**Day2**

1)      Alter Table:

* Add a new column linkedin\_profile to employees table to store LinkedIn URLs as varchar.

Alter table employees

Add column linkedin\_profile varchar;

A screenshot of a computer

AI-generated content may be incorrect.

* Change the linkedin\_profile column data type from VARCHAR to TEXT.

Alter table employees

Alter column linkedin\_profile

set data type text;

A screenshot of a message

AI-generated content may be incorrect.

* Add unique, not null constraint to linkedin\_profile

ALTER TABLE employees

ALTER COLUMN linkedin\_profile SET NOT NULL,

ADD constraint unique\_linkedin\_profile Unique(linkedin\_profile);

A screenshot of a computer

AI-generated content may be incorrect.

* Drop column linkedin\_profile

Alter table employees

Drop column linkedin\_profile;

A screenshot of a computer

AI-generated content may be incorrect.

2)      Querying (Select)

* Retrieve the employee name and title of all employees

Select employeename, title from employees;

A screenshot of a computer

AI-generated content may be incorrect.

* Find all unique unit prices of products

SELECT DISTINCT unitprice

FROM products;

A screenshot of a cell phone

AI-generated content may be incorrect.

* List all customers sorted by company name in ascending order

select \* from customers

Order by companyname ASC;

A screenshot of a computer

AI-generated content may be incorrect.

* Display product name and unit price, but rename the unit\_price column as price\_in\_usd

select productname, unitprice AS price\_in\_usd from products;

A screenshot of a computer

AI-generated content may be incorrect.

3)      Filtering

* Get all customers from Germany.

select \* from customers where country='Germany';

A screenshot of a computer

AI-generated content may be incorrect.

* Find all customers from France or Spain

select \* from customers where country='France' OR country='Spain';

A screenshot of a computer

AI-generated content may be incorrect.

* Retrieve all orders placed in 2014(based on order\_date), and either have freight greater than 50 or the shipped date available (i.e., non-NULL)  (Hint: EXTRACT(YEAR FROM order\_date))

select \* from orders

where EXTRACT(year from orderdate)=2014

AND (freight>50 OR shippeddate is not null);

A screenshot of a computer

AI-generated content may be incorrect.

4)      Filtering

* Retrieve the product\_id, product\_name, and unit\_price of products where the unit\_price is greater than 15.

Select productid, productname, unitprice from products

where unitprice>15;

A screenshot of a computer

AI-generated content may be incorrect.

* List all employees who are located in the USA and have the title "Sales Representative".

select \* from employees

Where country= 'USA' AND title='Sales Representative';

A screenshot of a phone

AI-generated content may be incorrect.

* Retrieve all products that are not discontinued and priced greater than 30.

select \* from products

where discontinued=false AND unitprice>30;

A screenshot of a menu

AI-generated content may be incorrect.

5)      LIMIT/FETCH

* Retrieve the first 10 orders from the orders table.

select \* from orders

limit 10;

A screenshot of a computer

AI-generated content may be incorrect.

* Retrieve orders starting from the 11th order, fetching 10 rows (i.e., fetch rows 11-20).

select \* from orders

**Offset** 10 rows

**Fetch** next 10 Rows only;

A screenshot of a computer

AI-generated content may be incorrect.

6)      Filtering (IN, BETWEEN)

* List all customers who are either Sales Representative, Owner

**Select \* from customers**

**where contacttitle IN ('Sales Representative', 'Owner');**

A screenshot of a computer

AI-generated content may be incorrect.

* Retrieve orders placed between January 1, 2013, and December 31, 2013.

select \* from orders

Where orderdate Between 2013-01-01 AND 2013-12-31;



7)      Filtering

* List all products whose category\_id is not 1, 2, or 3.

**select \* from products**

**where categoryid Not IN (1, 2, 3);**

**A screenshot of a computer

AI-generated content may be incorrect.**

* Find customers whose company name starts with "A".

**Select \* from customers**

**Where companyname Like 'A%';**

A screenshot of a computer

AI-generated content may be incorrect.

8)       INSERT into orders table:

 Task: Add a new order to the orders table with the following details:

Order ID: 11078

Customer ID: ALFKI

Employee ID: 5

Order Date: 2025-04-23

Required Date: 2025-04-30

Shipped Date: 2025-04-25

shipperID:2

Freight: 45.50

INSERT into orders(orderid, customerid, employeeid, orderdate, requireddate, shippeddate, shipperid, freight)

values (11078,'ALFKI', 5, '2025-04-23', '2025-04-30', '2025-04-25', 2, 45.50);

A screenshot of a computer

AI-generated content may be incorrect.

9)      Increase(Update)  the unit price of all products in category\_id =2 by 10%.

(HINT: unit\_price =unit\_price \* 1.10)

UPDATE products

SET unitprice = unitprice \* 1.10

WHERE categoryid = 2;

A screenshot of a computer

AI-generated content may be incorrect.

10) Sample Northwind database:

Download

1. Download northwind.sql from below link into your local. Sign in to Git first <https://github.com/pthom/northwind_psql>
2. Manually Create the database using pgAdmin:
   1. Right-click on "Databases" → Create → Database
   2. Give name as ‘northwind’ (all small letters)
   3. Click ‘Save’

Import database:

1. Open pgAdmin and connect to your server
2. Select the database  ‘northwind’
3. Right Click-> Query tool.
4. Click the folder icon to open your northwind.sql file
5. Press F5 or click the Execute button.
6. You will see total 14 tables loaded
7. Databases → your database → Schemas → public → Tables

A screenshot of a computer

AI-generated content may be incorrect.