

j3gpctuzz

July 29, 2025

1 Exploratory Data Analysis

Understanding the dataset to explore how the data is present in the database and if there is a need of creating some aggregated tables that can help with:

- Vendor selection for profitability
- Product Pricing Optimization

```
[9]: import pandas as pd
import sqlite3
```

```
[69]: #Creating Database Connection
conn = sqlite3.connect('inventory.db')
```

```
[70]: #checking tables present in database
tables = pd.read_sql_query("SELECT name FROM sqlite_master WHERE type = 'table'", conn)
tables
```

```
[70]:      name
0  begin_inventory
1   end_inventory
2    purchases
3 purchase_prices
4         sales
5  vendor_invoice
```

```
[84]: for table in tables['name']:
    print('-'*50, f'{table}', '-'*50)
    print('Count of records:', pd.read_sql(f"select count(*) as count from {table}", conn)['count'].values[0])
    display(pd.read_sql(f"select * from {table} limit 5", conn))
```

```
----- begin_inventory
-----
```

Count of records: 206529

	InventoryId	Store	City	Brand	Description \
0	1_HARDERSFIELD_58	1	HARDERSFIELD	58	Gekkeikan Black & Gold Sake

1	1_HARDERSFIELD_60	1	HARDERSFIELD	60	Canadian Club 1858 VAP
2	1_HARDERSFIELD_62	1	HARDERSFIELD	62	Herradura Silver Tequila
3	1_HARDERSFIELD_63	1	HARDERSFIELD	63	Herradura Reposado Tequila
4	1_HARDERSFIELD_72	1	HARDERSFIELD	72	No. 3 London Dry Gin

	Size	onHand	Price	startDate
0	750mL	8	12.99	2024-01-01
1	750mL	7	10.99	2024-01-01
2	750mL	6	36.99	2024-01-01
3	750mL	3	38.99	2024-01-01
4	750mL	6	34.99	2024-01-01

----- end_inventory

Count of records: 224489

	InventoryId	Store	City	Brand	Description \
0	1_HARDERSFIELD_58	1	HARDERSFIELD	58	Gekkeikan Black & Gold Sake
1	1_HARDERSFIELD_62	1	HARDERSFIELD	62	Herradura Silver Tequila
2	1_HARDERSFIELD_63	1	HARDERSFIELD	63	Herradura Reposado Tequila
3	1_HARDERSFIELD_72	1	HARDERSFIELD	72	No. 3 London Dry Gin
4	1_HARDERSFIELD_75	1	HARDERSFIELD	75	Three Olives Tomato Vodka

	Size	onHand	Price	endDate
0	750mL	11	12.99	2024-12-31
1	750mL	7	36.99	2024-12-31
2	750mL	7	38.99	2024-12-31
3	750mL	4	34.99	2024-12-31
4	750mL	7	14.99	2024-12-31

----- purchases

Count of records: 2372474

	InventoryId	Store	Brand	Description	Size \
0	69_MOUNTMEND_8412	69	8412	Tequila Ocho Plata Fresno	750mL
1	30_CULCHETH_5255	30	5255	TGI Fridays Ultimte Mudslide	1.75L
2	34_PITMERDEN_5215	34	5215	TGI Fridays Long Island Iced	1.75L
3	1_HARDERSFIELD_5255	1	5255	TGI Fridays Ultimte Mudslide	1.75L
4	76_DONCASTER_2034	76	2034	Glendalough Double Barrel	750mL

	VendorNumber	VendorName	PONumber	PODate \
0	105	ALTAMAR BRANDS LLC	8124	2023-12-21
1	4466	AMERICAN VINTAGE BEVERAGE	8137	2023-12-22
2	4466	AMERICAN VINTAGE BEVERAGE	8137	2023-12-22
3	4466	AMERICAN VINTAGE BEVERAGE	8137	2023-12-22
4	388	ATLANTIC IMPORTING COMPANY	8169	2023-12-24

	ReceivingDate	InvoiceDate	PayDate	PurchasePrice	Quantity	Dollars \
0	2024-01-02	2024-01-04	2024-02-16	35.71	6	214.26

1	2024-01-01	2024-01-07	2024-02-21	9.35	4	37.40
2	2024-01-02	2024-01-07	2024-02-21	9.41	5	47.05
3	2024-01-01	2024-01-07	2024-02-21	9.35	6	56.10
4	2024-01-02	2024-01-09	2024-02-16	21.32	5	106.60

Classification

0	1
1	1
2	1
3	1
4	1

----- purchase_prices

Count of records: 12261

	Brand	Description	Price	Size	Volume	Classification \
0	58	Gekkeikan Black & Gold Sake	12.99	750mL	750	1
1	62	Herradura Silver Tequila	36.99	750mL	750	1
2	63	Herradura Reposado Tequila	38.99	750mL	750	1
3	72	No. 3 London Dry Gin	34.99	750mL	750	1
4	75	Three Olives Tomato Vodka	14.99	750mL	750	1

	PurchasePrice	VendorNumber	VendorName
0	9.28	8320	SHAW ROSS INT L IMP LTD
1	28.67	1128	BROWN-FORMAN CORP
2	30.46	1128	BROWN-FORMAN CORP
3	26.11	9165	ULTRA BEVERAGE COMPANY LLP
4	10.94	7245	PROXIMO SPIRITS INC.

----- sales

Count of records: 12825363

	InventoryId	Store	Brand	Description	Size \
0	1_HARDERSFIELD_1004	1	1004	Jim Beam w/2 Rocks Glasses	750mL
1	1_HARDERSFIELD_1004	1	1004	Jim Beam w/2 Rocks Glasses	750mL
2	1_HARDERSFIELD_1004	1	1004	Jim Beam w/2 Rocks Glasses	750mL
3	1_HARDERSFIELD_1004	1	1004	Jim Beam w/2 Rocks Glasses	750mL
4	1_HARDERSFIELD_1005	1	1005	Maker's Mark Combo Pack	375mL 2 Pk

	SalesQuantity	SalesDollars	SalesPrice	SalesDate	Volume \
0	1	16.49	16.49	2024-01-01	750.0
1	2	32.98	16.49	2024-01-02	750.0
2	1	16.49	16.49	2024-01-03	750.0
3	1	14.49	14.49	2024-01-08	750.0
4	2	69.98	34.99	2024-01-09	375.0

	Classification	ExciseTax	VendorNo	VendorName
0	1	0.79	12546	JIM BEAM BRANDS COMPANY

1	1	1.57	12546	JIM BEAM BRANDS COMPANY
2	1	0.79	12546	JIM BEAM BRANDS COMPANY
3	1	0.79	12546	JIM BEAM BRANDS COMPANY
4	1	0.79	12546	JIM BEAM BRANDS COMPANY

----- vendor_invoice

Count of records: 5543

	VendorNumber	VendorName	InvoiceDate	PONumber	\
0	105	ALTAMAR BRANDS LLC	2024-01-04	8124	
1	4466	AMERICAN VINTAGE BEVERAGE	2024-01-07	8137	
2	388	ATLANTIC IMPORTING COMPANY	2024-01-09	8169	
3	480	BACARDI USA INC	2024-01-12	8106	
4	516	BANFI PRODUCTS CORP	2024-01-07	8170	

