Assignment 2 : Retrieve data using join with where clause.

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

Code:

```
select s.name,c.cust_name,s.city from salesman s inner join customer c on s.city = c.city;
```

Output:

	customer_id	cust_name	city	grade	salesman_id
1	2001	HOFFMAN	LONDON	100	1001
2	2002	GIOVANNI	ROME	200	1003
3	2003	LIU	SANJOSE	200	1002
4	2004	GRASS	BERLIN	300	1002
5	2006	CLEMENS	LONDON	100	1001
6	2008	CISNEROS	SANJOSE	300	1007
7	2007	PERIERA	ROME	100	1004

	salesman_id	name	city	commission
1	1001	PEEL	LONDON	0.12
2	1002	SERRES	SANJOSE	0.13
3	1004	NOTIKA	LONDON	0.11
4	1007	RIFKIN	BARCELONA	0.15
5	1003	AXELR...	NEWYORK	0.1

	name	cust_name	city
1	PEEL	HOFFMAN	LONDON
2	PEEL	CLEMENS	LONDON
3	SERRES	LIU	SANJOSE
4	SERRES	CISNEROS	SANJOSE
5	NOTIKA	HOFFMAN	LONDON
6	NOTIKA	CLEMENS	LONDON

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.

Code:

```
select o.ord_no,o.purch_amt,c.cust_name,c.city from orders o inner join customer c on o.customer_id=c.customer_id where o.purch_amt between 500 and 2000
```

Output:

1	3001	18.69	1990-03-10	2008	1007
2	3003	767.19	1990-03-10	2001	1001
3	3002	1900.1	1990-03-10	2007	1004
4	3005	5160.45	1990-03-10	2003	1002
5	3006	1098.16	1990-03-10	2008	1007
6	3009	1713.23	1990-04-10	2002	1003
7	3007	75.75	1990-04-10	2004	1002
8	3008	4723	1990-05-10	2006	1001
9	3010	1309.95	1990-06-10	2004	1002
10	3011	9891.88	1990-06-10	2006	1001

	customer_id	cust_name	city	grade	salesman_id
1	2001	HOFFMAN	LONDON	100	1001
2	2002	GIOVANNI	ROME	200	1003
3	2003	LIU	SANJOSE	200	1002
4	2004	GRASS	BERLIN	300	1002
5	2006	CLEMENS	LONDON	100	1001
6	2008	CISNEROS	SANJOSE	300	1007
7	2007	PERIERA	ROME	100	1004

	ord_no	purch_amt	cust_name	city
1	3003	767.19	HOFFMAN	LONDON
2	3009	1713.23	GIOVANNI	ROME
3	3010	1309.95	GRASS	BERLIN
4	3012	900.45	CLEMENS	LONDON
5	3006	1098.16	CISNER...	SANJO...
6	3002	1900.1	PERIERA	ROME

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

Code:

```
select c.cust_name,c.city,s.name,s.commission from salesman s inner join customer c on s.salesman_id = c.salesman_id;
```

Output:

	salesman_id	name	city	commission
1	1001	PEEL	LONDON	0.12
2	1002	SERRES	SANJOSE	0.13
3	1004	NOTIKA	LONDON	0.11
4	1007	RIFKIN	BARCELONA	0.15
5	1003	AXELROD	NEWYORK	0.1

	customer_id	cust_name	city	grade	salesman_id
1	2001	HOFFMAN	LONDON	100	1001
2	2002	GIOVANNI	ROME	200	1003
3	2003	LIU	SANJOSE	200	1002
4	2004	GRASS	BERLIN	300	1002
5	2006	CLEMENS	LONDON	100	1001
6	2008	CISNEROS	SANJOSE	300	1007
7	2007	PERIERA	ROME	100	1004

	cust_name	city	name	commission
1	HOFFMAN	LONDON	PEEL	0.12
2	GIOVANNI	ROME	AXELROD	0.1
3	LIU	SANJOSE	SERRES	0.13
4	GRASS	BERLIN	SERRES	0.13
5	CLEMENS	LONDON	PEEL	0.12
6	CISNER...	SANJOSE	RIFKIN	0.15
7	PERIERA	ROME	NOTIKA	0.11

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.

