

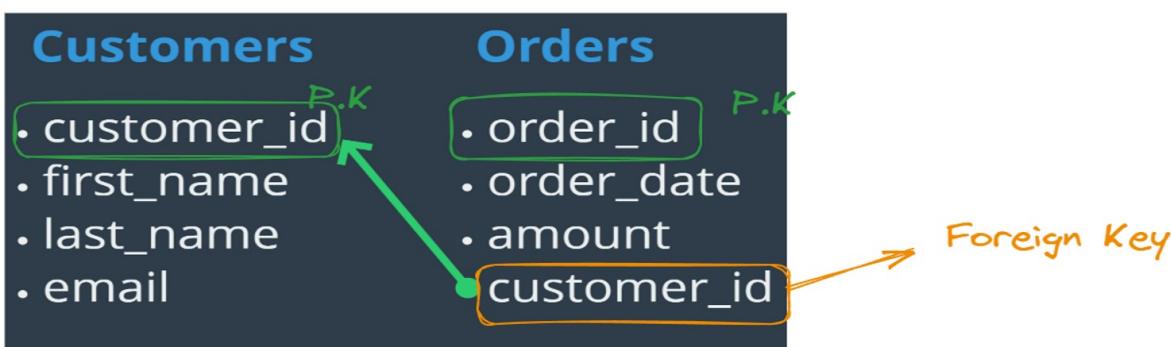
# Relationships & Joins - SQL Session 9

09 July 2024 11:56

## Relationships & Joins - SQL Session 9

1. One to One Relationship
2. One to Many Relationship
3. Many to Many Relationship

### 1:Many Relationship -



```
mysql> CREATE TABLE Customers(
    ->   customer_id INT PRIMARY KEY auto_increment,
    ->     first_name varchar(100),
    ->     last_name varchar(100),
    ->     email varchar(100)
    -> );
```

```
mysql> DESC Customers;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| customer_id | int | NO | PRI | NULL | auto_increment |
| first_name | varchar(100) | YES | | NULL | |
| last_name | varchar(100) | YES | | NULL | |
| email | varchar(100) | YES | | NULL | |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.01 sec)
```

```
mysql> CREATE TABLE Orders(
    ->   order_id INT primary key auto_increment,
    ->   order_date DATE,
    ->   amount DECIMAL(7,2),
    ->   customer_id INT,
    ->   FOREIGN KEY (customer_id) references Customers(customer_id)
    -> );
Query OK, 0 rows affected (0.06 sec)
```

```
mysql> DESC Orders;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| order_id | int | NO | PRI | NULL | auto_increment |
| order_date | date | YES | | NULL | |
+-----+-----+-----+-----+-----+-----+
```

```
+-----+-----+-----+-----+-----+-----+
| Field | Type   | Null | Key  | Default | Extra       |
+-----+-----+-----+-----+-----+-----+
| order_id | int    | NO   | PRI  | NULL    | auto_increment |
| order_date | date   | YES  |      | NULL    |               |
| amount | decimal(7,2) | YES  |      | NULL    |               |
| customer_id | int    | YES  | MUL  | NULL    |               |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.01 sec)
```

## CUSTOMERS

customer_id	first_name	last_name	email
1	Boy	George	george@gmail.com
2	George	Michael	gm@gmail.com
3	David	Bowie	david@gmail.com
4	Blue	Steele	blue@gmail.com

