEY (DestinationID) REFERENCES Location(LocationID)
);
"""

create_table(conn, create_shipment_table_sql)

```

```

# Create ShipmentProduct table

create_shipment_product_table_sql = """
CREATE TABLE IF NOT EXISTS ShipmentProduct (
    ShipmentID INTEGER,
    ProductID INTEGER,
    Quantity INTEGER,
    PRIMARY KEY (ShipmentID, ProductID),
    FOREIGN KEY (ShipmentID) REFERENCES Shipment(ShipmentID),
    FOREIGN KEY (ProductID) REFERENCES Product(ProductID)
);
"""

create_table(conn, create_shipment_product_table_sql)


# Populate Shipment and ShipmentProduct tables from Spreadsheet 1 and 2
for index, row in df1.iterrows():
    shipping_id = row['Shipping Identifier']
    quantity = row['Quantity']
    shipment_date = row['Date']

# Find origin and destination from df2
origin_id = df2.loc[df2['Shipping Identifier'] == shipping_id, 'OriginID'].values[0]
destination_id = df2.loc[df2['Shipping Identifier'] == shipping_id, 'DestinationID'].values[0]

# Insert into Shipment table
insert_shipment_sql = """
INSERT INTO Shipment (OriginID, DestinationID, Date)
VALUES (?, ?, ?);
"""

shipment_data = (origin_id, destination_id, shipment_date)
insert_data(conn, insert_shipment_sql, shipment_data)

```

```

# Insert into ShipmentProduct table

product_data = (row['Name'], row['ManufacturerID'])

select_product_sql = """

SELECT ProductID FROM Product WHERE Name = ? AND ManufacturerID = ?;

"""

c = conn.cursor()

c.execute(select_product_sql, product_data)

product_id = c.fetchone()[0]


insert_shipment_product_sql = """

INSERT INTO ShipmentProduct (ShipmentID, ProductID, Quantity)

VALUES (?, ?, ?);

"""

shipment_product_data = (shipping_id, product_id, quantity)

insert_data(conn, insert_shipment_product_sql, shipment_product_data)


# Commit and close connection

conn.commit()

conn.close()

print("Database populated successfully.")

else:

    print("Error! Cannot create database connection.")


if __name__ == "__main__":

    # Database file path

    database_file = 'walmart_pet_department.db'


# Spreadsheet files

spreadsheet_files = [

    'spreadsheet_0.xlsx',

```

```
'spreadsheet_1.xlsx',  
'spreadsheet_2.xlsx'  
]
```

```
# Populate the SQLite database
```

```
populate_database(database_file, spreadsheet_files)
```