Code:

```
select c.cust_name,c.city,s.name,s.commission from salesman s inner join customer c on s.salesman_id = c.salesman_id where s.commission > 0.12;
```

Output:

	salesman_id	name	city	commission
1	1001	PEEL	LONDON	0.12
2	1002	SERRES	SANJOSE	0.13
3	1004	NOTIKA	LONDON	0.11
4	1007	RIFKIN	BARCELONA	0.15
5	1003	AXELROD	NEWYORK	0.1

	customer_id	cust_name	city	grade	salesman_id
1	2001	HOFFMAN	LONDON	100	1001
2	2002	GIOVANNI	ROME	200	1003
3	2003	LIU	SANJOSE	200	1002
4	2004	GRASS	BERLIN	300	1002
5	2006	CLEMENS	LONDON	100	1001
6	2008	CISNEROS	SANJOSE	300	1007
7	2007	PERIERA	ROME	100	1004

	cust_name	city	name	commission
1	LIU	SANJOSE	SERRES	0.13
2	GRASS	BERLIN	SERRES	0.13
3	CISNER...	SANJOSE	RIFKIN	0.15

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.

Code:

```
select c.cust_name,c.city,s.name,s.city,s.commission from salesman s inner join customer c
on s.salesman_id = c.salesman_id where c.city != s.city and s.commission > 0.12;
```

Output:

	salesman_id	name	city	commission
1	1001	PEEL	LONDON	0.12
2	1002	SERRES	SANJOSE	0.13
3	1004	NOTIKA	LONDON	0.11
4	1007	RIFKIN	BARCELONA	0.15
5	1003	AXELROD	NEWYORK	0.1

	customer_id	cust_name	city	grade	salesman_id
1	2001	HOFFMAN	LONDON	100	1001
2	2002	GIOVANNI	ROME	200	1003
3	2003	LIU	SANJOSE	200	1002
4	2004	GRASS	BERLIN	300	1002
5	2006	CLEMENS	LONDON	100	1001
6	2008	CISNEROS	SANJOSE	300	1007
7	2007	PERIERA	ROME	100	1004

	cust_name	city	name	city	commission
1	GRASS	BERLIN	SERRES	SANJOSE	0.13
2	CISNEROS	SANJOSE	RIFKIN	BARCELONA	0.15

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

Code:

```
select o.ord_no,o.ord_date,o.purch_amt,c.cust_name,c.grade,s.name,s.commission from orders o inner join customer c on o.customer_id=c.customer_id inner join salesman s on c.salesman_id=s.salesman_id and s.salesman_id = o.salesman_id
```

Output:

	ord_no	ord_date	purch_amt	cust_name	grade	name	commission
1	3001	1990-03-10	18.69	CISNEROS	300	RIFKIN	0.15
2	3003	1990-03-10	767.19	HOFFMAN	100	PEEL	0.12
3	3002	1990-03-10	1900.1	PERIERA	100	NOTIKA	0.11
4	3005	1990-03-10	5160.45	LIU	200	SERRES	0.13
5	3006	1990-03-10	1098.16	CISNEROS	300	RIFKIN	0.15
6	3009	1990-04-10	1713.23	GIOVANNI	200	AXELROD	0.1
7	3007	1990-04-10	75.75	GRASS	300	SERRES	0.13
8	3008	1990-05-10	4723	CLEMENS	100	PEEL	0.12
9	3010	1990-06-10	1309.95	GRASS	300	SERRES	0.13
10	3011	1990-06-10	9891.88	CLEMENS	100	PEEL	0.12

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.

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.

Code:

```
select c.cust_name,c.city as cus_city,c.grade,s.name,s.city as sal_city from customer c inner join salesman s on c.salesman_id = s.salesman_id order by customer_id
```

Output:

	customer_id	cust_name	city	grade	salesman_id
1	2001	HOFFMAN	LONDON	100	1001
2	2002	GIOVANNI	ROME	200	1003
3	2003	LIU	SANJOSE	200	1002
4	2004	GRASS	BERLIN	300	1002
5	2006	CLEMENS	LONDON	100	1001
6	2008	CISNEROS	SANJOSE	300	1007
7	2007	PERIERA	ROME	100	1004

	salesman_id	name	city	commission
1	1001	PEEL	LONDON	0.12
2	1002	SERRES	SANJOSE	0.13
3	1004	NOTIKA	LONDON	0.11
4	1007	RIFKIN	BARCELONA	0.15
5	1003	AXELROD	NEWYORK	0.1

	cust_name	cus_city	grade	name	sal_city
1	HOFFMAN	LONDON	100	PEEL	LONDON
2	GIOVANNI	ROME	200	AXELROD	NEWYORK
3	LIU	SANJOSE	200	SERRES	SANJOSE
4	GRASS	BERLIN	300	SERRES	SANJOSE
5	CLEMENS	LONDON	100	PEEL	LONDON
6	PERIERA	ROME	100	NOTIKA	LONDON
7	CISNER...	SANJOSE	300	RIFKIN	BARCEL...

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.