## ORDERS

order_id	order_date	amount	customer_id
1	'2016/02/10'	99.99	1
2	'2017/11/11'	35.50	1
3	'2014/12/12'	800.67	2
4	'2015/01/03'	12.50	2

```
mysql> INSERT INTO Customers(first_name , last_name , email)
-> VALUES('Boy' , 'George' , 'george@gmail.com'),
-> ('George' , 'Michael' , 'gm@gmail.com'),
-> ('David' , 'Bowie' , 'david@gmail.com'),
-> ('Blue' , 'Steele' , 'blue@gmail.com');
Query OK, 4 rows affected (0.01 sec)
Records: 4  Duplicates: 0  Warnings: 0
```

```
mysql> SELECT * FROM Customers;
+-----+-----+-----+-----+
| customer_id | first_name | last_name | email        |
+-----+-----+-----+-----+
| 1 | Boy      | George    | george@gmail.com |
| 2 | George   | Michael   | gm@gmail.com    |
| 3 | David    | Bowie     | david@gmail.com |
| 4 | Blue     | Steele   | blue@gmail.com  |
+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

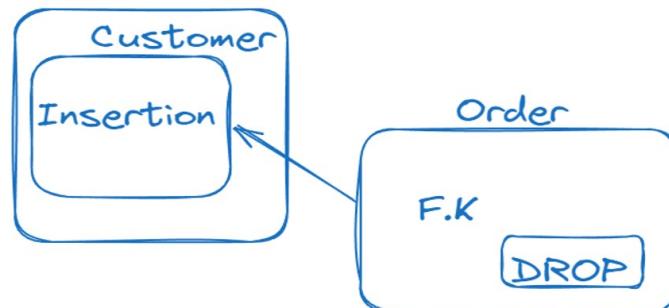
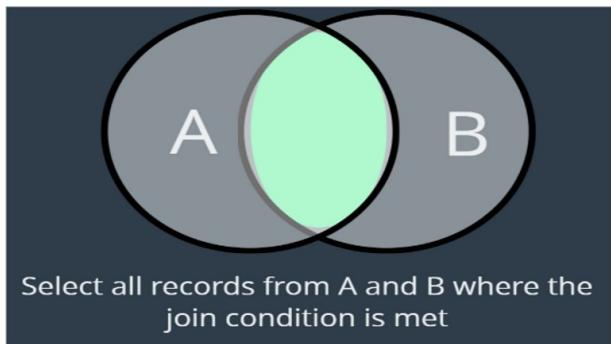
```
mysql> INSERT INTO Orders(order_date , amount , customer_id)
-> VALUES('2016/02/10' , 99.99 , 1),
-> ('2017/11/11' , 35.50 , 1),
-> ('2014/12/12' , 800.67 , 2),
-> ('2015/01/03' , 12.50 , 2);
Query OK, 4 rows affected, 4 warnings (0.01 sec)
Records: 4  Duplicates: 0  Warnings: 4
```

```
mysql> SELECT * FROM Orders;
+-----+-----+-----+-----+
| order_id | order_date | amount | customer_id |
+-----+-----+-----+-----+
| 1 | 2016-02-10 | 99.99 | 1 |
| 2 | 2017-11-11 | 35.50 | 1 |
| 3 | 2014-12-12 | 800.67 | 2 |
| 4 | 2015-01-03 | 12.50 | 2 |
+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

```
4 rows in set (0.00 sec)
```

