



Soha Khalid

BWT – Data Engineering

Task 03 – Exercise

- Create a new database (Resource link: https://www.youtube.com/watch?v=IugEHi_5kMA)
 - Create a schema in it
 - Create tables for the provided csvs by understanding the data in the csvs. Use correct data type for each column.
-
1. Write a query to fetch all customer names and sort them alphabetically. (Topic Sorting)
 2. Write a query to fetch all product names and their prices, sorted by price from low to high. (Topic Sorting)
 3. Write a query to fetch supplier names that start with the letter 'A' and sort them by their names. (Topic Sorting with Operators and Wildcards)
 4. Write a query to fetch all items and sort them by their status, placing NULL values first.
 5. Write a query to fetch all products, sort them first by category and then by price in descending order.
 6. Write a query to fetch all customer names and phone numbers, but sort them by the last four digits of their phone numbers in ascending order. (Hint use sorting with sub strings)

Creating Tables and Importing Data:

```
CREATE TABLE sales.categories (  
    category_id VARCHAR(100) PRIMARY KEY,  
    name VARCHAR(100),  
    status VARCHAR(20),  
    description VARCHAR(500)  
);
```

```
CREATE TABLE sales.cities (  
    city_id VARCHAR(100) PRIMARY KEY,  
    name VARCHAR(100),  
    province VARCHAR(20),  
    country VARCHAR(100),  
    status VARCHAR(20)  
);
```

```
CREATE TABLE sales.customers (  
    customer_id VARCHAR(100) PRIMARY KEY,  
    name VARCHAR(100),  
    phone VARCHAR(50),  
    location VARCHAR(50),  
    status VARCHAR(20)  
);
```

```
CREATE TABLE sales.items (  
    item_id VARCHAR(100) PRIMARY KEY,  
    order_id VARCHAR(100),  
    product_id VARCHAR(100),
```

```
        amount float,  
        status VARCHAR(20),  
        item_timestamp timestamp  
    );
```

```
CREATE TABLE sales.orders (  
    order_id VARCHAR(100) PRIMARY KEY,  
    customer_id VARCHAR(100),  
    status VARCHAR(50),  
    order_time timestamp,  
    total_amount float  
);
```

```
CREATE TABLE sales.products (  
    product_id VARCHAR(100) PRIMARY KEY,  
    name VARCHAR(50),  
    supplier_id VARCHAR(100),  
    category VARCHAR(50),  
    price float,  
    stock_available int,  
    status VARCHAR(50),  
    product_create_timestamp timestamp  
);
```

```
CREATE TABLE sales.suppliers (  
    supplier_id VARCHAR(100) PRIMARY KEY,  
    name VARCHAR(100),  
    phone VARCHAR(100),
```

```
        location VARCHAR(50),  
        status VARCHAR(50),  
        category VARCHAR(50)  
    );
```

```
COPY sales.categories(category_id, name, status, description)  
FROM 'C:\Users\Muniba Shah\Downloads\csvs\categories.csv'  
DELIMITER ','  
CSV HEADER;
```

```
COPY sales.cities(city_id, name, province, country, status)  
FROM 'C:\Users\Muniba Shah\Downloads\csvs\cities.csv'  
DELIMITER ','  
CSV HEADER;
```

```
COPY sales.customers(customer_id, name, phone, location, status)  
FROM 'C:\Users\Muniba Shah\Downloads\csvs\customers.csv'  
DELIMITER ','  
CSV HEADER;
```

```
COPY sales.items(item_id, order_id, product_id, amount, status, item_timestamp)  
FROM 'C:\Users\Muniba Shah\Downloads\csvs\items.csv'  
DELIMITER ','  
CSV HEADER;
```

```
COPY sales.orders(order_id, customer_id, status, order_time, total_amount)  
FROM 'C:\Users\Muniba Shah\Downloads\csvs\orders.csv'  
DELIMITER ','  
CSV HEADER;
```

COPY sales.products(product_id, name, supplier_id, category, price, stock_available,
status, product_createtimestamp)

FROM 'C:\Users\Muniba Shah\Downloads\csvs\products.csv'

DELIMITER ','

