

Assignment 2 : Retrieve data using join with where clause

Sample table1: saleman

-salesman_id

-name

-city

-commission

The screenshot shows the phpMyAdmin interface with the following details:

- Server: 127.0.0.1
- Database: sale
- Table: saleman

The table structure is as follows:

	salesman_id	name	city	commission
1	Raj	Surat	12	
2	Meet	Vadodara	14	
3	Dhruv	Bardoli	7	
4	Sahil	Valsad	18	
5	Mahesh	Rajkot	14	
6	Ram	Anand	9	
7	Sagar	Kutch	24	
8	Bhavesh	Ahmedabad	34	
9	Mayur	Surat	13	
10	Heet	Ankleshwar	27	
11	Naman	Mehsana	32	
12	Sohil	Bhavnagar	3	
13	Jenil	Palanpur	45	
14	Virag	Jamnagar	19	
15	Jeel	Amreli	11	
16	Mukesh	Surat	21	
17	Kishor	Vadodara	8	
18	Dharmesh	Surendranagar	16	
19	Ramesh	Bhavnagar	26	

Sample table2: customer

-customer_id

-cust_name

-city

-grade

-salesman_id

localhost:7882/phpmyadmin/index.php?route=/sql&db=sale&table=customer&pos=0

phpMyAdmin

Recent Favorites

- New
- abc
- demoapp
- information_schema
- javatpoint
- library_management
- login_register_youtube
- mysql
- performance_schema
- phpmyadmin
- sale
 - New
 - customer
 - orders
 - saleman
 - salesman
- student
- test
- user
- wordpress

Table: customer

	customer_id	cust_name	city	grade	salesman_id			
<input type="checkbox"/>	Edit	Copy	Delete	10	Ramesh	Surat	A	9
<input type="checkbox"/>	Edit	Copy	Delete	11	Mahesh	Valsad	A	19
<input type="checkbox"/>	Edit	Copy	Delete	12	Mahesh	Bardoli	B	12
<input type="checkbox"/>	Edit	Copy	Delete	13	Nand	Rajkot	D	15
<input type="checkbox"/>	Edit	Copy	Delete	14	Vibhav	Bhavnagar	B	3
<input type="checkbox"/>	Edit	Copy	Delete	15	Virah	Gandhinagar	B	4
<input type="checkbox"/>	Edit	Copy	Delete	16	Hamid	Valsad	C	20
<input type="checkbox"/>	Edit	Copy	Delete	17	Amrut	Mehsana	D	5
<input type="checkbox"/>	Edit	Copy	Delete	18	Manav	Junagadh	C	10
<input type="checkbox"/>	Edit	Copy	Delete	19	Krish	Palanpur	D	13
<input type="checkbox"/>	Edit	Copy	Delete	20	Rahul	Ahmedabad	A	7
<input type="checkbox"/>	Edit	Copy	Delete	21	Neel	Surat	D	2
<input type="checkbox"/>	Edit	Copy	Delete	22	Jayesh	Gandhinagar	B	18
<input type="checkbox"/>	Edit	Copy	Delete	23	Heet	Anand	D	14
<input type="checkbox"/>	Edit	Copy	Delete	24	Kishor	Junagadh	B	16
<input type="checkbox"/>	Edit	Copy	Delete	25	Raj	Sachin	C	6
<input type="checkbox"/>	Edit	Copy	Delete	26	Vibhav	Bhavnagar	C	11
<input type="checkbox"/>	Edit	Copy	Delete	27	Jatin	Bhuj	D	10
<input type="checkbox"/>	Console	Copy	Delete	28	Bipin	Palitana	B	1

Sample table3: orders

- ord_no
- purch_amt
- ord_date
- customer_id
- salesman_id

localhost:7882/phpmyadmin/index.php?route=/sql&server=1&db=sale&table=orders&pos=0

phpMyAdmin

Recent Favorites

- New
- abc
- demoapp
- information_schema
- javatpoint
- library_management
- login_register_youtube
- mysql
- performance_schema
- phpmyadmin
- sale
 - New
 - customer
 - orders
 - saleman
 - salesman
- student
- test
- user
- wordpress

Table: orders

	ord_no	purch_amt	ord_date	custmor_id	salesman_id			
<input type="checkbox"/>	Edit	Copy	Delete	51	2000	2023-01-01	22	8
<input type="checkbox"/>	Edit	Copy	Delete	52	1300	2023-01-06	12	18
<input type="checkbox"/>	Edit	Copy	Delete	53	1000	2022-12-29	16	3
<input type="checkbox"/>	Edit	Copy	Delete	54	3000	2022-11-09	23	15
<input type="checkbox"/>	Edit	Copy	Delete	55	4000	2022-12-08	27	17
<input type="checkbox"/>	Edit	Copy	Delete	56	1200	2023-01-18	19	12
<input type="checkbox"/>	Edit	Copy	Delete	57	2300	2023-01-24	17	1
<input type="checkbox"/>	Edit	Copy	Delete	58	3200	2023-01-08	14	9
<input type="checkbox"/>	Edit	Copy	Delete	59	4500	2023-01-10	21	16
<input type="checkbox"/>	Edit	Copy	Delete	60	1800	2023-02-02	13	2
<input type="checkbox"/>	Edit	Copy	Delete	61	2100	2023-01-19	26	14
<input type="checkbox"/>	Edit	Copy	Delete	62	3300	2022-10-12	10	20
<input type="checkbox"/>	Edit	Copy	Delete	63	6500	2022-11-26	25	6
<input type="checkbox"/>	Edit	Copy	Delete	64	900	2022-11-30	20	13
<input type="checkbox"/>	Edit	Copy	Delete	65	2400	2022-09-14	28	7
<input type="checkbox"/>	Edit	Copy	Delete	66	4500	2023-02-09	15	4
<input type="checkbox"/>	Edit	Copy	Delete	67	2400	2023-01-10	18	11
<input type="checkbox"/>	Edit	Copy	Delete	68	2800	2023-01-17	24	5
<input type="checkbox"/>	Console	Copy	Delete	69	2300	2022-09-21	11	19

