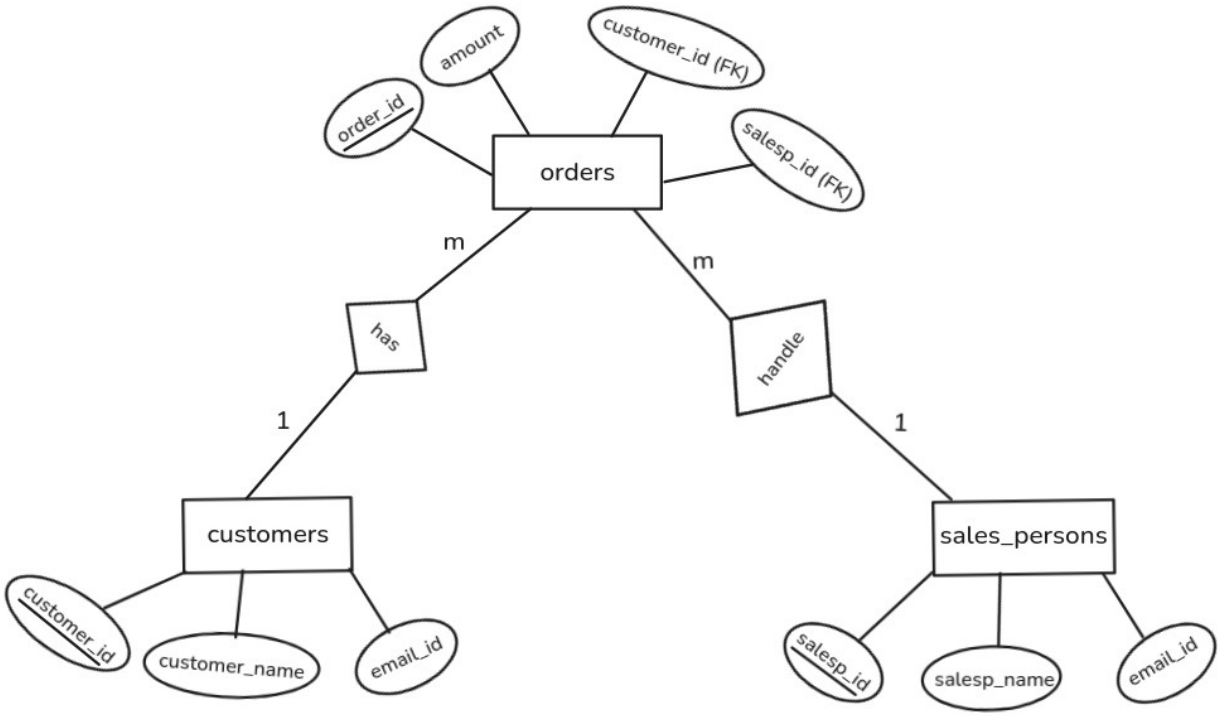
Databases Exercise

Problem Statement: There can be multiple customers, who can place multiple orders on the site. Now a sales person can handle these orders will distribute into multiple sales persons (One order will be assign to one salesperson only). So a sales person can have multiple orders of multiple customers

Q1. Create Database

```
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 15
Server version: 8.0.44-0ubuntu0.24.04.2 (Ubuntu)

Copyright (c) 2000, 2025, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> SHOW DATABASES;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
4 rows in set (0.01 sec)

mysql> CREATE DATABASE sales_db;
Query OK, 1 row affected (0.01 sec)

mysql> SHOW DATABASES;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| sales_db           |
| sys                |
+--------------------+
5 rows in set (0.00 sec)

mysql> USE sales_db
Database changed
mysql>
```

Q2. Design Schema



Database Schema:

1. customers
   - customer_id (Primary Key)
   - customer_name
   - email_id

2. sales_persons
   - salesp_id (Primary Key)
   - salesp_name
   - email_id

3. orders
   - order_id (Primary Key)
   - amount
   - customer_id (Foreign Key)
   - salesp_id (Foreign Key)

Relationships:

- A customer can place multiple orders, but each order is placed by only one customer.
- A salesperson can handle multiple orders, but each order is handled by only one salesperson.
- An order acts as a link between a customer and a salesperson.
- There is no direct relationship between customers and salespersons; they are connected only through orders.