## JOINS :

### - INNER JOIN



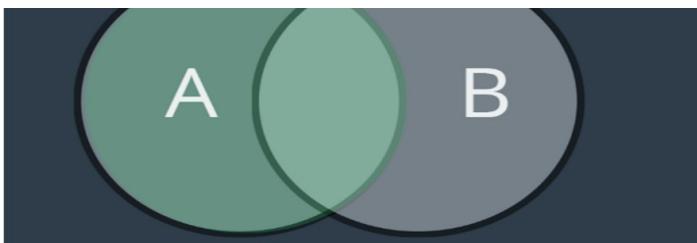
```
mysql> SELECT * FROM Customers
    -> JOIN Orders ON
    -> Customers.customer_id = Orders.customer_id;
+-----+-----+-----+-----+-----+-----+-----+
| customer_id | first_name | last_name | email           | order_id | order_date | amount | customer_id |
+-----+-----+-----+-----+-----+-----+-----+
| 1 | Boy      | George    | george@gmail.com | 1 | 2016-02-10 | 99.99 | 1 |
| 1 | Boy      | George    | george@gmail.com | 2 | 2017-11-11 | 35.50 | 1 |
| 2 | George   | Michael   | gm@gmail.com     | 3 | 2014-12-12 | 800.67 | 2 |
| 2 | George   | Michael   | gm@gmail.com     | 4 | 2015-01-03 | 12.50 | 2 |
+-----+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)

mysql> SELECT first_name , order_date , amount FROM Customers
    -> JOIN Orders ON
    -> Customers.customer_id = Orders.customer_id;
+-----+-----+-----+
| first_name | order_date | amount |
+-----+-----+-----+
| Boy        | 2016-02-10 | 99.99 |
| Boy        | 2017-11-11 | 35.50 |
| George     | 2014-12-12 | 800.67 |
| George     | 2015-01-03 | 12.50 |
+-----+-----+-----+
4 rows in set (0.00 sec)
```

```
mysql> SELECT first_name , last_name , SUM(amount) as total FROM Customers
    -> JOIN Orders ON
    -> Customers.customer_id = Orders.customer_id
    -> GROUP BY first_name , last_name
    -> ORDER BY total DESC;
+-----+-----+-----+
| first_name | last_name | total |
+-----+-----+-----+
| George     | Michael   | 813.17 |
| Boy        | George    | 135.49 |
+-----+-----+-----+
2 rows in set (0.00 sec)
```

## LEFT JOIN





Select everything from A, along with any matching records in B

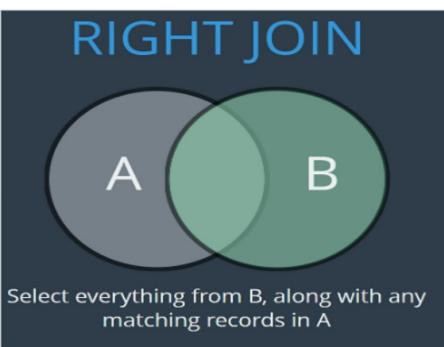
```
mysql> SELECT * FROM Customers
-> LEFT JOIN Orders ON
-> Customers.customer_id = Orders.customer_id;
+-----+-----+-----+-----+-----+-----+-----+-----+
| customer_id | first_name | last_name | email | order_id | order_date | amount | customer_id |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | Boy | George | george@gmail.com | 1 | 2016-02-10 | 99.99 | 1 |
| 1 | Boy | George | george@gmail.com | 2 | 2017-11-11 | 35.50 | 1 |
| 2 | George | Michael | gm@gmail.com | 3 | 2014-12-12 | 800.67 | 2 |
| 2 | George | Michael | gm@gmail.com | 4 | 2015-01-03 | 12.50 | 2 |
| 3 | David | Bowie | david@gmail.com | NULL | NULL | NULL | NULL |
| 4 | Blue | Steele | blue@gmail.com | NULL | NULL | NULL | NULL |
+-----+-----+-----+-----+-----+-----+-----+-----+
6 rows in set (0.00 sec)
```

Trick to know which join to apply.

if it is right side null,  
simply apply Left Join  
& vice-versa.

```
mysql> SELECT
->   first_name ,
->   last_name,
->   IFNULL(SUM(amount),0) as total
->   FROM Customers
->   LEFT JOIN Orders ON
->   Customers.customer_id = Orders.customer_id
->   GROUP BY first_name , last_name
->   ORDER BY total DESC;
+-----+-----+-----+
| first_name | last_name | total |
+-----+-----+-----+
| George | Michael | 813.17 |
| Boy | George | 135.49 |
| David | Bowie | 0.00 |
| Blue | Steele | 0.00 |
+-----+-----+-----+
4 rows in set (0.00 sec)
```

Handling Null Values  
to calculate.