1. write a SQL query to find the salesperson and customer who reside in the same city.
Return Salesman, cust_name and city

```
SELECT saleman.name, customer.cust_name, customer.city FROM saleman, customer  
WHERE saleman.city=customer.city;
```

name	cust_name	city
Raj	Ramesh	Surat
Mayur	Ramesh	Surat
Mukesh	Ramesh	Surat
Sahil	Mahesh	Valsad
Dhruv	Mahesh	Bardoli
Mahesh	Nand	Rajkot
Sohil	Vibhav	Bhavnagar
Ramesh	Vibhav	Bhavnagar
Mansukh	Virah	Gandhinagar
Sahil	Hamid	Valsad
Naman	Amrut	Mehsana
Jenil	Krish	Palanpur
Bhavesh	Rahul	Ahmedabad
Raj	Neel	Surat
Mayur	Neel	Surat
Mukesh	Neel	Surat
Mansukh	Jayesh	Gandhinagar
Ram	Heet	Anand
Sohil	Vibhav	Bhavnagar
Ramesh	Vibhav	Bhavnagar
Sohil	Mitesh	Bhavnagar
Ramesh	Mitesh	Bhavnagar

2. write a SQL query to find those orders where the order amount exists between 500 and 2000. Return ord_no, purch_amt, cust_name, city

```
SELECT o.ord_no, o.purch_amt, c.cust_name, c.city FROM orders o, customer c  
WHERE o.customer_id=c.customer_id AND o.purch_amt BETWEEN 500 AND 2000;
```

ord_no	purch_amt	cust_name	city
51	2000	Jayesh	Gandhinagar
52	1300	Mahesh	Bardoli
53	1000	Hamid	Valsad
56	1200	Krish	Palanpur
60	1800	Nand	Rajkot
64	900	Rahul	Ahmedabad
70	500	Mitesh	Bhavnagar

3. write a SQL query to find the salesperson(s) and the customer(s) he represents. Return Customer Name, city, Salesman, commission

```
SELECT c.cust_name AS "Customer Name",
c.city, s.name AS "Salesman", s.commission FROM customer c INNER JOIN
salesman s ON c.salesman_id=s.salesman_id;
```

The screenshot shows the phpMyAdmin interface with the database 'sale' selected. The left sidebar shows the schema structure with 'sale' expanded, revealing tables like 'customer', 'orders', 'salesman', and 'salesman'. The main area displays the 'salesman' table data:

Customer Name	city	Salesman	commission
Ramesh	Surat	Mayur	13
Maresh	Valsad	Ramesh	26
Maresh	Bardoli	Sohil	3
Nand	Rajkot	Jeel	11
Vibhav	Bhavnagar	Dhruv	7
Virah	Gandhinagar	Sahil	18
Hamid	Valsad	Mansukh	25
Amrut	Mehsana	Mahesh	14
Manav	Junagadh	Heet	27
Krish	Palanpur	Jenil	45
Rahul	Ahmedabad	Sagar	24
Neel	Surat	Meet	14
Jayesh	Gandhinagar	Dharmesh	16
Heet	Anand	Virag	19
Kishor	Junagadh	Mukesh	21
Raj	Sachin	Ram	9
Vibhav	Bhavnagar	Naman	32
Jatin	Bhuj	Heet	27
Bipin	Palitana	Raj	12
Mitesh	Bhavnagar	Bhavesh	34

4. write a SQL query to find salespeople who received commissions of more than 12 percent from the company. Return Customer Name, customer city, Salesman, commission.

```
select c.cust_name as 'Customer Name', c.city as 'Customer city',
s.name as 'Salesman', s.commission FROM customer c, salesman s WHERE
s.commission > 12;
```

The screenshot shows the phpMyAdmin interface with the database 'sale' selected. The left sidebar shows the schema structure with 'sale' expanded, revealing tables like 'customer', 'orders', 'salesman', and 'salesman'. The main area displays the 'salesman' table data, filtered by commission > 12:

Customer Name	customer city	Salesman	commission
Ramesh	Surat	Meet	14
Ramesh	Surat	Sahil	18
Ramesh	Surat	Mahesh	14
Ramesh	Surat	Sagar	24
Ramesh	Surat	Bhavesh	34
Ramesh	Surat	Mayur	13
Ramesh	Surat	Heet	27
Ramesh	Surat	Naman	32
Ramesh	Surat	Jenil	45
Ramesh	Surat	Virag	19
Ramesh	Surat	Mukesh	21
Ramesh	Surat	Dharmesh	16
Ramesh	Surat	Ramesh	26
Ramesh	Surat	Mansukh	25
Maresh	Valsad	Meet	14
Maresh	Valsad	Sahil	18
Maresh	Valsad	Maresh	14
Maresh	Valsad	Sagar	24
Maresh	Valsad	Bhavesh	34
Maresh	Valsad	Mayur	13
Maresh	Valsad	Heet	27
Maresh	Valsad	Naman	32
Maresh	Valsad	Jenil	45