	PODate	PayDate	Quantity	Dollars	Freight	Approval
0	2023-12-21	2024-02-16	6	214.26	3.47	None
1	2023-12-22	2024-02-21	15	140.55	8.57	None
2	2023-12-24	2024-02-16	5	106.60	4.61	None
3	2023-12-20	2024-02-05	10100	137483.78	2935.20	None
4	2023-12-24	2024-02-12	1935	15527.25	429.20	None

```
[98]: #Exploring The Data choosing a Vendor
purchases = pd.read_sql_query("select * from purchases where VendorNumber_
↳=4466", conn )
purchases
```

```
[98]:
```

	InventoryId	Store	Brand	Description	Size	\
0	30_CULCHETH_5255	30	5255	TGI Fridays Ultimte Mudslide	1.75L	
1	34_PITMERDEN_5215	34	5215	TGI Fridays Long Island Iced	1.75L	
2	1_HARDERSFIELD_5255	1	5255	TGI Fridays Ultimte Mudslide	1.75L	
3	38_GOULCREST_5215	38	5215	TGI Fridays Long Island Iced	1.75L	
4	59_CLAETHORPES_5215	59	5215	TGI Fridays Long Island Iced	1.75L	
...	
2187	81_PEMBROKE_5215	81	5215	TGI Fridays Long Island Iced	1.75L	
2188	62_KILMARNOCK_5255	62	5255	TGI Fridays Ultimte Mudslide	1.75L	
2189	34_PITMERDEN_5215	34	5215	TGI Fridays Long Island Iced	1.75L	
2190	6_GOULCREST_5215	6	5215	TGI Fridays Long Island Iced	1.75L	
2191	35_HALIVAARA_5255	35	5255	TGI Fridays Ultimte Mudslide	1.75L	

	VendorNumber	VendorName	PONumber	PODate	\
0	4466	AMERICAN VINTAGE BEVERAGE	8137	2023-12-22	
1	4466	AMERICAN VINTAGE BEVERAGE	8137	2023-12-22	
2	4466	AMERICAN VINTAGE BEVERAGE	8137	2023-12-22	
3	4466	AMERICAN VINTAGE BEVERAGE	8207	2023-12-27	
4	4466	AMERICAN VINTAGE BEVERAGE	8207	2023-12-27	
...	

2187	4466	AMERICAN VINTAGE BEVERAGE	13595	2024-12-20
2188	4466	AMERICAN VINTAGE BEVERAGE	13595	2024-12-20
2189	4466	AMERICAN VINTAGE BEVERAGE	13595	2024-12-20
2190	4466	AMERICAN VINTAGE BEVERAGE	13595	2024-12-20
2191	4466	AMERICAN VINTAGE BEVERAGE	13595	2024-12-20

	ReceivingDate	InvoiceDate	PayDate	PurchasePrice	Quantity	Dollars	\
0	2024-01-01	2024-01-07	2024-02-21	9.35	4	37.40	
1	2024-01-02	2024-01-07	2024-02-21	9.41	5	47.05	
2	2024-01-01	2024-01-07	2024-02-21	9.35	6	56.10	
3	2024-01-07	2024-01-19	2024-02-26	9.41	6	56.46	
4	2024-01-05	2024-01-19	2024-02-26	9.41	6	56.46	
...	
2187	2024-12-29	2025-01-04	2025-02-10	9.41	6	56.46	
2188	2024-12-28	2025-01-04	2025-02-10	9.35	5	46.75	
2189	2024-12-28	2025-01-04	2025-02-10	9.41	5	47.05	
2190	2024-12-31	2025-01-04	2025-02-10	9.41	6	56.46	
2191	2024-12-27	2025-01-04	2025-02-10	9.35	5	46.75	

	Classification
0	1
1	1
2	1
3	1
4	1
...	...
2187	1
2188	1
2189	1
2190	1
2191	1

[2192 rows x 16 columns]

```
[102]: purchase_prices = pd.read_sql_query("select * from purchase_prices where_
↳VendorNumber =4466", conn )
purchase_prices
```

	Brand	Description	Price	Size	Volume	Classification	\
0	5215	TGI Fridays Long Island Iced	12.99	1750mL	1750	1	
1	5255	TGI Fridays Ultimte Mudslide	12.99	1750mL	1750	1	
2	3140	TGI Fridays Orange Dream	14.99	1750mL	1750	1	

	PurchasePrice	VendorNumber	VendorName
0	9.41	4466	AMERICAN VINTAGE BEVERAGE
1	9.35	4466	AMERICAN VINTAGE BEVERAGE
2	11.19	4466	AMERICAN VINTAGE BEVERAGE

```
[104]: vendor_invoice = pd.read_sql_query("select * from vendor_invoice where
↳VendorNumber =4466", conn )
vendor_invoice
```

```
[104]:
```

	VendorNumber	VendorName	InvoiceDate	PONumber	\
0	4466	AMERICAN VINTAGE BEVERAGE	2024-01-07	8137	
1	4466	AMERICAN VINTAGE BEVERAGE	2024-01-19	8207	
2	4466	AMERICAN VINTAGE BEVERAGE	2024-01-18	8307	
3	4466	AMERICAN VINTAGE BEVERAGE	2024-01-27	8469	
4	4466	AMERICAN VINTAGE BEVERAGE	2024-02-04	8532	
5	4466	AMERICAN VINTAGE BEVERAGE	2024-02-09	8604	
6	4466	AMERICAN VINTAGE BEVERAGE	2024-02-17	8793	
7	4466	AMERICAN VINTAGE BEVERAGE	2024-03-01	8892	
8	4466	AMERICAN VINTAGE BEVERAGE	2024-03-07	8995	
9	4466	AMERICAN VINTAGE BEVERAGE	2024-03-12	9033	
10	4466	AMERICAN VINTAGE BEVERAGE	2024-03-16	9180	
11	4466	AMERICAN VINTAGE BEVERAGE	2024-03-23	9244	
12	4466	AMERICAN VINTAGE BEVERAGE	2024-03-31	9371	
13	4466	AMERICAN VINTAGE BEVERAGE	2024-04-09	9491	
14	4466	AMERICAN VINTAGE BEVERAGE	2024-04-17	9583	
15	4466	AMERICAN VINTAGE BEVERAGE	2024-04-20	9639	
16	4466	AMERICAN VINTAGE BEVERAGE	2024-04-29	9800	
17	4466	AMERICAN VINTAGE BEVERAGE	2024-05-09	9886	
18	4466	AMERICAN VINTAGE BEVERAGE	2024-05-14	9999	
19	4466	AMERICAN VINTAGE BEVERAGE	2024-05-16	10095	
20	4466	AMERICAN VINTAGE BEVERAGE	2024-05-28	10169	
21	4466	AMERICAN VINTAGE BEVERAGE	2024-06-04	10257	
22	4466	AMERICAN VINTAGE BEVERAGE	2024-06-12	10346	
23	4466	AMERICAN VINTAGE BEVERAGE	2024-06-17	10445	
24	4466	AMERICAN VINTAGE BEVERAGE	2024-06-22	10600	
25	4466	AMERICAN VINTAGE BEVERAGE	2024-06-29	10695	
26	4466	AMERICAN VINTAGE BEVERAGE	2024-07-09	10777	
27	4466	AMERICAN VINTAGE BEVERAGE	2024-07-08	10836	
28	4466	AMERICAN VINTAGE BEVERAGE	2024-07-11	10969	
29	4466	AMERICAN VINTAGE BEVERAGE	2024-07-19	11085	
30	4466	AMERICAN VINTAGE BEVERAGE	2024-07-26	11187	
31	4466	AMERICAN VINTAGE BEVERAGE	2024-08-03	11244	
32	4466	AMERICAN VINTAGE BEVERAGE	2024-08-11	11362	
33	4466	AMERICAN VINTAGE BEVERAGE	2024-08-18	11489	
34	4466	AMERICAN VINTAGE BEVERAGE	2024-08-24	11540	
35	4466	AMERICAN VINTAGE BEVERAGE	2024-09-02	11716	
36	4466	AMERICAN VINTAGE BEVERAGE	2024-09-12	11771	
37	4466	AMERICAN VINTAGE BEVERAGE	2024-09-20	11901	
38	4466	AMERICAN VINTAGE BEVERAGE	2024-09-25	11993	
39	4466	AMERICAN VINTAGE BEVERAGE	2024-10-01	12125	
40	4466	AMERICAN VINTAGE BEVERAGE	2024-10-08	12235	
41	4466	AMERICAN VINTAGE BEVERAGE	2024-10-09	12253	