Select everything from B, along with any matching records in A

```
mysql> SELECT * FROM Customers
```

matching records in A

```
mysql> SELECT * FROM Customers
    -> RIGHT JOIN Orders ON
    -> Customers.customer_id = Orders.customer_id;
+-----+-----+-----+-----+-----+-----+-----+-----+
| customer_id | first_name | last_name | email      | order_id | order_date | amount | customer_id |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | Boy       | George     | george@gmail.com | 1 | 2016-02-10 | 99.99 | 1 |
| 1 | Boy       | George     | george@gmail.com | 2 | 2017-11-11 | 35.50 | 1 |
| 2 | George    | Michael    | gm@gmail.com    | 3 | 2014-12-12 | 800.67 | 2 |
| 2 | George    | Michael    | gm@gmail.com    | 4 | 2015-01-03 | 12.50 | 2 |
+-----+-----+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

Quick Challenge :

## WRITE THIS SCHEMA

### STUDENTS

- id
- first\_name

### PAPERS

- title
- grade
- student\_id

```
INSERT INTO students (first_name) VALUES
('Caleb'), ('Samantha'), ('Raj'), ('Carlos'), ('Lisa');
```

```
INSERT INTO papers (student_id, title, grade ) VALUES
(1, 'My First Book Report', 60),
(1, 'My Second Book Report', 75),
(2, 'Russian Lit Through The Ages', 94),
(2, 'De Montaigne and The Art of The Essay', 98),
(4, 'Borges and Magical Realism', 89);
```

```
mysql> CREATE TABLE Students(
    -> id INT PRIMARY KEY auto_increment,
    -> first_name varchar(100)
    -> );
Query OK, 0 rows affected (0.07 sec)
```

```
mysql> DESC students;
+-----+-----+-----+-----+-----+-----+
| Field      | Type       | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| id         | int        | NO   | PRI | NULL    | auto_increment |
| first_name | varchar(100) | YES  |     | NULL    |             |
+-----+-----+-----+-----+-----+-----+
2 rows in set (0.02 sec)
```

```
mysql> CREATE TABLE papers(
    -> title varchar(100),
    -> grade int,
    -> student_id int,
    -> foreign key (student_id) references Students(id)
    -> );
Query OK, 0 rows affected (0.03 sec)
```

```
mysql> DESC papers;
+-----+-----+-----+-----+-----+-----+
| Field      | Type       | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| title      | varchar(100) | YES  |     | NULL    |             |
| grade      | int        | YES  |     | NULL    |             |
+-----+-----+-----+-----+-----+-----+
```

```

+-----+-----+-----+-----+-----+-----+
| Field | Type  | Null | Key  | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| title | varchar(100) | YES |       | NULL    |       |
| grade | int        | YES |       | NULL    |       |
| student_id | int        | YES | MUL   | NULL    |       |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.01 sec)

```

## PRINT THIS

```

+-----+-----+-----+
| first_name | title           | grade |
+-----+-----+-----+
| Samantha   | De Montaigne and The Art of The Essay | 98  |
| Samantha   | Russian Lit Through The Ages          | 94  |
| Carlos     | Borges and Magical Realism            | 89  |
| Caleb      | My Second Book Report                 | 75  |
| Caleb      | My First Book Report                  | 60  |
+-----+-----+-----+

```

```

mysql> SELECT first_name , title , grade
-> FROM Students JOIN papers
-> ON Students.id = papers.student_id
-> ORDER BY grade DESC;
+-----+-----+-----+
| first_name | title           | grade |
+-----+-----+-----+
| Samantha   | De Montaigne and The Art of The Essay | 98  |
| Samantha   | Russian Lit Through The Ages          | 94  |
| Carlos     | Borges and Magical Realism            | 89  |
| Caleb      | My Second Book Report                 | 75  |
| Caleb      | My First Book Report                  | 60  |
+-----+-----+-----+
5 rows in set (0.00 sec)

```

## PRINT THIS

first_name	title	grade
Caleb	My First Book Report	60
Caleb	My Second Book Report	75
Samantha	Russian Lit Through The Ages	94
Samantha	De Montaigne and The Art of The Essay	98
Raj	NULL	NULL
Carlos	Borges and Magical Realism	89
Lisa	NULL	NULL

