DBMSL

ROLL NO : 31147.

Assignment 4

Problem Statement:

Design at least 10 SQL queries for suitable database application using SQL DML statements. All types of join, Sub-query and view.

Objectives:

1) Understand the concept of Foreign key.
2) Join two or more tables to extract the output.
3) Understand DML statements.

Outcomes:

1) Should be able to relate two tables.
2) Should be able to understand the significance of relating two tables.

S/W & H/W Requirements:

1) Operating system (Windows/Ubuntu).
2) MySQL Database.

Theory:

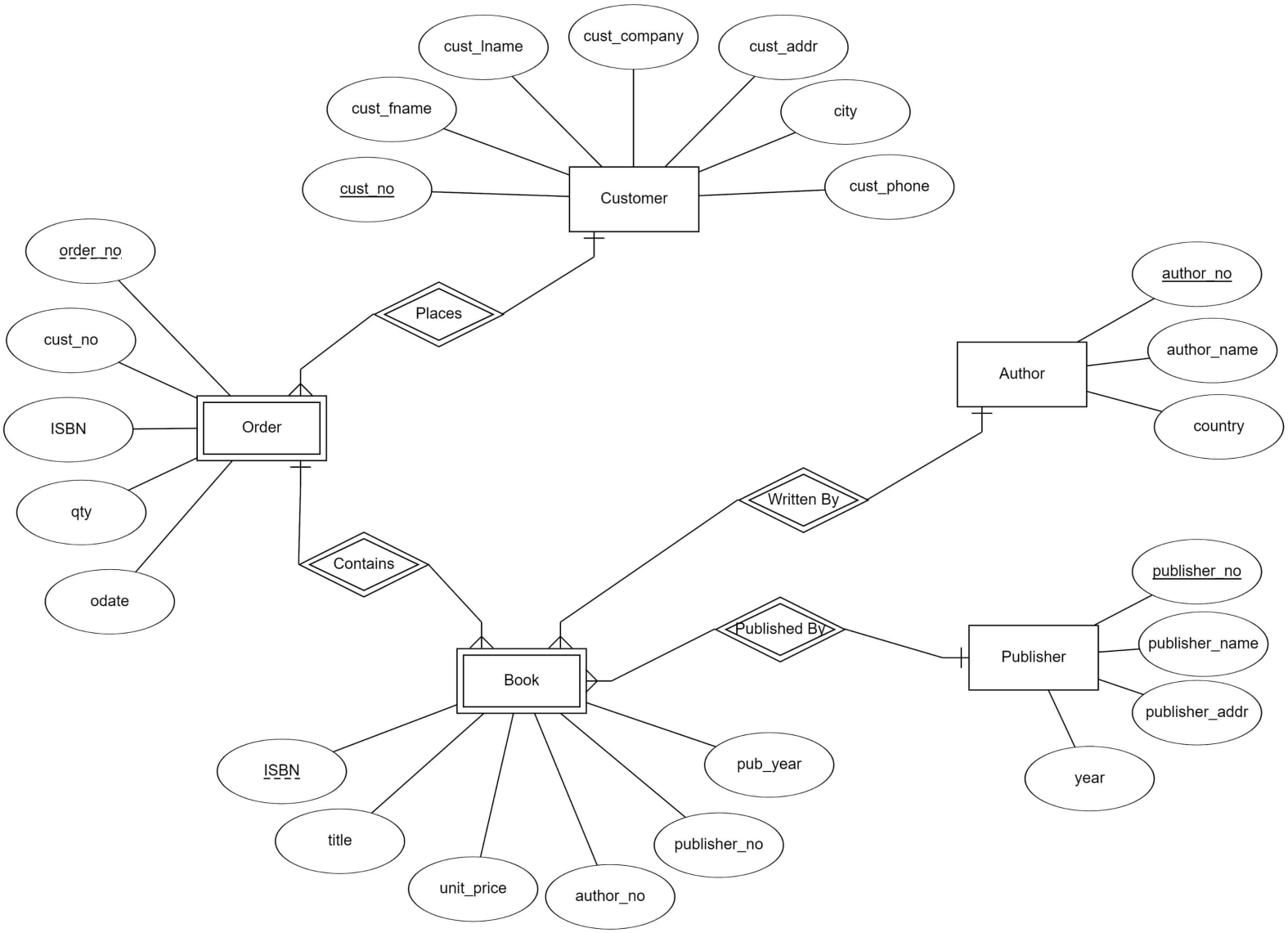
Join in MySQL?

Join is a way of combining two tables according to a row which has a foreign key from one table to another.

Different types of Join:

1) INNER JOIN
2) LEFT OUTER JOIN
3) RIGHT OUTER JOIN
4) FULL OUTER JOIN.
5) NATURAL JOIN

Natural Join : It joins two tables creating an implicit join clause for columns in two table which are common.



1) INNER JOIN: It selects records from two tables having matching values (columns).

a) SELECT * FROM Customer INNER JOIN orders
ON customer.cust_no = orders.cust_no;

b) SELECT * FROM Customer INNER JOIN orders
USING (cust_no);

2) LEFT JOIN: The left join returns all rows from left table and the matched record from the right table. If there is no match NULL value is returned.

a) SELECT * FROM customer LEFT JOIN Orders
ON customer.cust_no = orders.cust_no;

3) RIGHT JOIN: The right join returns all rows from right table and the matched row from left table and if there is no match NULL is returned.

a) SELECT * FROM customer RIGHT JOIN orders ON
customer.cust_no = orders.cust_no;

4) FULL OUTER JOIN: It performs left outer join as well as right join.

SELECT * From customer LEFT JOIN orders Using(cust_no)
UNION ALL
SELECT * FROM customer RIGHT JOIN orders using(cust_no);

Views:

Views are data objects which include the contents of the table Views are copies of base tables. Views like tables have columns Views can be used to insert, delete, update from tables. They are for security reasons for hiding the real table name.

Test Cases:

1) Find the ~~professor~~ customer details and ~~department~~ orders details USING NATURAL JOIN

SELECT * FROM customer NATURAL JOIN orders;

Cust_no	Cust_fname	Cust_lname	Cust_company	Cust_addr	city	Cust_phone	order_no	ISBN	qty
1	Alfred	Futter	Facebook	Obere Str. 57	Berlin	9988877340	1	—	2
2	John	DOE	ICICI BANK	Trimurti chowk	Pune	9988872341	2	—	2
⋮									

2) Display total books and customer name.

```
SELECT * cust_fname, SUM(qty)
FROM customer INNER JOIN orders
WHERE customer.cust_no = orders.cust_no
GROUP BY ordr.cust_no;
```

Cust_fname	Sum(qty)
Alfred	3
John	2
Anonio	1
Harshit	9
Harshad	5

3) Create View:

```
CREATE VIEW cust_view AS
SELECT * FROM customers;
```

4) INSERT VIEW:

```
INSERT INTO cust_view VALUES(10, 'Akbar', 'Ali', 'GS', 'addr', 'Pune'
, 9960716253);
```

5) Drop View:

```
DROP VIEW cust_view;
```

Conclusion: Learnt about interrelation between two tables and joining of two table and performing operations on view through command line.