42	4466	AMERICAN VINTAGE BEVERAGE	2024-10-12	12321
43	4466	AMERICAN VINTAGE BEVERAGE	2024-10-20	12466
44	4466	AMERICAN VINTAGE BEVERAGE	2024-10-27	12515
45	4466	AMERICAN VINTAGE BEVERAGE	2024-11-07	12702
46	4466	AMERICAN VINTAGE BEVERAGE	2024-11-12	12752
47	4466	AMERICAN VINTAGE BEVERAGE	2024-11-20	12828
48	4466	AMERICAN VINTAGE BEVERAGE	2024-11-27	12929
49	4466	AMERICAN VINTAGE BEVERAGE	2024-11-28	13092
50	4466	AMERICAN VINTAGE BEVERAGE	2024-12-06	13134
51	4466	AMERICAN VINTAGE BEVERAGE	2024-12-16	13254
52	4466	AMERICAN VINTAGE BEVERAGE	2024-12-26	13432
53	4466	AMERICAN VINTAGE BEVERAGE	2024-12-30	13483
54	4466	AMERICAN VINTAGE BEVERAGE	2025-01-09	13627

	PODate	PayDate	Quantity	Dollars	Freight	Approval
0	2023-12-22	2024-02-21	15	140.55	8.57	None
1	2023-12-27	2024-02-26	335	3142.33	16.97	None
2	2024-01-03	2024-02-18	41	383.35	1.99	None
3	2024-01-14	2024-03-11	72	673.20	3.30	None
4	2024-01-19	2024-03-15	79	740.21	3.48	None
5	2024-01-24	2024-03-15	347	3261.37	17.61	None
6	2024-02-05	2024-04-02	72	675.36	3.17	None
7	2024-02-12	2024-03-28	117	1096.05	5.15	None
8	2024-02-19	2024-04-02	129	1209.27	5.44	None
9	2024-02-22	2024-04-16	147	1377.87	6.61	None
10	2024-03-03	2024-04-19	211	1979.33	9.50	None
11	2024-03-08	2024-04-21	161	1510.69	6.95	None
12	2024-03-17	2024-05-13	176	1649.20	8.91	None
13	2024-03-24	2024-05-08	215	2016.43	10.08	None
14	2024-03-31	2024-05-12	110	1035.10	5.69	None
15	2024-04-04	2024-06-04	515	5323.01	25.02	None
16	2024-04-15	2024-06-07	275	2775.01	13.60	None
17	2024-04-21	2024-06-12	312	3050.86	13.73	None
18	2024-04-29	2024-06-26	310	3117.82	14.97	None
19	2024-05-06	2024-06-27	215	2123.01	11.46	None
20	2024-05-11	2024-07-04	327	3298.55	15.83	None
21	2024-05-17	2024-07-08	376	3708.18	20.39	None
22	2024-05-23	2024-07-20	640	6458.38	29.71	None
23	2024-05-30	2024-07-19	288	2937.56	15.86	None
24	2024-06-09	2024-08-01	308	3213.28	17.03	None
25	2024-06-16	2024-08-12	143	1585.75	8.09	None
26	2024-06-22	2024-08-15	18	179.88	0.99	None
27	2024-06-25	2024-08-14	8	85.96	0.43	None
28	2024-06-29	2024-08-18	94	993.12	5.06	None
29	2024-07-06	2024-09-04	601	5883.09	31.18	None
30	2024-07-14	2024-08-26	1535	15406.69	77.03	None
31	2024-07-18	2024-09-04	266	2888.22	15.31	None

32	2024-07-26	2024-09-24	206	2171.64	11.08	None
33	2024-08-04	2024-09-16	768	7292.98	37.92	None
34	2024-08-08	2024-10-02	1207	12242.43	64.88	None
35	2024-08-19	2024-09-29	433	4334.43	20.37	None
36	2024-08-23	2024-10-11	370	3640.18	18.20	None
37	2024-09-01	2024-10-30	358	3626.74	16.32	None
38	2024-09-07	2024-10-23	233	2264.85	12.23	None
39	2024-09-16	2024-11-07	284	2813.46	15.47	None
40	2024-09-23	2024-11-20	258	2592.08	11.92	None
41	2024-09-23	2024-11-14	1	11.19	0.05	None
42	2024-09-26	2024-11-19	172	1685.98	7.59	None
43	2024-10-05	2024-11-26	280	2798.60	14.55	None
44	2024-10-09	2024-11-30	178	1763.28	8.64	None
45	2024-10-21	2024-12-11	183	1812.77	8.70	None
46	2024-10-25	2024-12-11	216	2156.16	10.57	None
47	2024-10-30	2024-12-18	262	2597.36	12.99	None
48	2024-11-06	2025-01-04	270	2622.32	12.85	None
49	2024-11-16	2024-12-30	209	2036.17	11.20	None
50	2024-11-20	2025-01-18	305	3018.89	15.70	None
51	2024-11-28	2025-01-13	262	2530.36	11.39	None
52	2024-12-09	2025-01-27	231	2263.43	11.32	None
53	2024-12-13	2025-02-11	221	2178.27	11.33	None
54	2024-12-22	2025-02-05	413	3985.03	19.53	None

```
[108]: sales = pd.read_sql_query("select * from sales where VendorNo =4466", conn )
sales
```

```
[108]:
```

	InventoryId	Store	Brand	Description	Size	\
0	1_HARDERSFIELD_5215	1	5215	TGI Fridays Long Island Iced	1.75L	
1	1_HARDERSFIELD_5215	1	5215	TGI Fridays Long Island Iced	1.75L	
2	1_HARDERSFIELD_5215	1	5215	TGI Fridays Long Island Iced	1.75L	
3	1_HARDERSFIELD_5215	1	5215	TGI Fridays Long Island Iced	1.75L	
4	1_HARDERSFIELD_5215	1	5215	TGI Fridays Long Island Iced	1.75L	
...	
9448	9_BLACKPOOL_5215	9	5215	TGI Fridays Long Island Iced	1.75L	
9449	9_BLACKPOOL_5255	9	5255	TGI Fridays Ultimte Mudslide	1.75L	
9450	9_BLACKPOOL_5255	9	5255	TGI Fridays Ultimte Mudslide	1.75L	
9451	9_BLACKPOOL_5255	9	5255	TGI Fridays Ultimte Mudslide	1.75L	
9452	9_BLACKPOOL_5255	9	5255	TGI Fridays Ultimte Mudslide	1.75L	

	SalesQuantity	SalesDollars	SalesPrice	SalesDate	Volume	\
0	1	12.99	12.99	2024-01-09	1750.0	
1	1	12.99	12.99	2024-01-12	1750.0	
2	1	12.99	12.99	2024-01-15	1750.0	
3	1	12.99	12.99	2024-01-21	1750.0	
4	1	12.99	12.99	2024-01-23	1750.0	
...	

