```
!git clone https://github.com/DanielNehemiahPeter/forage-walmart-task-4.git
```

```
Cloning into 'forage-walmart-task-4'...
remote: Enumerating objects: 13, done.
remote: Counting objects: 100% (12/12), done.
remote: Compressing objects: 100% (9/9), done.
remote: Total 13 (delta 1), reused 9 (delta 1), pack-reused 1
Receiving objects: 100% (13/13), 46.63 KiB | 2.74 MiB/s, done.
Resolving deltas: 100% (1/1), done.
```

```
!ls
```

```
Combined.db            path_to_database.db       shipping_data_0.csv  shipping_data_2.csv
forage-walmart-task-4  path_to_your_database.db  shipping_data_1.csv
```

```
cd forage-walmart-task-4/
```

```
/content/forage-walmart-task-4/data/forage-walmart-task-4
```

```
ls
```

```
data/  README.md  shipment_database.db
```

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

```
cd data/
```

```
/content/forage-walmart-task-4/data/forage-walmart-task-4/data
```

```
ls
```

```
shipping_data_0.csv  shipping_data_1.csv  shipping_data_2.csv
```

```
df0 = pd.read_csv('shipping_data_0.csv')
df1 = pd.read_csv('shipping_data_1.csv')
df2 = pd.read_csv('shipping_data_2.csv')
```

```
df0.head()
```

| | origin_warehouse | destination_store | product | on_time | product_quantity | driver_identifier |
|---|---|---|---|---|---|---|
| 0 | d5566b15-b071-4acf-8e8e-c98433083b2d | 50d33715-4c77-4dd9-8b9d-ff1ca372a2a2 | lotion | True | 59 | d8da0460-cf39-4f38-9fff-6c9b4e344d8a |
| 1 | c42f0de8-b4f0-4167-abd1-ae79e5e18eea | 172eb8f3-1033-4fb6-b66b-d0df09df3161 | windows | True | 28 | 293ccaec-6592-4f04-aae5-3e238fe62614 |
| 2 | b145f396-de9b-42f1-9cc9-f5b52c3a941c | 65e4544d-42ae-4751-9580-bdcb90e5fcda | skis | True | 63 | 80988f09-91a3-4e1b-8e69-13551c53f318 |
| | f4372224-759f-43b3-bc83- | 745bee4e-710c-4538-8df1- | | | | 5f79b402-655f-4d8e-8ff3- |

Next steps: Generate code with df0 View recommended plots

```
df2.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 20 entries, 0 to 19
Data columns (total 4 columns):
 #   Column               Non-Null Count  Dtype 
---  ------               --------------  ----- 
 0   shipment_identifier  20 non-null     object
 1   origin_warehouse     20 non-null     object
 2   destination_store    20 non-null     object
 3   driver_identifier    20 non-null     object
dtypes: object(4)
memory usage: 768.0+ bytes
```

```
df1.head()
```

| | shipment_identifier | product | on_time |
|---|---|---|---|
| 0 | 449263b4-6c93-4f19-8b6a-0d99a29fc637 | pants | False |
| 1 | 449263b4-6c93-4f19-8b6a-0d99a29fc637 | pants | False |
| 2 | 449263b4-6c93-4f19-8b6a-0d99a29fc637 | pants | False |
| 3 | 449263b4-6c93-4f19-8b6a-0d99a29fc637 | keyboards | False |
| 4 | 449263b4-6c93-4f19-8b6a-0d99a29fc637 | keyboards | False |

Next steps: Generate code with df1 View recommended plots

```
df2.head()
```

| | shipment_identifier | origin_warehouse | destination_store | driver_identifier |
|---|---|---|---|---|
| 0 | 449263b4-6c93-4f19-8b6a-0d99a29fc637 | bb75bf7d-c008-4267-bf92-6089cff5fe56 | 5e9405de-a078-4b00-99c6-96564568b63c | c12025e6-6f9c-4728-8c3c-9f840bde6f1a |
| 1 | 76e5b84a-9d09-4efb-8b43-a0c932b958bb | 372fd2b1-b2a7-4553-b6d7-426a1bc88e56 | e34973c8-9ca9-4a06-b497-7a8b49625fc2 | 85b8d394-a67c-48b6-b1de-55be323ba622 |
| 2 | b541a47d-89b1-4805- | 469d957f-28ef-4eac-956a- | fcadc756-61e9-41bb- | 47bdfc40-f3db-4678- |

