

# Database Integration with SQLite 3

- By Sangini Phansekar

## 1. Introduction to the Program

This document provides a step-by-step guide to integrating Flipkart product data with SQLite3. It details how to create a database, define a table, load data from a CSV file, insert records, and execute queries for data retrieval and analysis.

## 2. Understanding the Input Data

The input data is a cleaned CSV file containing Flipkart product details. The dataset typically includes:

- Product Name
- Price
- Rating
- Stock Status

This structured format enables efficient data storage and retrieval in SQLite3.

## 3. Sample of Data

Below is a sample representation of the dataset:

Product Name	Price	Rating	Stock Status
--------------	-------	--------	--------------

Phone Model A	15000	4.5	1
---------------	-------	-----	---

Laptop X	55000	4.3	1
----------	-------	-----	---

Headphones Y	2000	3.9	0
--------------	------	-----	---

## 4. Code Walkthrough

### Step 1: Establish Database Connection

- Connect to an SQLite database (creates database.db if not exists).
- Initialize a cursor for executing SQL commands.

```
import sqlite3
```

```
import pandas as pd
```

```
conn = sqlite3.connect("database.db")
```

```
cursor = conn.cursor()
```

## Step 2: Create a Table

- Define a table structure named flipkart\_data.

```
cursor.execute("""
CREATE TABLE IF NOT EXISTS flipkart_data (
    id INTEGER PRIMARY KEY AUTOINCREMENT,
    Product_name TEXT,
    Price FLOAT,
    Rating REAL,
    Stock_status INTEGER
)
""")
conn.commit()
```

## Step 3: Load Data from CSV

- Read the CSV file containing Flipkart product data using Pandas.

```
csv_file = "path/to/flipkart_data.csv" # Update with the correct path
df = pd.read_csv(csv_file)
```

## Step 4: Insert Data into SQLite Table

- Use Pandas' to\_sql method to insert data.

```
df.to_sql("flipkart_data", conn, if_exists="replace", index=False)
```

## Step 5: Query the Database

- Retrieve sample records to verify successful insertion.

```
query = "SELECT * FROM flipkart_data LIMIT 5"
```

```
result = pd.read_sql_query(query, conn)
```

```
print(result)
```

- Retrieve products with ratings below a threshold.

```
query = "SELECT * FROM flipkart_data WHERE Rating < 80"
```

```
df_low_rating = pd.read_sql_query(query, conn)
```

```
print(df_low_rating)
```

## 5. Conclusion

This process enables seamless integration of Flipkart product data into SQLite3, allowing structured data storage and retrieval. The approach ensures an efficient data pipeline that can be extended for analytics and reporting.