9448	1	12.99	12.99	2024-12-21	1750.0
9449	1	12.99	12.99	2024-12-02	1750.0
9450	1	12.99	12.99	2024-12-09	1750.0
9451	1	12.99	12.99	2024-12-23	1750.0
9452	1	12.99	12.99	2024-12-31	1750.0

	Classification	ExciseTax	VendorNo	VendorName
0	1	1.84	4466	AMERICAN VINTAGE BEVERAGE
1	1	1.84	4466	AMERICAN VINTAGE BEVERAGE
2	1	1.84	4466	AMERICAN VINTAGE BEVERAGE
3	1	1.84	4466	AMERICAN VINTAGE BEVERAGE
4	1	1.84	4466	AMERICAN VINTAGE BEVERAGE
...
9448	1	1.84	4466	AMERICAN VINTAGE BEVERAGE
9449	1	1.84	4466	AMERICAN VINTAGE BEVERAGE
9450	1	1.84	4466	AMERICAN VINTAGE BEVERAGE
9451	1	1.84	4466	AMERICAN VINTAGE BEVERAGE
9452	1	1.84	4466	AMERICAN VINTAGE BEVERAGE

[9453 rows x 14 columns]

```
[110]: purchases.groupby(['Brand', 'PurchasePrice'])[['Quantity', 'Dollars']].sum()
```

```
[110]:
```

		Quantity	Dollars
Brand	PurchasePrice		
3140	11.19	4640	51921.60
5215	9.41	4923	46325.43
5255	9.35	6215	58110.25

```
[112]: purchase_prices
```

```
[112]:
```

	Brand	Description	Price	Size	Volume	Classification	\
0	5215	TGI Fridays Long Island Iced	12.99	1750mL	1750		1
1	5255	TGI Fridays Ultimte Mudslide	12.99	1750mL	1750		1
2	3140	TGI Fridays Orange Dream	14.99	1750mL	1750		1

	PurchasePrice	VendorNumber	VendorName
0	9.41	4466	AMERICAN VINTAGE BEVERAGE
1	9.35	4466	AMERICAN VINTAGE BEVERAGE
2	11.19	4466	AMERICAN VINTAGE BEVERAGE

```
[114]: sales.groupby('Brand')[['SalesDollars', 'SalesPrice', 'SalesQuantity']].sum()
```

```
[114]:
```

	SalesDollars	SalesPrice	SalesQuantity
Brand			
3140	50531.10	30071.85	3890
5215	60416.49	41542.02	4651

5255 79187.04 51180.60 6096

- The purchases table contains actual purchase data, including the date of purchase, products (brands) purchased by vendors, the amount paid (in dollars), and the quantity purchased.
- The purchase price column is derived from the purchase_prices table, which provides product-wise actual and purchase prices. The combination of vendor and brand is unique in this table.
- The vendor_invoice table aggregates data from the purchases table, summarizing quantity and dollar amounts, along with an additional column for freight. This table maintains uniqueness based on vendor and PO number.
-

1.1 The sales table captures actual sales transactions, detailing the brands purchased by vendors, the quantity sold, the selling price, and the revenue earned.

As the data that we need for analysis is distributed in different tables, we need to create a summary table containing:

- purchase transactions made by vendors
- sales transaction data
- freight costs for each vendor
- actual product prices from vendors

```
[125]: vendor_invoice.columns
```

```
[125]: Index(['VendorNumber', 'VendorName', 'InvoiceDate', 'PONumber', 'PODate',  
          'PayDate', 'Quantity', 'Dollars', 'Freight', 'Approval'],  
          dtype='object')
```

```
[137]: freight_summary = pd.read_sql_query("""select VendorNumber,SUM(Freight) as_  
      ↳FreightCost  
      from vendor_invoice  
      Group BY VendorNumber""",conn)
```

```
[139]: freight_summary
```

```
[139]:        VendorNumber   FreightCost  
0                    2           27.08
```

1	54	0.48
2	60	367.52
3	105	62.39
4	200	6.19
..
121	98450	856.02
122	99166	130.09
123	172662	178.34
124	173357	202.50
125	201359	0.09

[126 rows x 2 columns]

```
[155]: pd.read_sql_query("""
SELECT
    p.VendorNumber,
    p.VendorName,
    p.Brand,
    p.PurchasePrice,
    pp.Volume,
    pp.Price AS Actualprice,
    SUM(p.Quantity) AS TotalPurchaseQuantity,
    SUM(p.Dollars) AS TotalPurchaseDollars
FROM purchases p
JOIN purchase_prices pp
    ON p.Brand = pp.Brand
WHERE p.PurchasePrice > 0
GROUP BY
    p.VendorNumber,
    p.VendorName,
    p.Brand,
    p.PurchasePrice,
    pp.Volume,
    pp.Price
ORDER BY TotalPurchaseDollars
""", conn)
```

```
[155]:
```

	VendorNumber	VendorName	Brand	PurchasePrice	Volume	\
0	7245	PROXIMO SPIRITS INC.	3065	0.71	50	
1	3960	DIAGEO NORTH AMERICA INC	6127	1.47	200	
2	3924	HEAVEN HILL DISTILLERIES	9123	0.74	50	
3	8004	SAZERAC CO INC	5683	0.39	50	
4	9815	WINE GROUP INC	8527	1.32	750	
...	
10687	3960	DIAGEO NORTH AMERICA INC	3545	21.89	1750	
10688	3960	DIAGEO NORTH AMERICA INC	4261	16.17	1750	
10689	17035	PERNOD RICARD USA	8068	18.24	1750	

10690	4425	MARTIGNETTI COMPANIES	3405	23.19	1750
10691	1128	BROWN-FORMAN CORP	1233	26.27	1750

	Actualprice	TotalPurchaseQuantity	TotalPurchaseDollars
0	0.99	1	0.71
1	1.99	1	1.47
2	0.99	2	1.48
3	0.49	6	2.34
4	4.99	2	2.64
...
10687	29.99	138109	3023206.01
10688	22.99	201682	3261197.94
10689	24.99	187407	3418303.68
10690	28.99	164038	3804041.22
10691	36.99	145080	3811251.60

[10692 rows x 8 columns]