5. write a SQL query to locate those salespeople who do not live in the same city where their customers live and have received a commission of more than 12% from the company. Return Customer Name, customer city, Salesman, salesman city, commission

```
SELECT c.cust_name AS "CustomerName", c.city, s.name AS "Salesman", s.city,
s.commission FROM customer c INNER JOIN saleman s ON
c.salesman_id=s.salesman_id WHERE s.commission>.12 AND c.city<>s.city;
```

Customer Name	city	Salesman	city	commission
Mahesh	Valsad	Ramesh	Bhavnagar	26
Mahesh	Bardoli	Sohil	Bhavnagar	3
Nand	Rajkot	Jeel	Amreli	11
Vibhav	Bhavnagar	Dhruv	Bardoli	7
Virah	Gandhinagar	Sahil	Valsad	18
Hamid	Valsad	Mansukh	Gandhinagar	25
Amrut	Mehsana	Mahesh	Rajkot	14
Manav	Junagadh	Heet	Ankleshwar	27
Rahul	Ahmedabad	Sagar	Kutch	24
Neel	Surat	Meet	Vadodara	14
Jayesh	Gandhinagar	Dharmesh	Surendranagar	16
Heet	Anand	Virag	Jamnagar	19
Kishor	Junagadh	Mukesh	Surat	21
Raj	Sachin	Ram	Anand	9
Vibhav	Bhavnagar	Naman	Mehsana	32
Jatin	Bhuj	Heet	Ankleshwar	27
Bipin	Palitana	Raj	Surat	12
Mitesh	Bhavnagar	Bhavesh	Ahmedabad	34

6. write a SQL query to find the details of an order. Return ord_no, ord_date, purch_amt, Customer Name, grade, Salesman, commission

```
SELECT o.ord_no, o.ord_date, o.purch_amt, c.cust_name AS "Customer
Name", c.grade, s.name AS "Salesman", s.commission FROM orders o INNER JOIN
customer c ON o.customer_id=c.customer_id INNER JOIN saleman s ON o.salesm
an_id=s.salesman_id;
```

ord_no	ord_date	purch_amt	Customer Name	grade	Salesman	commission
51	2023-01-01	2000	Jayesh	B	Bhavesh	34
52	2023-01-06	1300	Mahesh	B	Dharmesh	16
53	2022-12-29	1000	Hamid	C	Dhruv	7
54	2022-11-09	3000	Heet	D	Jeel	11
55	2022-12-08	4000	Jatin	D	Kishor	8
56	2023-01-18	1200	Krish	D	Sohil	3
57	2023-01-24	2300	Amrut	D	Raj	12
58	2023-01-08	3200	Vibhav	B	Mayur	13
59	2023-01-10	4500	Neel	D	Mukesh	21
60	2023-02-02	1800	Nand	D	Meet	14
61	2023-01-19	2100	Vibhav	C	Virag	19
62	2022-10-12	3300	Ramesh	A	Mansukh	25
63	2022-11-26	6500	Raj	C	Ram	9
64	2022-11-30	900	Rahul	A	Jenil	45
65	2022-09-14	2400	Bipin	B	Sagar	24
66	2023-02-09	4500	Virah	B	Sahil	18
67	2023-01-10	2400	Manav	C	Naman	32
68	2023-01-17	2800	Kishor	B	Mahesh	14
69	2022-09-21	2300	Mahesh	A	Ramesh	26
70	2023-01-25	500	Mitesh	B	Heet	27

7. Write a SQL statement to join the tables salesman, customer and orders so that the same column of each table appears once and only the relational rows are returned.

```
SELECT c.customer_id, s.salesman_id, o.ord_no, o.purch_amt, o.ord_date,
c.cust_name, c.city "C_City", c.grade, s.city as "S_City", s.commission FROM
orders o INNER JOIN customer c ON c.customer_id = o.customer_id INNER JOIN
salesman s ON s.salesman_id = c.salesman_id;
```