CSV HEADER;

COPY sales.suppliers(supplier_id, name, phone, location, status, category)

FROM 'C:\Users\Muniba Shah\Downloads\csvs\suppliers.csv'

DELIMITER ','

CSV HEADER;

✓ Servers

✓ PostgreSQL 16

✓ Databases (2)

✓ SALES

> Casts

> Catalogs

> Event Triggers

> Extensions

> Foreign Data Wrappers

> Languages

> Publications

✓ Schemas (2)

> public

✓ sales

✓ Tables (7)

> categories

> cities

> customers

> items

> orders

> products

> suppliers

Query Query History

```

41
42 -- Question 1
43 ✓ SELECT name AS customer_name
44 FROM sales.customers
45 ORDER BY customer_name;
46
47
48

```

Data Output Messages Notifications

≡+

📄

▼

📋

▼


🗑️

🗄️

⬇️

📈

SQL

	customer_name 
1	Aaron Hayes
2	Aaron Jones
3	Aaron Schwartz
4	Abigail Reed
5	Adam David
6	Adam Johns
7	Adam Martinez
8	Alan Massey
9	Alex Smith
10	Alexander Chen

Total rows: 1000 of 1000 Query complete 00:00:00.261 Ln 57, Col 1

Query Query History

```
55
56
57  -- Question 2
58  ✓ SELECT name AS product_name, price
59  FROM sales.products
60  ORDER BY price;
61
62
```

Data Output Messages Notifications



	product_name character varying (50) 🔒	price double precision 🔒
1	deal	11.1
2	interview	12.37
3	step	12.72
4	especially	12.74
5	game	12.78
6	level	13.59
7	include	13.75
8	their	16.27
9	build	17.25
10	behavior	18.83

Total rows: 1000 of 1000

Query complete 00:00:00.220

Ln 61, Col 1

Query Query History

```
63
64
65 -- Question 3
66 v SELECT name AS supplier_name
67 FROM sales.suppliers
68 WHERE name LIKE 'A%'
69 ORDER BY name;
70
```

Data Output Messages Notifications



	supplier_name character varying (100)	🔒
1	Acosta-Freeman	
2	Adams-Roberts	
3	Adams-Schwartz	
4	Adams, Hull and Wise	
5	Adams, Rojas and Morgan	
6	Adkins and Sons	
7	Aguilar-Carter	
8	Aguilar Ltd	
9	Aguirre, Taylor and Gregory	
10	Alexander and Sons	

Total rows: 46 of 46 Query complete 00:00:00.198 Ln 68, Col 21

QueryQuery History

70

-- Question 4

72 SELECT *

73 FROM sales.items

74 ORDER BY status NULLS FIRST;

75

76

77

Data OutputMessagesNotifications

SQL

	item_id [PK] character varying (100)	order_id character varying (100)	product_id character varying (100)	amount double precision	status character
1	55d21c57-d8a7-4c8e-96f1-af1c3db13160	6f66a611-407a-4851-b699-4ce8d8d095...	d03892f1-f8f3-44fc-bb86-e431717e4b6f	14.31	Delivered
2	8a2c9420-43b5-4b51-91aa-1fcae29b7811	7ea54b60-64d2-4583-96df-bf4ef031da...	5a49b5e1-a14d-4a84-b131-4b2de86b63...	32.95	Delivered
3	72af47e3-9998-420a-86ed-56c01cb9bd81	ddee01dc-20a1-42ff-a286-7124267ad0...	7217ab96-f866-4c74-b9ae-9c3eaf767ab4	86.44	Delivered
4	4b5a19b1-64be-41d7-ae74-a2543ca734e4	b5e30740-1585-4ea6-9c99-cd6a7877f4...	f8366f6b-ecf1-4e75-935d-ab91137c2933	8.1	Delivered
5	0bdb0f27-6bf6-4e0c-9b49-9fee26a36b60	411c82eb-100e-48fc-8095-befe7791b6f6	e6d06da3-c252-4117-9a45-6c6511aa71...	56.5	Delivered
6	e7df11b8-2943-4e28-9168-12e136a1ada0	1170d4f6-9a0f-4de6-b0e5-f671870a49...	9ff2250b-0845-41a1-9ee4-b2bbf5cc50bb	77.22	Delivered
7	03f57e54-db61-4fbd-b59a-8d5d45baa180	160a6358-7348-4e75-89dc-a37a8caf60...	3bc16381-bd29-4b7f-8366-3ecf707883f8	51.07	Delivered
8	72fd19f-5c38-473f-a712-ad354c8a89bc	ce967796-9486-406e-9955-6c53442a8...	dd9a1016-b858-4245-a173-dc99ea0f63...	55.03	Delivered
9	5932d11f-c4f2-4905-9ac5-bbf4123b10c7	b12b51a7-2e8e-4aab-b272-9ba4fdc80d...	4de9f1be-dfeb-4100-9463-566d527f4784	75.21	Delivered
10	58c47008-24b5-47f5-8c64-b668a170a2d7	58015f1-2d81-4007-8577-57370c45f4...	c7b55008-11d0-4001-8c14-b648714000	58.71	Delivered