-Filtered the data where we having Purchase Power Zero

```
[161]: sales.columns
```

```
[161]: Index(['InventoryId', 'Store', 'Brand', 'Description', 'Size', 'SalesQuantity',
          'SalesDollars', 'SalesPrice', 'SalesDate', 'Volume', 'Classification',
          'ExciseTax', 'VendorNo', 'VendorName'],
          dtype='object')
```

```
[3]: import pandas as pd
import sqlite3

conn = sqlite3.connect('inventory.db')
pd.read_sql_query("SELECT name FROM sqlite_master WHERE type='table';", conn)
```

```
[3]:          name
0  begin_inventory
1  end_inventory
2  purchases
3  purchase_prices
4  sales
5  vendor_invoice
```

```
[47]: pd.read_sql_query("SELECT * FROM sales LIMIT 5;", conn)
```

```
[47]:          InventoryId  Store  Brand          Description  Size \
0  1_HARDERSFIELD_1004      1   1004  Jim Beam w/2 Rocks Glasses  750mL
1  1_HARDERSFIELD_1004      1   1004  Jim Beam w/2 Rocks Glasses  750mL
2  1_HARDERSFIELD_1004      1   1004  Jim Beam w/2 Rocks Glasses  750mL
```

3	1_HARDERSFIELD_1004	1	1004	Jim Beam w/2 Rocks Glasses	750mL
4	1_HARDERSFIELD_1005	1	1005	Maker's Mark Combo Pack	375mL 2 Pk

	SalesQuantity	SalesDollars	SalesPrice	SalesDate	Volume \
0	1	16.49	16.49	2024-01-01	750.0
1	2	32.98	16.49	2024-01-02	750.0
2	1	16.49	16.49	2024-01-03	750.0
3	1	14.49	14.49	2024-01-08	750.0
4	2	69.98	34.99	2024-01-09	375.0

	Classification	ExciseTax	VendorNo	VendorName
0	1	0.79	12546	JIM BEAM BRANDS COMPANY
1	1	1.57	12546	JIM BEAM BRANDS COMPANY
2	1	0.79	12546	JIM BEAM BRANDS COMPANY
3	1	0.79	12546	JIM BEAM BRANDS COMPANY
4	1	0.79	12546	JIM BEAM BRANDS COMPANY

```
[57]: df = pd.read_sql_query("""
SELECT
    VendorNo,
    Brand,
    SUM(SalesDollars) AS TotalSalesDollars,
    SUM(SalesPrice) AS TotalSalesPrices,
    SUM(SalesQuantity) AS TotalSalesQuantity,
    SUM(ExciseTax) AS TotalExciseTax
FROM sales
GROUP BY VendorNo, Brand
ORDER BY TotalSalesDollars
""", conn)

pd.concat([df.head(5), df.tail(5)])
```

```
[57]:
```

	VendorNo	Brand	TotalSalesDollars	TotalSalesPrices \
0	8004	5287	0.98	0.98
1	9206	2773	0.99	0.99
2	3252	3933	1.98	0.99
3	3924	9123	1.98	0.99
4	10050	3623	1.98	1.98
11267	3960	3545	4223107.62	545778.28
11268	3960	4261	4475972.88	420050.01
11269	17035	8068	4538120.60	461140.15
11270	4425	3405	4819073.49	561512.37
11271	1128	1233	5101919.51	672819.31

	TotalSalesQuantity	TotalExciseTax
0	2	0.10
1	1	0.05

2	2	0.10
3	2	0.10
4	2	0.10
11267	135838	249587.83
11268	200412	368242.80
11269	187140	343854.07
11270	160247	294438.66
11271	142049	260999.20

```
[5]: vendor_sales_summary = pd.read_sql_query("""
WITH FreightSummary AS (
    SELECT
        VendorNumber,
        SUM(Freight) AS FreightCost
    FROM vendor_invoice
    GROUP BY VendorNumber
),

PurchaseSummary AS (
    SELECT
        p.VendorNumber,
        p.VendorName,
        p.Brand,
        p.Description,
        p.PurchasePrice,
        pp.Price AS ActualPrice,
        pp.Volume,
        SUM(p.Quantity) AS TotalPurchaseQuantity,
        SUM(p.Dollars) AS TotalPurchaseDollars
    FROM purchases p
    JOIN purchase_prices pp
        ON p.Brand = pp.Brand
    WHERE p.PurchasePrice > 0
    GROUP BY p.VendorNumber, p.VendorName, p.Brand, p.Description, p.
    ↪PurchasePrice, pp.Price, pp.Volume
),

SalesSummary AS (
    SELECT
        VendorNo,
        Brand,
        SUM(SalesQuantity) AS TotalSalesQuantity,
        SUM(SalesDollars) AS TotalSalesDollars,
        SUM(SalesPrice) AS TotalSalesPrice,
        SUM(ExciseTax) AS TotalExciseTax
    FROM sales
    GROUP BY VendorNo, Brand
```

```
)

SELECT
    ps.VendorNumber,
    ps.VendorName,
    ps.Brand,
    ps.Description,
    ps.PurchasePrice,
    ps.ActualPrice,
    ps.Volume,
    ps.TotalPurchaseQuantity,
    ps.TotalPurchaseDollars,
    ss.TotalSalesQuantity,
    ss.TotalSalesDollars,
    ss.TotalSalesPrice,
    ss.TotalExciseTax,
    fs.FreightCost
FROM PurchaseSummary ps
LEFT JOIN SalesSummary ss
    ON ps.VendorNumber = ss.VendorNo
    AND ps.Brand = ss.Brand
LEFT JOIN FreightSummary fs
    ON ps.VendorNumber = fs.VendorNumber
ORDER BY ps.TotalPurchaseDollars DESC
""", conn)
```

[7]: vendor_sales_summary

```
[7]:
```

	VendorNumber	VendorName	Brand	\
0	1128	BROWN-FORMAN CORP	1233	
1	4425	MARTIGNETTI COMPANIES	3405	
2	17035	PERNOD RICARD USA	8068	
3	3960	DIAGEO NORTH AMERICA INC	4261	
4	3960	DIAGEO NORTH AMERICA INC	3545	
...	
10687	9815	WINE GROUP INC	8527	
10688	8004	SAZERAC CO INC	5683	
10689	3924	HEAVEN HILL DISTILLERIES	9123	
10690	3960	DIAGEO NORTH AMERICA INC	6127	
10691	7245	PROXIMO SPIRITS INC.	3065	

	Description	PurchasePrice	ActualPrice	Volume	\
0	Jack Daniels No 7 Black	26.27	36.99	1750	
1	Tito's Handmade Vodka	23.19	28.99	1750	
2	Absolut 80 Proof	18.24	24.99	1750	
3	Capt Morgan Spiced Rum	16.17	22.99	1750	
4	Ketel One Vodka	21.89	29.99	1750	

```

...
10687 Concannon Glen Ellen Wh Zin          1.32          4.99       750
10688 Dr McGillicuddy's Apple Pie          0.39          0.49        50
10689          Deep Eddy Vodka              0.74          0.99        50
10690 The Club Strawbry Margarita          1.47          1.99       200
10691    Three Olives Grape Vodka           0.71          0.99        50

      TotalPurchaseQuantity  TotalPurchaseDollars  TotalSalesQuantity \
0                145080                3811251.60                142049.0
1                164038                3804041.22                160247.0
2                187407                3418303.68                187140.0
3                201682                3261197.94                200412.0
4                138109                3023206.01                135838.0
...
10687          2                2.64                5.0
10688          6                2.34               134.0
10689          2                1.48                2.0
10690          1                1.47               72.0
10691          1                0.71               86.0

      TotalSalesDollars  TotalSalesPrice  TotalExciseTax  FreightCost
0          5101919.51          672819.31          260999.20          68601.68
1          4819073.49          561512.37          294438.66          144929.24
2          4538120.60          461140.15          343854.07          123780.22
3          4475972.88          420050.01          368242.80          257032.07
4          4223107.62          545778.28          249587.83          257032.07
...
10687          15.95          10.96           0.55          27100.41
10688          65.66           1.47           7.04          50293.62
10689           1.98           0.99           0.10          14069.87
10690          143.28          77.61          15.12          257032.07
10691           85.14          33.66           4.46          38994.78

```

[10692 rows x 14 columns]

- This query generates a vendor-wise sales and purchase summary, which is valuable for:

- **Performance Optimization:**

- * The query involves heavy joins and aggregations on large datasets like sales and purchases.
- * Storing the pre-aggregated results avoids repeated expensive computations.
- * Helps in analyzing sales, purchases, and pricing for different vendors and brands.