Q3. Create tables

```
mysql> CREATE TABLE customers (
    -> customer_id INT PRIMARY KEY AUTO_INCREMENT,
    -> customer_name VARCHAR(100) NOT NULL,
    -> email_id VARCHAR(100) UNIQUE
    -> );
Query OK, 0 rows affected (0.04 sec)

mysql> CREATE TABLE sales_persons (
    -> salesp_id INT PRIMARY KEY AUTO_INCREMENT,
    -> salesp_name VARCHAR(100) NOT NULL,
    -> email_id VARCHAR(100) UNIQUE
    -> );
Query OK, 0 rows affected (0.06 sec)
```

```
mysql> CREATE TABLE orders (
    -> order_id INT PRIMARY KEY AUTO_INCREMENT,
    -> amount DECIMAL(10,2) NOT NULL,
    -> customer_id INT NOT NULL,
    -> salesp_id INT NOT NULL,
    -> FOREIGN KEY (customer_id) REFERENCES customers(customer_id),
    -> FOREIGN KEY (salesp_id) REFERENCES sales_persons(salesp_id)
    -> );
Query OK, 0 rows affected (0.05 sec)

mysql>
```

Q4. Insert sample data

```
mysql> INSERT INTO customers (customer_name, email_id)
    -> VALUES
    -> ('Amit Sharma', 'amit@gmail.com'),
    -> ('Neha Verma', 'neha@gmail.com'),
    -> ('Rahul Singh', 'rahul@gmail.com'),
    -> ('Pooja Gupta', 'pooja@gmail.com');
Query OK, 4 rows affected (0.02 sec)
Records: 4  Duplicates: 0  Warnings: 0

mysql> INSERT INTO sales_persons (salesp_name, email_id)
    -> VALUES
    -> ('Rohit Kumar', 'rohit@gmail.com'),
    -> ('Priya Mehta', 'priya@gmail.com'),
    -> ('Ankit Verma', 'ankit@gmail.com');
Query OK, 3 rows affected (0.01 sec)
Records: 3  Duplicates: 0  Warnings: 0

mysql> INSERT INTO orders (amount, customer_id, salesp_id)
    -> VALUES
    -> (5000.00, 1, 1),
    -> (12000.00, 1, 2),
    -> (3000.00, 2, 1),
    -> (7500.75, 3, 3),
    -> (9800.00, 4, 2),
    -> (4100.25, 4, 1);
Query OK, 6 rows affected (0.01 sec)
Records: 6  Duplicates: 0  Warnings: 0

mysql>
```

Q5. Find the sales person have multiple orders.

```
mysql> SELECT * FROM sales_persons s
    -> INNER JOIN
    -> orders o
    -> ON s.salesp_id = o.salesp_id;
+-----------+-------------+-----------------+----------+----------+-------------+-----------+
| salesp_id | salesp_name | email_id        | order_id | amount   | customer_id | salesp_id |
+-----------+-------------+-----------------+----------+----------+-------------+-----------+
|         1 | Rohit Kumar | rohit@gmail.com |        1 |  5000.00 |           1 |         1 |
|         1 | Rohit Kumar | rohit@gmail.com |        3 |  3000.00 |           2 |         1 |
|         1 | Rohit Kumar | rohit@gmail.com |        6 |  4100.25 |           4 |         1 |
|         2 | Priya Mehta | priya@gmail.com |        2 | 12000.00 |           1 |         2 |
|         2 | Priya Mehta | priya@gmail.com |        5 |  9800.00 |           4 |         2 |
|         3 | Ankit Verma | ankit@gmail.com |        4 |  7500.75 |           3 |         3 |
+-----------+-------------+-----------------+----------+----------+-------------+-----------+
6 rows in set (0.00 sec)

mysql> SELECT s.salesp_name, COUNT(o.order_id) AS 'no_of_orders' FROM sales_persons s
    -> INNER JOIN
    -> orders o
    -> ON s.salesp_id = o.salesp_id
    -> GROUP BY s.salesp_id HAVING no_of_orders > 1;
+-------------+--------------+
| salesp_name | no_of_orders |
+-------------+--------------+
| Rohit Kumar |            3 |
| Priya Mehta |            2 |
+-------------+--------------+
2 rows in set (0.00 sec)

mysql>
```