customer_id	salesman_id	ord_no	purch_amt	ord_date	cust_name	C_City	grade	S_City	commission
22	18	51	2000	2023-01-01	Jayesh	Gandhinagar	B	Surendranagar	16
12	12	52	1300	2023-01-06	Mahesh	Bardoli	B	Bhavnagar	3
16	20	53	1000	2022-12-29	Hamid	Valsad	C	Gandhinagar	25
23	14	54	3000	2022-11-09	Heet	Anand	D	Jamnagar	19
27	10	55	4000	2022-12-08	Jatin	Bhuj	D	Ankleshwar	27
19	13	56	1200	2023-01-18	Krish	Palanpur	D	Palanpur	45
17	5	57	2300	2023-01-24	Amrut	Mehsana	D	Rajkot	14
14	3	58	3200	2023-01-08	Vibhav	Bhavnagar	B	Bardoli	7
21	2	59	4500	2023-01-10	Neel	Surat	D	Vadodara	14
13	15	60	1800	2023-02-02	Nand	Rajkot	D	Amreli	11
26	11	61	2100	2023-01-19	Vibhav	Bhavnagar	C	Mehsana	32
10	9	62	3300	2022-10-12	Ramesh	Surat	A	Surat	13
25	6	63	6500	2022-11-26	Raj	Sachin	C	Anand	9
20	7	64	900	2022-11-30	Rahul	Ahmedabad	A	Kutch	24
28	1	65	2400	2022-09-14	Binpin	Palitana	B	Surat	12
15	4	66	4500	2023-02-09	Virah	Gandhinagar	B	Valsad	18
18	10	67	2400	2023-01-10	Manav	Junagadh	C	Ankleshwar	27
24	16	68	2800	2023-01-17	Kishor	Junagadh	B	Surat	21
11	19	69	2300	2022-09-21	Mahesh	Valsad	A	Bhavnagar	26
29	8	70	500	2023-01-25	Mitesh	Bhavnagar	B	Ahmedabad	34

8. write a SQL query to display the customer name, customer city, grade, salesman, salesman city. The results should be sorted by ascending customer_id.

```
SELECT c.cust_name, c.city, c.grade, s.name, s.city FROM customer c, salesman s
ORDER BY customer_id;
```

Customer Name	customer city	Salesman	commission
Ramesh	Surat	Meet	14
Ramesh	Surat	Sahil	18
Ramesh	Surat	Maheesh	14
Ramesh	Surat	Sagar	24
Ramesh	Surat	Bhavesh	34
Ramesh	Surat	Mayur	13
Ramesh	Surat	Heet	27
Ramesh	Surat	Naman	32
Ramesh	Surat	Jenil	45
Ramesh	Surat	Virag	19
Ramesh	Surat	Mukesh	21
Ramesh	Surat	Dharmesh	16
Ramesh	Surat	Ramesh	26
Ramesh	Surat	Mansukh	25
Maheesh	Valsad	Meet	14
Maheesh	Valsad	Sahil	18
Maheesh	Valsad	Maheesh	14
Maheesh	Valsad	Sagar	24
Maheesh	Valsad	Bhavesh	34
Maheesh	Valsad	Mayur	13
Maheesh	Valsad	Heet	27
Maheesh	Valsad	Naman	32
Maheesh	Valsad	Jenil	45

9. write a SQL query to find those customers with a grade less than 300. Return cust_name, customer city, grade, Salesman, salesmancity. The result should be ordered by ascending customer_id.

```
SELECT c.cust_name, c.city, c.grade, s.name AS "Salesman", s.city FROM
customer c INNER JOIN saleman s ON c.salesman_id=s.salesman_id ORDER
BY c.customer_id;
```

cust_name	city	grade	Salesman	city
Ramesh	Surat	A	Mayur	Surat
Mahesh	Valsad	A	Ramesh	Bhavnagar
Mahesh	Bardoli	B	Sohil	Bhavnagar
Nand	Rajkot	D	Jeel	Amreli
Vibhav	Bhavnagar	B	Dhruv	Bardoli
Virah	Gandhinagar	B	Sahil	Valsad
Hamid	Valsad	C	Mansukh	Gandhinagar
Amrut	Mehsana	D	Mahesh	Rajkot
Manav	Junagadh	C	Heet	Ankleshwar
Krish	Palanpur	D	Jenil	Palanpur
Rahul	Ahmedabad	A	Sagar	Kutch
Neel	Surat	D	Meet	Vaddoda
Jayesh	Gandhinagar	B	Dharmesh	Surendranagar
Heet	Anand	D	Virag	Jamnagar
Kishor	Junagadh	B	Mukesh	Surat
Raj	Sachin	C	Ram	Anand
Vibhav	Bhavnagar	C	Naman	Mehsana
Jatin	Bhuj	D	Heet	Ankleshwar
Bipin	Palitana	B	Raj	Surat
Mitesh	Bhavnagar	B	Bhavesh	Ahmedabad

10. Write a SQL statement to make a report with customer name, city, order number, order date, and order amount in ascending order according to the order date to determine whether any of the existing customers have placed an order or not

```
SELECT c.cust_name, c.city, o.ord_no, o.ord_date, o.purch_amt
AS "Order Amount" FROM customer c INNER JOIN orders o ON
c.customer_id=o.customer_id order by o.ord_date;
```

cust_name	city	ord_no	ord_date	Order Amount
Bipin	Palitana	65	2022-09-14	2400
Mahesh	Valsad	69	2022-09-21	2300
Ramesh	Surat	62	2022-10-12	3300
Heet	Anand	54	2022-11-09	3000
Raj	Sachin	63	2022-11-26	6500
Rahul	Ahmedabad	64	2022-11-30	900
Jatin	Bhuj	55	2022-12-08	4000
Hamid	Valsad	53	2022-12-29	1000
Jayesh	Gandhinagar	51	2023-01-01	2000
Mahesh	Bardoli	52	2023-01-06	1300
Vibhav	Bhavnagar	58	2023-01-08	3200
Manav	Junagadh	67	2023-01-10	2400
Neel	Surat	59	2023-01-10	4500
Kishor	Junagadh	68	2023-01-17	2800
Krish	Palanpur	56	2023-01-18	1200
Vibhav	Bhavnagar	61	2023-01-19	2100
Amrut	Mehsana	57	2023-01-24	2300
Mitesh	Bhavnagar	70	2023-01-25	500
Nand	Rajkot	60	2023-02-02	1800
Virah	Gandhinagar	66	2023-02-09	4500