- **Future Benefits of Storing this data:**

- * Useful for faster Dashboarding & Reporting.
- * Instead of running expensive queries each time, dashboards can fetch data quickly from `vendor_sales_summary`.

```
[15]: vendor_sales_summary.dtypes
```



```
[15]: VendorNumber          int64
      VendorName            object
      Brand                 int64
      Description           object
      PurchasePrice         float64
      ActualPrice           float64
      Volume                object
      TotalPurchaseQuantity int64
      TotalPurchaseDollars  float64
      TotalSalesQuantity    float64
      TotalSalesDollars     float64
      TotalSalesPrice       float64
      TotalExciseTax        float64
      FreightCost           float64
      dtype: object
```

```
[19]: vendor_sales_summary.isnull().sum()
```

```
[19]: VendorNumber          0
      VendorName            0
      Brand                 0
      Description           0
      PurchasePrice         0
      ActualPrice           0
      Volume                0
      TotalPurchaseQuantity 0
      TotalPurchaseDollars  0
      TotalSalesQuantity    178
      TotalSalesDollars     178
      TotalSalesPrice       178
      TotalExciseTax        178
      FreightCost           0
      dtype: int64
```

```
[35]: vendor_sales_summary['VendorName'].unique()
```

```
[35]: array(['BROWN-FORMAN CORP', 'MARTIGNETTI COMPANIES', 'PERNOD RICARD USA',
        'DIAGEO NORTH AMERICA INC', 'BACARDI USA INC',
        'JIM BEAM BRANDS COMPANY', 'MAJESTIC FINE WINES',
        'ULTRA BEVERAGE COMPANY LLP', 'STOLI GROUP,(USA) LLC',
        'PROXIMO SPIRITS INC.', 'MOET HENNESSY USA INC', 'CAMPARI AMERICA',
        'SAZERAC CO INC', 'CONSTELLATION BRANDS INC', 'M S WALKER INC',
        'SAZERAC NORTH AMERICA INC.', 'PALM BAY INTERNATIONAL INC',
        'REMY COINTREAU USA INC', 'SIDNEY FRANK IMPORTING CO',
        'E & J GALLO WINERY', 'WILLIAM GRANT & SONS INC',
        'HEAVEN HILL DISTILLERIES', 'DISARONNO INTERNATIONAL LLC',
        'EDRINGTON AMERICAS', 'CASTLE BRANDS CORP.',
```

```

'SOUTHERN WINE & SPIRITS NE', 'STE MICHELLE WINE ESTATES',
'TRINCHERO FAMILY ESTATES', 'MHW LTD', 'WINE GROUP INC',
'PERFECTA WINES', 'LUXCO INC', 'TREASURY WINE ESTATES',
'DIAGEO CHATEAU ESTATE WINES', 'SHAW ROSS INT L IMP LTD',
'PINE STATE TRADING CO', 'PHILLIPS PRODUCTS CO.',
'CALEDONIA SPIRITS INC', 'STATE WINE & SPIRITS',
'KOBAND CORPORATION', 'BANFI PRODUCTS CORP',
'VINEYARD BRANDS INC', 'DELICATO VINEYARDS INC',
'FABRIZIA SPIRITS LLC', 'DUGGANS DISTILLED PRODUCTS',
'Serralles Usa LLC', 'SEA HAGG DISTILLERY LLC',
'OLE SMOKY DISTILLERY LLC', 'VRANKEN AMERICA', 'KLIN SPIRITS LLC',
'LAIRD & CO', 'ADAMBA IMPORTS INTL INC',
'LATITUDE BEVERAGE COMPANY', 'FREDERICK WILDMAN & SONS',
'MCCORMICK DISTILLING CO', 'CHARLES JACQUIN ET CIE INC',
'WESTERN SPIRITS BEVERAGE CO', 'MARSALLE COMPANY',
'AMERICAN VINTAGE BEVERAGE', 'MANGO BOTTLING INC',
'SWEET BABY VINEYARD', 'NICHE W & S', 'LABELLE VYDS AND WINERY',
'FLAG HILL WINERY & VINEYARD', 'SMOKY QUARTZ DISTILLERY LLC',
'PREMIUM PORT WINES, INC.', 'Russian Standard Vodka',
'Dunn Wine Brokers', 'WEIN BAUER INC', 'BULLY BOY DISTILLERS',
'ATLANTIC IMPORTING COMPANY', 'PREMIER DISTRIBUTORS',
'VINILANDIA USA', 'PARK STREET IMPORTS LLC', 'TAKARA SAKE USA INC',
'SEA BREEZE CELLARS LLC', 'STARK BREWING COMPANY', 'TY KU LLC',
'PSP WINES', 'TAMWORTH DISTILLING', 'ZORVINO VINEYARDS',
'SOUTHERN GLAZERS W&S OF NE', 'HOOD RIVER DISTILLERS, Inc.',
'CRUSH WINES', 'POVERTY LANE ORCHARDS', 'DJINN SPIRITS LLC',
'MOONLIGHT MEADERY', 'TALL SHIP DISTILLERY LLC',
'FORTUNE WINE BROKERS LLC', 'BLACK COVE BEVERAGES', 'VINEXTRA INC',
'SURVILLE ENTERPRISES CORP', 'JEWELL TOWNE VINEYARDS',
'SWEETWATER FARM', 'ALTAMAR BRANDS LLC', 'CANDIA VINEYARDS',
'INCREDIBREW INC', 'ALISA CARR BEVERAGES',
'STELLAR IMPORTING CO LLC', 'FULCHINO VINEYARD INC',
'IRA GOLDMAN AND WILLIAMS, LLP', 'Circa Wines',
'VINEDREA WINES LLC', 'BLACK PRINCE DISTILLERY INC',
'VINEYARD BRANDS LLC', 'THE IMPORTED GRAPE LLC',
'WALPOLE MTN VIEW WINERY', 'GILMANTON WINERY & VINEYARD',
'HAUNTING WHISPER VYDS', 'STAR INDUSTRIES INC.',
'LOYAL DOG WINERY', 'R.P.IMPORTS INC', 'THE PIERPONT GROUP LLC',
'APPOLO VINEYARDS LLC', 'BLACK ROCK SPIRITS LLC',
'CENTEUR IMPORTS LLC', 'HIGHLAND WINE MERCHANTS LLC',
'AMERICAN SPIRITS EXCHANGE', 'UNCORKED', 'BRONCO WINE COMPANY',
'MILTONS DISTRIBUTING CO', 'TRUETT HURST', 'LAUREATE IMPORTS CO',
'FANTASY FINE WINES CORP', 'AAPER ALCOHOL & CHEMICAL CO',
'SILVER MOUNTAIN CIDERS', 'CAPSTONE INTERNATIONAL',
'FLAVOR ESSENCE INC'], dtype=object)