Next steps: Generate code with df2 View recommended plots

```
#merge table 1 and 2
df1_2 = df1.merge(df2, on='shipment_identifier')


#example
df1_2['product_count'] = df1_2.groupby('shipment_identifier')['product'].transform('count')


# Establish a connection to the SQLite database
conn = sqlite3.connect('path_to_database.db')
cursor = conn.cursor()


df0.to_sql('table_df0', conn, if_exists='append', index=False)
df1_2.to_sql('table_df1_2', conn, if_exists='append', index=False)
```

```
110
```

```
# Commit the changes and close the connection
conn.commit()
conn.close()


import sqlite3
import pandas as pd

# Connect to the SQLite database
conn = sqlite3.connect('path_to_database.db')

# Query to select all data from the tables
df0_data = pd.read_sql_query("SELECT * FROM table_df0", conn)
df1_2_data = pd.read_sql_query("SELECT * FROM table_df1_2", conn)

# Display the dataframes
print("Data in df0 table:")
print(df0_data)
print("\nData in df1_2 table:")
print(df1_2_data)
```

```
Data in df0 table:
                        origin_warehouse  \
0    d5566b15-b071-4acf-8e8e-c98433083b2d
1    c42f0de8-b4f0-4167-abd1-ae79e5e18eea
2    b145f396-de9b-42f1-9cc9-f5b52c3a941c
3    f4372224-759f-43b3-bc83-ca6106bba1af
4    49d0edae-9091-41bb-a08d-ab1c66bd08d5
```

```
..                                    ...
105  d2ee1b75-2218-4753-9487-dcca23d667c6
106  6a6d3fce-c5aa-4154-a6a3-b56cb41f709f
107  b19cec0d-357e-4c6b-9257-8be52b1c71b5
108  d2a2460e-00d1-41f2-84cc-eba01eb88d75
109  75891066-59b4-437b-951f-ec553fb26b94

                        destination_store            product  on_time  \
0    50d33715-4c77-4dd9-8b9d-ff1ca372a2a2             lotion        1
1    172eb8f3-1033-4fb6-b66b-d0df09df3161            windows        1
2    65e4544d-42ae-4751-9580-bdcb90e5fcda               skis        1
3    745bee4e-710c-4538-8df1-5c146e1092a6              bikes        1
4    425b7a1a-b744-4c6b-898e-d424dd8cf18e              candy        0
..                                    ...                ...      ...
105  0a994581-341f-43bf-979d-ece1e58de7ec              paint        1
106  403bf915-a897-4918-933b-3996e144e960             snakes        0
107  d3b17672-60fb-443f-a047-2c379132dcb1        alternators        0
108  b9f78d5b-79ae-441e-9dbf-592767af34a5  pencil sharpeners        0
109  28fff0d2-38ea-40a7-b2ef-c2a2f7e69370             apples        0

     product_quantity                    driver_identifier
0                  59  d8da0460-cf39-4f38-9fff-6c9b4e344d8a
1                  28  293ccaec-6592-4f04-aae5-3e238fe62614
2                  63  80988f09-91a3-4e1b-8e69-13551c53f318
3                  47  5f79b402-655f-4d8e-8ff3-5ef05870e0ad
4                  73  58beb5d3-98f8-4077-a964-1f04f7cb11e5
..                ...                                   ...
105                95  a9784b8d-d222-4cdf-93fb-b3886c8033c5
106                54  2fd9a976-bac5-4803-be43-bf93cc618ad1
107                20  45c9bd5b-caf6-4ec1-b1eb-09fe615fbdc6
108                 7  d7432792-20ad-4a7f-a395-81f04fee89fe
109                35  cebc86e8-c327-46f7-96b3-35684d169455

[110 rows x 6 columns]

Data in df1_2 table:
                      shipment_identifier       product  on_time  \
0    449263b4-6c93-4f19-8b6a-0d99a29fc637         pants        0
1    449263b4-6c93-4f19-8b6a-0d99a29fc637         pants        0
2    449263b4-6c93-4f19-8b6a-0d99a29fc637         pants        0
3    449263b4-6c93-4f19-8b6a-0d99a29fc637     keyboards        0
4    449263b4-6c93-4f19-8b6a-0d99a29fc637     keyboards        0
..                                    ...           ...      ...
105  c2237ca1-b7e3-40ab-b798-e1ea469301dc     keyboards        1
106  cfa8a834-54bd-4f47-99ca-8912df32913b  animal masks        0
107  cfa8a834-54bd-4f47-99ca-8912df32913b     furniture        0
108  cfa8a834-54bd-4f47-99ca-8912df32913b     furniture        0
109  cfa8a834-54bd-4f47-99ca-8912df32913b     furniture        0

                        origin_warehouse  \
0    bb75bf7d-c008-4267-bf92-6089cff5fe56
```

```
# Close the database connection
conn.close()
```

Start coding or generate with AI.