11. Write a SQL statement to generate a report with customer name, city, order number, order date, order amount, salesperson name, and commission to determine if any of the existing customers have not placed orders or if they have placed orders through their salesman or by themselves

```
SELECT c.cust_name, c.city, o.ord_no, o.ord_date, o.purch_amt AS "OrderAmount"
, s.name, s.commission FROM customer c LEFT OUTER JOIN orders o ON
c.customer_id=o.customer_id LEFT OUTER JOIN saleman s ON
s.salesman_id=o.salesman_id;
```

cust_name	city	ord_no	ord_date	Order Amount	name	commission
Ramesh	Surat	62	2022-10-12	3300	Mansukh	25
Maheesh	Valsad	69	2022-09-21	2300	Ramesh	26
Maheesh	Bardoli	52	2023-01-06	1300	Dharmesh	16
Nand	Rajkot	60	2023-02-02	1800	Meet	14
Vibhav	Bhavnagar	58	2023-01-08	3200	Mayur	13
Virah	Gandhinagar	66	2023-02-09	4500	Sahil	18
Hamid	Valsad	53	2022-12-29	1000	Dhruv	7
Amrut	Mehsana	57	2023-01-24	2300	Raj	12
Manav	Junagadh	67	2023-01-10	2400	Naman	32
Krish	Palanpur	56	2023-01-18	1200	Sohil	3
Rahul	Ahmedabad	64	2022-11-30	900	Jenil	45
Neel	Surat	59	2023-01-10	4500	Mukesh	21
Jayesh	Gandhinagar	51	2023-01-01	2000	Bhavesh	34
Heet	Anand	54	2022-11-09	3000	Joel	11
Kishor	Junagadh	68	2023-01-17	2800	Maheesh	14
Raj	Sachin	63	2022-11-26	6500	Ram	9
Vibhav	Bhavnagar	61	2023-01-19	2100	Virag	19
Jatin	Bhuj	55	2022-12-08	4000	Kishor	8
Bipin	Palitana	65	2022-09-14	2400	Sagar	24
Mitesh	Bhavnagar	70	2023-01-25	500	Heet	27

12. Write a SQL statement to generate a list in ascending order of salespersons who work either for one or more customers or have not yet joined any of the customers

```
SELECT s.name "sales person" FROM saleman s LEFT JOIN customer c ON
c.salesman_id = s.salesman_id ORDER BY s.name;
```

sales person
Bhavesh
Dharmesh
Dhruv
Heet
Jeel
Jenil
Kishor
Maheesh
Mansukh
Mayur
Meet
Mukesh
Naman
Raj
Ram
Ramesh
Sagar
Sahil
Sohil
Virag

13. write a SQL query to list all salespersons along with customer name, city, grade, order number, date, and amount.

```
SELECT c.cust_name, c.city, c.grade, s.name AS "Salesman", o.ord_no, o.ord_date, o.purch_amt FROM customer c RIGHT OUTER JOIN saleman s ON s.salesman_id = c.salesman_id RIGHT OUTER JOIN orders o ON o.customer_id = c.customer_id;
```

The screenshot shows the phpMyAdmin interface with the 'orders' table selected. The table structure includes columns: cust_name, city, grade, Salesman, ord_no, ord_date, and purch_amt. The data contains 20 rows of information, such as Jayesh from Gandhinagar with grade B and order number 51.

cust_name	city	grade	Salesman	ord_no	ord_date	purch_amt
Jayesh	Gandhinagar	B	Dhamesh	51	2023-01-01	2000
Maheesh	Bardoli	B	Sohil	52	2023-01-06	1300
Hamid	Valsad	C	Mansukh	53	2022-12-29	1000
Heet	Anand	D	Virag	54	2022-11-09	3000
Jatin	Bhuj	D	Heet	55	2022-12-08	4000
Krish	Palanpur	D	Jenil	56	2023-01-18	1200
Amrit	Mehsana	D	Mahesh	57	2023-01-24	2300
Vibhav	Bhavnagar	B	Dhruv	58	2023-01-08	3200
Neel	Surat	D	Meet	59	2023-01-10	4500
Nand	Rajkot	D	Jeel	60	2023-02-02	1800
Vibhav	Bhavnagar	C	Naman	61	2023-01-19	2100
Ramesh	Surat	A	Mayur	62	2022-10-12	3300
Raj	Sachin	C	Ram	63	2022-11-26	6500
Rahul	Ahmedabad	A	Sagar	64	2022-11-30	900
Bipin	Palitana	B	Raj	65	2022-09-14	2400
Virah	Gandhinagar	B	Sahil	66	2023-02-09	4500
Manav	Junagadh	C	Heet	67	2023-01-10	2400
Kishor	Junagadh	B	Mukesh	68	2023-01-17	2800
Mahesh	Valsad	A	Ramesh	69	2022-09-21	2300
Mitesh	Bhavnagar	B	Bhavesh	70	2023-01-25	500