```

```
[25]: vendor_sales_summary['Description'].unique()
```

```
[25]: array(['Jack Daniels No 7 Black', "Tito's Handmade Vodka",
        'Absolut 80 Proof', ..., 'Crown Royal Apple',
        'Concannon Glen Ellen Wh Zin', 'The Club Strawbry Margarita'],
        dtype=object)
```

```
[27]: vendor_sales_summary['Volume'] = vendor_sales_summary['Volume'].
        ↳astype('float64')
```

```
[29]: vendor_sales_summary.fillna(0, inplace= True)
```

```
[33]: vendor_sales_summary['VendorName'] =vendor_sales_summary['VendorName'].str.
        ↳strip()
```

```
[37]: vendor_sales_summary['GrossProfit'] = vendor_sales_summary['TotalSalesDollars']_
        ↳- vendor_sales_summary['TotalPurchaseDollars']
```

```
[39]: vendor_sales_summary
```

```
[39]:
```

	VendorNumber	VendorName	Brand \
0	1128	BROWN-FORMAN CORP	1233
1	4425	MARTIGNETTI COMPANIES	3405
2	17035	PERNOD RICARD USA	8068
3	3960	DIAGEO NORTH AMERICA INC	4261
4	3960	DIAGEO NORTH AMERICA INC	3545
...
10687	9815	WINE GROUP INC	8527
10688	8004	SAZERAC CO INC	5683
10689	3924	HEAVEN HILL DISTILLERIES	9123
10690	3960	DIAGEO NORTH AMERICA INC	6127
10691	7245	PROXIMO SPIRITS INC.	3065

	Description	PurchasePrice	ActualPrice	Volume \
0	Jack Daniels No 7 Black	26.27	36.99	1750.0
1	Tito's Handmade Vodka	23.19	28.99	1750.0
2	Absolut 80 Proof	18.24	24.99	1750.0
3	Capt Morgan Spiced Rum	16.17	22.99	1750.0
4	Ketel One Vodka	21.89	29.99	1750.0
...
10687	Concannon Glen Ellen Wh Zin	1.32	4.99	750.0
10688	Dr McGillicuddy's Apple Pie	0.39	0.49	50.0
10689	Deep Eddy Vodka	0.74	0.99	50.0
10690	The Club Strawbry Margarita	1.47	1.99	200.0
10691	Three Olives Grape Vodka	0.71	0.99	50.0

	TotalPurchaseQuantity	TotalPurchaseDollars	TotalSalesQuantity \
0	145080	3811251.60	142049.0
1	164038	3804041.22	160247.0

2	187407	3418303.68	187140.0
3	201682	3261197.94	200412.0
4	138109	3023206.01	135838.0
...
10687	2	2.64	5.0
10688	6	2.34	134.0
10689	2	1.48	2.0
10690	1	1.47	72.0
10691	1	0.71	86.0

	TotalSalesDollars	TotalSalesPrice	TotalExciseTax	FreightCost	\
0	5101919.51	672819.31	260999.20	68601.68	
1	4819073.49	561512.37	294438.66	144929.24	
2	4538120.60	461140.15	343854.07	123780.22	
3	4475972.88	420050.01	368242.80	257032.07	
4	4223107.62	545778.28	249587.83	257032.07	
...	
10687	15.95	10.96	0.55	27100.41	
10688	65.66	1.47	7.04	50293.62	
10689	1.98	0.99	0.10	14069.87	
10690	143.28	77.61	15.12	257032.07	
10691	85.14	33.66	4.46	38994.78	

	GrossProfit
0	1290667.91
1	1015032.27
2	1119816.92
3	1214774.94
4	1199901.61
...	...
10687	13.31
10688	63.32
10689	0.50
10690	141.81
10691	84.43

[10692 rows x 15 columns]

```
[43]: vendor_sales_summary['ProfitMargin'] = (vendor_sales_summary['GrossProfit']/
      ↪ vendor_sales_summary['TotalSalesDollars'])*100
```

```
[45]: vendor_sales_summary['ProfitMargin']
```

```
[45]: 0      25.297693
      1      21.062810
      2      24.675786
      3      27.139908
```

```

4          28.412764
...
10687      83.448276
10688      96.436186
10689      25.252525
10690      98.974037
10691      99.166079
Name: ProfitMargin, Length: 10692, dtype: float64

```

```

[47]: vendor_sales_summary['StockTurnover'] =
      ↪ vendor_sales_summary['TotalSalesQuantity']/
      ↪ vendor_sales_summary['TotalPurchaseQuantity']

```

```

[49]: vendor_sales_summary['StockTurnover']

```

```

[49]: 0          0.979108
      1          0.976890
      2          0.998575
      3          0.993703
      4          0.983556
...
10687      2.500000
10688     22.333333
10689      1.000000
10690     72.000000
10691     86.000000
Name: StockTurnover, Length: 10692, dtype: float64

```

```

[51]: vendor_sales_summary['SalesToPurchaseRatio'] =
      ↪ vendor_sales_summary['TotalSalesDollars']/
      ↪ vendor_sales_summary['TotalPurchaseDollars']

```

```

[ ]:

```

```

[53]: cursor = conn.cursor()

```

```

[59]: cursor.execute("""
CREATE TABLE vendor_sales_summary (
    VendorNumber INT,
    VendorName VARCHAR(100),
    Brand INT,
    Description VARCHAR(100),
    PurchasePrice DECIMAL(10,2),
    ActualPrice DECIMAL(10,2),
    Volume DECIMAL(10,2),
    TotalPurchaseQuantity INT,
    TotalPurchaseDollars DECIMAL(15,2),

```

```

TotalSalesQuantity INT,
TotalSalesDollars DECIMAL(15,2),
TotalSalesPrice DECIMAL(15,2),
TotalExciseTax DECIMAL(15,2),
FreightCost DECIMAL(15,2),
GrossProfit DECIMAL(15,2),
ProfitMargin DECIMAL(15,2),
StockTurnover DECIMAL(15,2),
SalesToPurchaseRatio DECIMAL(15,2),
PRIMARY KEY (VendorNumber, Brand)
);
"""

```

[59]: <sqlite3.Cursor at 0x2dde04c6e40>

```
[61]: pd.read_sql_query("select * from vendor_sales_summary",conn)
```

[61]: Empty DataFrame
Columns: [VendorNumber, VendorName, Brand, Description, PurchasePrice, ActualPrice, Volume, TotalPurchaseQuantity, TotalPurchaseDollars, TotalSalesQuantity, TotalSalesDollars, TotalSalesPrice, TotalExciseTax, FreightCost, GrossProfit, ProfitMargin, StockTurnover, SalesToPurchaseRatio]
Index: []

```
[63]: vendor_sales_summary.to_sql('vendor_sales_summary',conn, if_exists = 'replace',
    ↪, index = False)
```

[63]: 10692

```
[67]: pd.read_sql_query("select * from vendor_sales_summary",conn)
```

```
[67]:
```