Total rows: 1000 of 1000 Query complete 00:00:00.245 Ln 74, Col 10

QueryQuery History

77

-- Question 5

79 SELECT *

80 FROM sales.products

81 ORDER BY category, PRICE DESC

82

83

84

Data OutputMessagesNotifications

SQL

	product_id [PK] character varying (100)	name character varying (50)	supplier_id character varying (100)	category character varying (50)	price double precision	st in
1	12ea3377-cca5-4965-a41c-9a30998310...	of	3f89d404-d78f-4aca-8c67-ee4e434674d5	a	302.68	
2	494c69e7-db73-42fd-9eef-7880822585a5	offer	01abe709-0567-47dc-ae92-3b077c44b8...	able	385.18	
3	b8adc046-eef0-4f58-9c7a-7c6f58a2d628	share	6252ec97-c226-4c94-8b70-7ce1642719...	able	76.71	
4	a63e349b-fb6e-4222-aca0-5079603843bf	wide	16157d6a-f40a-4140-b233-c8e5e8542b...	about	398.27	
5	e61f1dad-1ec7-4114-aaf5-7a449f91c7b4	factor	c9a89c46-4f89-4cc4-a804-be28738fc04b	about	91.72	
6	2d61572a-ad68-4eb9-a327-cbde171ce7...	middle	f292cfe7-c77d-4dd4-8b75-b5eb72f08ece	above	337.62	
7	e37e126c-f0d4-4b67-8efc-5d4226a23a18	door	7b539636-fcdb-4749-9e31-f9e38c83cec8	above	281.2	
8	dad2a61c-41d1-49ea-96b1-6c78daa687...	Congress	c2721d21-27ef-474c-82a6-5dcf1e3b6fcd	above	93.24	
9	ef209ab5-5443-4496-8669-8021d28b88...	yeah	2ae9ab92-e46f-4c86-b155-cc020004c8...	accept	88.18	
10	8005d8c-1078-4156-b104-1673e20a51...	relies	475402-7618-41c8-8005-801170f071b...	conradia	167.76	

Total rows: 1000 of 1000 Query complete 00:00:00.201 Ln 82, Col 1

Query

Query History

82

83

84

85

86

87

88

89

-- Question 6

SELECT name AS customer_names, phone

FROM sales.customers

ORDER BY RIGHT(phone,4)

Data Output

Messages

Notifications

≡+

📄

▼

📋

▼

🗑️

🗄️

⬇️

📈

SQL

	customer_names character varying (100) 🔒	phone character varying (50) 🔒
1	Elizabeth Lin	001-965-883-8361x0012
2	Ashley Cobb	001-587-449-0043
3	Erin Ward	+1-640-549-0044
4	Erin Miller	4327710045
5	Samuel Maddox	495-356-2786x0046
6	Sierra Johnson	(767)912-4013x0051
7	Andrew Jones	(209)855-8805x0073
8	Robert Gonzalez	598.888.5382x10083
9	Jordan Miller	298.307.0089
10	Abigail Reed	203-341-0115

Total rows: 1000 of 1000

Query complete 00:00:00.139

Ln 89, Col 24