Q6. Find the all sales person details along with order details

```
mysql> SELECT s.salesp_id, s.salesp_name, s.email_id, o.order_id, o.amount, o.customer_id FROM sales_persons s
    -> LEFT JOIN
    -> orders o
    -> ON s.salesp_id = o.salesp_id;
+-----------+-------------+-----------------+----------+----------+-------------+
| salesp_id | salesp_name | email_id        | order_id | amount   | customer_id |
+-----------+-------------+-----------------+----------+----------+-------------+
|         1 | Rohit Kumar | rohit@gmail.com |        1 |  5000.00 |           1 |
|         1 | Rohit Kumar | rohit@gmail.com |        3 |  3000.00 |           2 |
|         1 | Rohit Kumar | rohit@gmail.com |        6 |  4100.25 |           4 |
|         2 | Priya Mehta | priya@gmail.com |        2 | 12000.00 |           1 |
|         2 | Priya Mehta | priya@gmail.com |        5 |  9800.00 |           4 |
|         3 | Ankit Verma | ankit@gmail.com |        4 |  7500.75 |           3 |
+-----------+-------------+-----------------+----------+----------+-------------+
6 rows in set (0.00 sec)

mysql>
```

Q7. Create index

```
mysql> CREATE INDEX idx_orders_salesp_id ON orders(salesp_id);
Query OK, 0 rows affected (0.06 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> CREATE INDEX idx_orders_customer_id ON orders(customer_id);
Query OK, 0 rows affected (0.07 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql>
```

Q8. How to show index on a table

```
mysql> SHOW INDEX FROM orders;
+--------+------------+-------------+--------------+-------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+------------+
| Table  | Non_unique | Key_name    | Seq_in_index | Column_name | Collation | Cardinality | Sub_part | Packed | Null | Index_type | Comment | Index_comment | Visible | Expression |
+--------+------------+-------------+--------------+-------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+------------+
| orders |          0 | PRIMARY     |            1 | order_id    | A         |           6 |     NULL |   NULL |      | BTREE      |         |               | YES     | NULL       |
| orders |          1 | customer_id |            1 | customer_id | A         |           4 |     NULL |   NULL |      | BTREE      |         |               | YES     | NULL       |
| orders |          1 | salesp_id   |            1 | salesp_id   | A         |           3 |     NULL |   NULL |      | BTREE      |         |               | YES     | NULL       |
+--------+------------+-------------+--------------+-------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+------------+
3 rows in set (0.02 sec)
```

Q9. Find the order number, sale person name, along with the customer to whom that order belongs to

```
mysql> SELECT o.order_id, s.salesp_name, c.customer_name FROM orders o
    -> INNER JOIN
    -> sales_persons s
    -> ON o.salesp_id = s.salesp_id
    -> INNER JOIN
    -> customers c
    -> ON c.customer_id = o.customer_id
    -> ORDER BY o.order_id;
+----------+-------------+---------------+
| order_id | salesp_name | customer_name |
+----------+-------------+---------------+
|        1 | Rohit Kumar | Amit Sharma   |
|        2 | Priya Mehta | Amit Sharma   |
|        3 | Rohit Kumar | Neha Verma    |
|        4 | Ankit Verma | Rahul Singh   |
|        5 | Priya Mehta | Pooja Gupta   |
|        6 | Rohit Kumar | Pooja Gupta   |
+----------+-------------+---------------+
6 rows in set (0.00 sec)

mysql>
```