	VendorNumber	VendorName	Brand	\
0	1128	BROWN-FORMAN CORP	1233	
1	4425	MARTIGNETTI COMPANIES	3405	
2	17035	PERNOD RICARD USA	8068	
3	3960	DIAGEO NORTH AMERICA INC	4261	
4	3960	DIAGEO NORTH AMERICA INC	3545	
...	
10687	9815	WINE GROUP INC	8527	
10688	8004	SAZERAC CO INC	5683	
10689	3924	HEAVEN HILL DISTILLERIES	9123	
10690	3960	DIAGEO NORTH AMERICA INC	6127	
10691	7245	PROXIMO SPIRITS INC.	3065	

	Description	PurchasePrice	ActualPrice	Volume	\
0	Jack Daniels No 7 Black	26.27	36.99	1750.0	
1	Tito's Handmade Vodka	23.19	28.99	1750.0	

2	Absolut 80 Proof	18.24	24.99	1750.0
3	Capt Morgan Spiced Rum	16.17	22.99	1750.0
4	Ketel One Vodka	21.89	29.99	1750.0
...
10687	Concannon Glen Ellen Wh Zin	1.32	4.99	750.0
10688	Dr McGillicuddy's Apple Pie	0.39	0.49	50.0
10689	Deep Eddy Vodka	0.74	0.99	50.0
10690	The Club Strawbry Margarita	1.47	1.99	200.0
10691	Three Olives Grape Vodka	0.71	0.99	50.0

	TotalPurchaseQuantity	TotalPurchaseDollars	TotalSalesQuantity	\
0	145080	3811251.60	142049.0	
1	164038	3804041.22	160247.0	
2	187407	3418303.68	187140.0	
3	201682	3261197.94	200412.0	
4	138109	3023206.01	135838.0	
...	
10687	2	2.64	5.0	
10688	6	2.34	134.0	
10689	2	1.48	2.0	
10690	1	1.47	72.0	
10691	1	0.71	86.0	

	TotalSalesDollars	TotalSalesPrice	TotalExciseTax	FreightCost	\
0	5101919.51	672819.31	260999.20	68601.68	
1	4819073.49	561512.37	294438.66	144929.24	
2	4538120.60	461140.15	343854.07	123780.22	
3	4475972.88	420050.01	368242.80	257032.07	
4	4223107.62	545778.28	249587.83	257032.07	
...	
10687	15.95	10.96	0.55	27100.41	
10688	65.66	1.47	7.04	50293.62	
10689	1.98	0.99	0.10	14069.87	
10690	143.28	77.61	15.12	257032.07	
10691	85.14	33.66	4.46	38994.78	

	GrossProfit	ProfitMargin	StockTurnover	SalesToPurchaseRatio
0	1290667.91	25.297693	0.979108	1.338647
1	1015032.27	21.062810	0.976890	1.266830
2	1119816.92	24.675786	0.998575	1.327594
3	1214774.94	27.139908	0.993703	1.372493
4	1199901.61	28.412764	0.983556	1.396897
...
10687	13.31	83.448276	2.500000	6.041667
10688	63.32	96.436186	22.333333	28.059829
10689	0.50	25.252525	1.000000	1.337838
10690	141.81	98.974037	72.000000	97.469388

10691 84.43 99.166079 86.000000 119.915493

[10692 rows x 18 columns]

```
[1]: import sqlite3
import pandas as pd
import logging
logging.basicConfig(
    filename="logs/get_vendor_summary.log",
    level=logging.DEBUG,
    format="%(asctime)s - %(levelname)s - %(message)s",
    filemode="a"
)

def ingest_db(df, table_name, engine):
    '''this function will ingest the dataframe into database table'''
    df.to_sql(table_name, con = engine, if_exists = 'replace', index = False)

def create_vendor_summary(conn):
    '''this function will merge the different tables to get the overall vendor_
    ↪summary and adding new columns in the resultant data'''
    vendor_sales_summary = pd.read_sql_query("""WITH FreightSummary AS (
        SELECT
            VendorNumber,
            SUM(Freight) AS FreightCost
        FROM vendor_invoice
        GROUP BY VendorNumber
    ),

    PurchaseSummary AS (
        SELECT
            p.VendorNumber,
            p.VendorName,
            p.Brand,
            p.Description,
            p.PurchasePrice,
            pp.Price AS ActualPrice,
            pp.Volume,
            SUM(p.Quantity) AS TotalPurchaseQuantity,
            SUM(p.Dollars) AS TotalPurchaseDollars
        FROM purchases p
        JOIN purchase_prices pp
            ON p.Brand = pp.Brand
        WHERE p.PurchasePrice > 0
        GROUP BY p.VendorNumber, p.VendorName, p.Brand, p.Description, p.
    ↪PurchasePrice, pp.Price, pp.Volume
    ),
```



```

SalesSummary AS (
    SELECT
        VendorNo,
        Brand,
        SUM(SalesQuantity) AS TotalSalesQuantity,
        SUM(SalesDollars) AS TotalSalesDollars,
        SUM(SalesPrice) AS TotalSalesPrice,
        SUM(ExciseTax) AS TotalExciseTax
    FROM sales
    GROUP BY VendorNo, Brand
)

```

```

SELECT
    ps.VendorNumber,
    ps.VendorName,
    ps.Brand,
    ps.Description,
    ps.PurchasePrice,
    ps.ActualPrice,
    ps.Volume,
    ps.TotalPurchaseQuantity,
    ps.TotalPurchaseDollars,
    ss.TotalSalesQuantity,
    ss.TotalSalesDollars,
    ss.TotalSalesPrice,
    ss.TotalExciseTax,
    fs.FreightCost
FROM PurchaseSummary ps
LEFT JOIN SalesSummary ss
    ON ps.VendorNumber = ss.VendorNo
    AND ps.Brand = ss.Brand
LEFT JOIN FreightSummary fs
    ON ps.VendorNumber = fs.VendorNumber
ORDER BY ps.TotalPurchaseDollars DESC""",conn)

```

```

return vendor_sales_summary

```

```

def clean_data(df):
    '''this function will clean the data'''
    # changing datatype to float
    df['Volume'] = df['Volume'].astype('float')

    # filling missing value with 0
    df.fillna(0,inplace = True)

```

```

# removing spaces from categorical columns
df['VendorName'] = df['VendorName'].str.strip()
df['Description'] = df['Description'].str.strip()

# creating new columns for better analysis
df['GrossProfit'] = df['TotalSalesDollars'] - df['TotalPurchaseDollars']
df['ProfitMargin'] = (df['GrossProfit'] / df['TotalSalesDollars'])*100
df['StockTurnover'] = df['TotalSalesQuantity'] / df['TotalPurchaseQuantity']
df['SalesToPurchaseRatio'] = df['TotalSalesDollars'] /
↳df['TotalPurchaseDollars']

return df

if __name__ == '__main__':
    # creating database connection
    conn = sqlite3.connect('inventory.db')

    logging.info('Creating Vendor Summary Table.....')
    summary_df = create_vendor_summary(conn)
    logging.info(summary_df.head())

    logging.info('Cleaning Data.....')
    clean_df = clean_data(summary_df)
    logging.info(clean_df.head())

    logging.info('Ingesting data.....')
    ingest_db(clean_df, 'vendor_sales_summary', conn)
    logging.info('Completed')

```

```
[ ]:
```