14. Write a SQL statement to make a list for the salesmen who either work for one or more customers or yet to join any of the customers. The customer may have placed, either one or more orders on or above order amount 2000 and must have a grade, or he may not have placed any order to the associated supplier.

```
SELECT c.cust_name, c.city, c.grade, s.name AS "Salesman", o.ord_no,
o.ord_date, o.purch_amt FROM saleman s LEFT JOIN customer c ON
c.salesman_id=s.salesman_id LEFT JOIN orders o ON
```

```

o.customer_id=c.customer_id WHERE o.purch_amt>=2000 AND c.grade IS NOT NULL
;

```

The screenshot shows the phpMyAdmin interface for the 'sale' database. The 'saleman' table is selected. The table structure includes columns: cust_name, city, grade, Salesman, ord_no, ord_date, and purch_amt. The data shows various salesmen from different cities like Gandhinagar, Anand, Bhuj, etc., with their respective grades (A, B, C) and purchase amounts ranging from 2000 to 6500.

cust_name	city	grade	Salesman	ord_no	ord_date	purch_amt
Jayesh	Gandhinagar	B	Dharmesh	51	2023-01-01	2000
Heet	Anand	D	Virag	54	2022-11-09	3000
Jatin	Bhuj	D	Heet	55	2022-12-08	4000
Amrut	Mehsana	D	Maheesh	57	2023-01-24	2300
Vibhav	Bhavnagar	B	Dhruv	58	2023-01-08	3200
Neel	Surat	D	Meet	59	2023-01-10	4500
Vibhav	Bhavnagar	C	Naman	61	2023-01-19	2100
Ramesh	Surat	A	Mayur	62	2022-10-12	3300
Raj	Sachin	C	Ram	63	2022-11-26	6500
Bipin	Palitana	B	Raj	65	2022-09-14	2400
Virah	Gandhinagar	B	Sahil	66	2023-02-09	4500
Manav	Junagadh	C	Heet	67	2023-01-10	2400
Kishor	Junagadh	B	Mukesh	68	2023-01-17	2800
Maheesh	Valsad	A	Ramesh	69	2022-09-21	2300

15. Write a SQL statement to generate a list of all the salesmen who either work for one or more customers or have yet to join any of them. The customer may have placed one or more orders at or above order amount 2000, and must have a grade, or he may not have placed any orders to the associated supplier.

```

SELECT c.cust_name,c.city,c.grade, s.name AS "Salesman", o.ord_no,
o.ord_date, o.purch_amt FROM saleman s LEFT JOIN customer c ON
c.salesman_id=s.salesman_id LEFT JOIN orders o ON
o.customer_id=c.customer_id WHERE o.purch_amt>=2000 AND c.grade IS NOT NULL
;

```

The screenshot shows the phpMyAdmin interface for the 'sale' database. The 'saleman' table is selected. The table structure includes columns: cust_name, city, grade, Salesman, ord_no, ord_date, and purch_amt. The data shows various salesmen from different cities like Gandhinagar, Anand, Bhuj, etc., with their respective grades (A, B, C) and purchase amounts ranging from 2000 to 6500.

cust_name	city	grade	Salesman	ord_no	ord_date	purch_amt
Jayesh	Gandhinagar	B	Dharmesh	51	2023-01-01	2000
Heet	Anand	D	Virag	54	2022-11-09	3000
Jatin	Bhuj	D	Heet	55	2022-12-08	4000
Amrut	Mehsana	D	Maheesh	57	2023-01-24	2300
Vibhav	Bhavnagar	B	Dhruv	58	2023-01-08	3200
Neel	Surat	D	Meet	59	2023-01-10	4500
Vibhav	Bhavnagar	C	Naman	61	2023-01-19	2100
Ramesh	Surat	A	Mayur	62	2022-10-12	3300
Raj	Sachin	C	Ram	63	2022-11-26	6500
Bipin	Palitana	B	Raj	65	2022-09-14	2400
Virah	Gandhinagar	B	Sahil	66	2023-02-09	4500
Manav	Junagadh	C	Heet	67	2023-01-10	2400
Kishor	Junagadh	B	Mukesh	68	2023-01-17	2800
Maheesh	Valsad	A	Ramesh	69	2022-09-21	2300

16. Write a SQL statement to generate a report with the customer name, city, order no. order date, purchase amount for only those customers on the list who must have a grade and placed one or more orders or which order(s) have been placed by the customer who neither is on the list nor has a grade.

```
SELECT * FROM customer c INNER JOIN orders o ON
o.customer_id = c.customer_id where c.grade IS NOT NULL;
```

The screenshot shows the 'customer' table in the 'sale' database. The table has columns: customer_id, cust_name, city, grade, salesman_id, ord_no, purch_amt, ord_date, customer_id, and salesman_id. The data consists of 29 rows, with row 14 highlighted in blue. The 'customer' table is part of a schema that includes 'customer', 'orders', and 'salesman' tables.