```

mysql> SELECT first_name , title , grade
-> FROM Students LEFT JOIN papers
-> ON Students.id = papers.student_id;
+-----+-----+-----+
| first_name | title           | grade |
+-----+-----+-----+
| Caleb      | My First Book Report | 60  |
| Caleb      | My Second Book Report | 75  |
| Samantha   | Russian Lit Through The Ages | 94  |
| Samantha   | De Montaigne and The Art of The Essay | 98  |
| Raj        | NULL             | NULL |
| Carlos     | Borges and Magical Realism            | 89  |
| Lisa       | NULL             | NULL |
+-----+-----+-----+
7 rows in set (0.00 sec)

```

## PRINT THIS

first_name	title	grade
Caleb	My First Book Report	60

first_name	title	grade
Caleb	My First Book Report	60
Caleb	My Second Book Report	75
Samantha	Russian Lit Through The Ages	94
Samantha	De Montaigne and The Art of The Essay	98
Raj	MISSING	0
Carlos	Borges and Magical Realism	89
Lisa	MISSING	0

```
mysql> SELECT
->   first_name,
->   IFNULL(title , "MISSING") AS title,
->   IFNULL(grade , 0) AS grade
->   FROM Students LEFT JOIN papers
->   ON Students.id = papers.student_id;
+-----+-----+
| first_name | title          | grade |
+-----+-----+
| Caleb      | My First Book Report | 60   |
| Caleb      | My Second Book Report | 75   |
| Samantha   | Russian Lit Through The Ages | 94   |
| Samantha   | De Montaigne and The Art of The Essay | 98   |
| Raj        | MISSING           | 0    |
| Carlos     | Borges and Magical Realism | 89   |
| Lisa       | MISSING           | 0    |
+-----+-----+
7 rows in set (0.00 sec)
```

## PRINT THIS

first_name	average
Samantha	96.0000
Carlos	89.0000
Caleb	67.5000
Raj	0
Lisa	0

```
mysql> SELECT
->   first_name ,
->   IFNULL(AVG(grade),0) as average
->   FROM Students LEFT JOIN Papers
->   ON Students.id = papers.student_id
->   GROUP BY first_name
->   ORDER BY average DESC;
+-----+-----+
| first_name | average |
+-----+-----+
| Samantha  | 96.0000 |
| Carlos    | 89.0000 |
| Caleb     | 67.5000 |
| Raj       | 0.0000  |
| Lisa      | 0.0000  |
+-----+-----+
5 rows in set (0.00 sec)
```

## PRINT THIS

first_name	average	passing_status
Samantha	96.0000	

first_name	average	passing_status
Samantha	96.0000	PASSING
Carlos	89.0000	PASSING
Caleb	67.5000	FAILING
Raj	0	FAILING
Lisa	0	FAILING

```
mysql> SELECT
    ->   first_name ,
    ->   IFNULL(AVG(grade),0) as average,
    ->   CASE
    ->     WHEN IFNULL(AVG(grade),0) >=75 THEN 'Passing'
    ->     ELSE 'Failing'
    ->   END as passing_status
    -> FROM Students LEFT JOIN Papers
    -> ON Students.id = papers.student_id
    -> GROUP BY first_name
    -> ORDER BY average DESC;
+-----+-----+-----+
| first_name | average | passing_status |
+-----+-----+-----+
| Samantha  | 96.0000 | Passing
| Carlos    | 89.0000 | Passing
| Caleb     | 67.5000 | Failing
| Raj       | 0.0000  | Failing
| Lisa      | 0.0000  | Failing
+-----+-----+-----+
5 rows in set (0.00 sec)
```