**Day1**

1. Create table for categories.csv – Enforce Primary Key and foreign Key constraints
2. Create table for customers.csv – Enforce Primary Key and foreign Key constraints
3. Create table for employees.csv – Enforce Primary Key and foreign Key constraints
4. Create table for order\_details.csv – Enforce Primary Key and foreign Key constraints
5. Create table for orders.csv – Enforce Primary Key and foreign Key constraints
6. Create table for shippers.csv – Enforce Primary Key and foreign Key constraints
7. Create table for products.csv – Enforce Primary Key and foreign Key constraints
   1. Apply below Constraints while creating products Table:

|  |  |
| --- | --- |
| **Column Name** | **Constraint to Apply** |
| productID | Primary Key (auto-generated) |
| productName | Cannot be NULL (NOT NULL), UNIQUE |
| quantityPerUnit | Cannot be NULL (NOT NULL) |
| unitPrice | NOT NULL ,Cannot be negative (CHECK (price >= 0)) |
| discontinued | NOT NULL , Should default to 0 (DEFAULT 0) |
| categoryID | Foreign Key, Cannot be NULL (NOT NULL) |

After creating, tables, Import data from csv to all the tables.

**Day2**

1. Alter Table:

* Add a new column linkedin\_profile to employees table to store LinkedIn URLs as varchar.
* Change the linkedin\_profile column data type from VARCHAR to TEXT.
* Add unique, not null constraint to linkedin\_profile
* Drop column linkedin\_profile

1. Querying (Select)

* Retrieve the employee name, and title of all employees
* Find all **unique unit prices** of products
* List all customers sorted by **company name in ascending order**
* Display product name and unit price, but rename the unit\_price column as price\_in\_usd

1. Filterimg
   * Get all customers from **Germany**.
   * Find all customers from **France** or **Spain**
   * 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))
2. Filtering
   * Retrieve the product\_id, product\_name, and unit\_price of products where the unit\_price is greater than **15**.
   * List all employees who are located in **USA** and have the title **"Sales Representative"**.
   * Retrieve all products that are **not discontinued** and **priced greater than 30**.
3. LIMIT/FETCH

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

1. Filtering (IN, BETWEEN)

* List all customers who are either **Sales Representative, Owner**
* Retrieve orders placed **between January 1, 2013, and December 31, 2013.**

1. Filtering

* List all products whose **category\_id** is not **1, 2, or 3**.
* Find customers whose company name starts with **"A"**.

1. 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

1. **I**ncrease(Update) the unit price of all products in **category\_id =2** by **10%**.

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

1. 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>

1. Manually Create the database using pgAdmin:

a. Right-click on "Databases" → **Create** → **Database**

b. Give name as ‘**northwind**’ (all small letters)

c. 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 Execute button.
    6. You will see total 14 tables loaded
    7. Databases → your database → Schemas → public → Tables