customer_id	cust_name	city	grade	salesman_id	ord_no	purch_amt	ord_date	customer_id	salesman_id
22	Jayesh	Gandhinagar	B	18	51	2000	2023-01-01	22	8
12	Mahesh	Bardoli	B	12	52	1300	2023-01-06	12	18
16	Hamid	Valsad	C	20	53	1000	2022-12-29	16	3
23	Heet	Anand	D	14	54	3000	2022-11-09	23	15
27	Jatin	Bhuj	D	10	55	4000	2022-12-08	27	17
19	Krish	Palanpur	D	13	56	1200	2023-01-18	19	12
17	Amrut	Mehsana	D	5	57	2300	2023-01-24	17	1
14	Vibhav	Bhavnagar	B	3	58	3200	2023-01-08	14	9
21	Neel	Surat	D	2	59	4500	2023-01-10	21	16
13	Nand	Rajkot	D	15	60	1800	2023-02-02	13	2
26	Vibhav	Bhavnagar	C	11	61	2100	2023-01-19	26	14
10	Ramesh	Surat	A	9	62	3300	2022-10-12	10	20
25	Raj	Sachin	C	6	63	6500	2022-11-26	25	6
20	Rahul	Ahmedabad	A	7	64	900	2022-11-30	20	13
28	Bipin	Palitana	B	1	65	2400	2022-09-14	28	7
15	Virah	Gandhinagar	B	4	66	4500	2023-02-09	15	4
18	Manav	Junagadh	C	10	67	2400	2023-01-10	18	11
24	Kishor	Junagadh	B	16	68	2800	2023-01-17	24	5
11	Mahesh	Valsad	A	19	69	2300	2022-09-21	11	19
29	Mitesh	Bhavnagar	B	8	70	500	2023-01-25	29	10

17. Write a SQL query to combine each row of the salesman table with each row of the customer table

```
SELECT * FROM salesman s CROSS JOIN customer c;
```

The screenshot shows the 'salesman' table in the 'sale' database. The table has columns: salesman_id, name, city, commission, customer_id, cust_name, city, grade, and salesman_id. The data consists of 20 rows, with row 9 highlighted in blue. The 'salesman' table is part of a schema that includes 'customer', 'orders', and 'salesman' tables.

salesman_id	name	city	commission	customer_id	cust_name	city	grade	salesman_id
1	Raj	Surat	12	10	Ramesh	Surat	A	9
2	Meet	Vadodara	14	10	Ramesh	Surat	A	9
3	Dhruv	Bardoli	7	10	Ramesh	Surat	A	9
4	Sahil	Valsad	18	10	Ramesh	Surat	A	9
5	Mahesh	Rajkot	14	10	Ramesh	Surat	A	9
6	Ram	Anand	9	10	Ramesh	Surat	A	9
7	Sagar	Kutch	24	10	Ramesh	Surat	A	9
8	Bhavesh	Ahmedabad	34	10	Ramesh	Surat	A	9
9	Mayur	Surat	13	10	Ramesh	Surat	A	9
10	Heet	Ankleshwar	27	10	Ramesh	Surat	A	9
11	Naman	Mehsana	32	10	Ramesh	Surat	A	9
12	Sohil	Bhavnagar	3	10	Ramesh	Surat	A	9
13	Jenil	Palanpur	45	10	Ramesh	Surat	A	9
14	Virag	Jamnagar	19	10	Ramesh	Surat	A	9
15	Jeel	Amreli	11	10	Ramesh	Surat	A	9
16	Mukesh	Surat	21	10	Ramesh	Surat	A	9
17	Kishor	Vadodara	8	10	Ramesh	Surat	A	9
18	Dharmesh	Surendranagar	16	10	Ramesh	Surat	A	9
19	Ramesh	Bhavnagar	26	10	Ramesh	Surat	A	9
20	Mansukh	Gandhinagar	25	10	Ramesh	Surat	A	9
1	Raj	Surat	12	11	Mahesh	Valsad	A	19
2	Meet	Vadodara	14	11	Mahesh	Valsad	A	19

18. Write a SQL statement to create a Cartesian product between salesperson and customer, i.e. each salesperson will appear for all customers and vice versa for that salesperson who belongs to that city

```
SELECT * FROM saleman s CROSS JOIN customer c where s.city = c.city;
```

salesman_id	name	city	commission	customer_id	cust_name	city	grade	salesman_id
1	Raj	Surat	12	10	Ramesh	Surat	A	9
9	Mayur	Surat	13	10	Ramesh	Surat	A	9
16	Mukesh	Surat	21	10	Ramesh	Surat	A	9
4	Sahil	Valsad	18	11	Mahesh	Valsad	A	19
3	Dhruv	Bardoli	7	12	Mahesh	Bardoli	B	12
5	Mahesh	Rajkot	14	13	Nand	Rajkot	D	15
12	Sohil	Bhavnagar	3	14	Vibhav	Bhavnagar	B	3
19	Ramesh	Bhavnagar	26	14	Vibhav	Bhavnagar	B	3
20	Mansukh	Gandhinagar	25	15	Virah	Gandhinagar	B	4
4	Sahil	Valsad	18	16	Hamid	Valsad	C	20
11	Naman	Mehsana	32	17	Amrut	Mehsana	D	5
13	Jenil	Palanpur	45	19	Krish	Palanpur	D	13
8	Bhavesh	Ahmedabad	34	20	Rahul	Ahmedabad	A	7
1	Raj	Surat	12	21	Neel	Surat	D	2
9	Mayur	Surat	13	21	Neel	Surat	D	2
16	Mukesh	Surat	21	21	Neel	Surat	D	2
20	Mansukh	Gandhinagar	25	22	Jayesh	Gandhinagar	B	18
6	Ram	Anand	9	23	Heet	Anand	D	14
12	Sohil	Bhavnagar	3	26	Vibhav	Bhavnagar	C	11
19	Ramesh	Bhavnagar	26	26	Vibhav	Bhavnagar	C	11
12	Sohil	Bhavnagar	3	29	Mitesh	Bhavnagar	B	8
19	Ramesh	Bhavnagar	26	29	Mitesh	Bhavnagar	B	8

19. Write a SQL statement to create a Cartesian product between salesperson and customer, i.e. each salesperson will appear for every customer and vice versa for those salesmen who belong to a city and customers who require a grade

```
SELECT * FROM saleman s CROSS JOIN customer c where s.city IS NOT NULL AND
c.grade IS NOT NULL;
```

salesman_id	name	city	commission	customer_id	cust_name	city	grade	salesman_id
1	Raj	Surat	12	10	Ramesh	Surat	A	9
2	Meet	Vadodara	14	10	Ramesh	Surat	A	9
3	Dhruv	Bardoli	7	10	Ramesh	Surat	A	9
4	Sahil	Valsad	18	10	Ramesh	Surat	A	9
5	Mahesh	Rajkot	14	10	Ramesh	Surat	A	9
6	Ram	Anand	9	10	Ramesh	Surat	A	9
7	Sagar	Kutch	24	10	Ramesh	Surat	A	9
8	Bhavesh	Ahmedabad	34	10	Ramesh	Surat	A	9
9	Mayur	Surat	13	10	Ramesh	Surat	A	9
10	Heet	Ankleshwar	27	10	Ramesh	Surat	A	9
11	Naman	Mehsana	32	10	Ramesh	Surat	A	9
12	Sohil	Bhavnagar	3	10	Ramesh	Surat	A	9
13	Jenil	Palanpur	45	10	Ramesh	Surat	A	9
14	Virag	Jamnagar	19	10	Ramesh	Surat	A	9
15	Jeel	Amreli	11	10	Ramesh	Surat	A	9
16	Mukesh	Surat	21	10	Ramesh	Surat	A	9
17	Kishor	Vadodara	8	10	Ramesh	Surat	A	9
18	Dharmesh	Surendranagar	16	10	Ramesh	Surat	A	9
19	Ramesh	Bhavnagar	26	10	Ramesh	Surat	A	9
20	Mansukh	Gandhinagar	25	10	Ramesh	Surat	A	9
1	Raj	Surat	12	11	Mahesh	Valsad	A	19
2	Meet	Vadodara	14	11	Mahesh	Valsad	A	19
3	Dhruv	Bardoli	7	11	Mahesh	Valsad	A	19

20. Write a SQL statement to make a Cartesian product between salesman and customer i.e. each salesman will appear for all customers and vice versa for those salesmen who must belong to a city which is not the same as his customer and the customers should have their own grade

```
SELECT * FROM saleman s CROSS JOIN customer c WHERE s.city IS NOT NULL AND c.grade IS NOT NULL AND s.city<>c.city;
```

The screenshot shows the phpMyAdmin interface with the database 'sale' selected. The 'saleman' table is currently being viewed. The table structure includes columns: salesman_id, name, city, commission, customer_id, cust_name, city, grade, and salesman_id (which is a primary key). The data consists of 20 rows, each representing a different salesman and their assigned customer. The 'customer' table is also visible in the schema tree.

salesman_id	name	city	commission	customer_id	cust_name	city	grade	salesman_id
2	Meet	Vadodara	14	10	Ramesh	Surat	A	9
3	Dhruv	Bardoli	7	10	Ramesh	Surat	A	9
4	Sahil	Valsad	18	10	Ramesh	Surat	A	9
5	Mahesh	Rajkot	14	10	Ramesh	Surat	A	9
6	Ram	Anand	9	10	Ramesh	Surat	A	9
7	Sagar	Kutch	24	10	Ramesh	Surat	A	9
8	Bhavesh	Ahmedabad	34	10	Ramesh	Surat	A	9
10	Heet	Ankleswar	27	10	Ramesh	Surat	A	9
11	Naman	Mehsana	32	10	Ramesh	Surat	A	9
12	Sohil	Bhavnagar	3	10	Ramesh	Surat	A	9
13	Jenil	Palanpur	45	10	Ramesh	Surat	A	9
14	Virag	Jammagar	19	10	Ramesh	Surat	A	9
15	Jeel	Amreli	11	10	Ramesh	Surat	A	9
17	Kishor	Vaddoda	8	10	Ramesh	Surat	A	9
18	Dharmesh	Surendranagar	16	10	Ramesh	Surat	A	9
19	Ramesh	Bhavnagar	26	10	Ramesh	Surat	A	9
20	Mansukh	Gandhinagar	25	10	Ramesh	Surat	A	9
1	Raj	Surat	12	11	Mahesh	Valsad	A	19
2	Meet	Vadodara	14	11	Mahesh	Valsad	A	19
3	Dhruv	Bardoli	7	11	Mahesh	Valsad	A	19
5	Mahesh	Rajkot	14	11	Mahesh	Valsad	A	19
6	Ram	Anand	9	11	Mahesh	Valsad	A	19
7	Sagar	Kutch	24	11	Mahesh	Valsad	A	19