Code:

```
select c.cust_name,c.city as cus_city,c.grade,s.name,s.city as sal_city from customer c inner join salesman s on c.salesman_id = s.salesman_id where grade<300 order by customer_id
```

Output:

	customer_id	cust_name	city	grade	salesman_id
1	2001	HOFFMAN	LONDON	100	1001
2	2002	GIOVANNI	ROME	200	1003
3	2003	LIU	SANJOSE	200	1002
4	2004	GRASS	BERLIN	300	1002
5	2006	CLEMENS	LONDON	100	1001
6	2008	CISNEROS	SANJOSE	300	1007
7	2007	PERIERA	ROME	100	1004

	salesman_id	name	city	commission
1	1001	PEEL	LONDON	0.12
2	1002	SERRES	SANJOSE	0.13
3	1004	NOTIKA	LONDON	0.11
4	1007	RIFKIN	BARCELONA	0.15
5	1003	AXELROD	NEWYORK	0.1

	cust_name	cus_city	grade	name	sal_city
1	HOFFMAN	LONDON	100	PEEL	LONDON
2	GIOVANNI	ROME	200	AXELROD	NEWYORK
3	LIU	SANJOSE	200	SERRES	SANJOSE
4	CLEMENS	LONDON	100	PEEL	LONDON
5	PERIERA	ROME	100	NOTIKA	LONDON

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

Code:

```
select o.ord_no,o.purch_amt,c.cust_name,c.city,o.ord_date from customer c left join orders o on o.customer_id=c.customer_id order by ord_date
```

Output:

	ord_no	purch_amt	cust_name	city	ord_date
1	3003	767.19	HOFFMAN	LONDON	1990-03-10
2	3005	5160.45	LIU	SANJOSE	1990-03-10
3	3012	900.45	CLEMENS	LONDON	1990-03-10
4	3001	18.69	CISNER...	SANJOSE	1990-03-10
5	3006	1098.16	CISNER...	SANJOSE	1990-03-10
6	3002	1900.1	PERIERA	ROME	1990-03-10
7	3007	75.75	GRASS	BERLIN	1990-04-10
8	3009	1713.23	GIOVANNI	ROME	1990-04-10

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

Code:

```
select o.ord_no,o.purch_amt,c.cust_name,c.city,o.ord_date from customer c left join orders o on o.customer_id=c.customer_id left join salesman s on c.salesman_id=s.salesman_id
```

Output:

	ord_no	purch_amt	cust_name	city	ord_date
1	3003	767.19	HOFFMAN	LONDON	1990-03-10
2	3009	1713.23	GIOVANNI	ROME	1990-04-10
3	3005	5160.45	LIU	SANJO...	1990-03-10
4	3007	75.75	GRASS	BERLIN	1990-04-10
5	3010	1309.95	GRASS	BERLIN	1990-06-10
6	3008	4723	CLEMENS	LONDON	1990-05-10
7	3011	9891.88	CLEMENS	LONDON	1990-06-10
8	3012	900.45	CLEMENS	LONDON	1990-03-10
9	3001	18.69	CISNER...	SANJO...	1990-03-10
10	3006	1098.16	CISNER...	SANJO...	1990-03-10
11	3002	1900.1	PERIERA	ROME	1990-03-10

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

Code:

```
select * from salesman s left join customer c on s.salesman_id=c.salesman_id order by s.salesman_id
```

Output:

	salesman_id	name	city	commission	customer_id	cust_name	city	grade	salesman_id
1	1001	PEEL	LONDON	0.12	2001	HOFFMAN	LONDON	100	1001
2	1001	PEEL	LONDON	0.12	2006	CLEMENS	LONDON	100	1001
3	1002	SERRES	SANJOSE	0.13	2003	LIU	SANJOSE	200	1002
4	1002	SERRES	SANJOSE	0.13	2004	GRASS	BERLIN	300	1002
5	1003	AXELROD	NEWYORK	0.1	2002	GIOVANNI	ROME	200	1003
6	1004	NOTIKA	LONDON	0.11	2007	PERIERA	ROME	100	1004
7	1007	RIFKIN	BARCELONA	0.15	2008	CISNEROS	SANJOSE	300	1007

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

Code:

```
select o.ord_no,o.purch_amt,c.cust_name,c.city,o.ord_date from salesman s left join customer c on c.salesman_id=s.salesman_id left join orders o on c.customer_id = o.customer_id
```

Output:

	ord_no	purch_amt	cust_name	city	ord_date
1	3003	767.19	HOFFMAN	LONDON	1990-03-10
2	3008	4723	CLEMENS	LONDON	1990-05-10
3	3011	9891.88	CLEMENS	LONDON	1990-06-10
4	3012	900.45	CLEMENS	LONDON	1990-03-10
5	3005	5160.45	LIU	SANJO...	1990-03-10
6	3007	75.75	GRASS	BERLIN	1990-04-10
7	3010	1309.95	GRASS	BERLIN	1990-06-10
8	3002	1900.1	PERIERA	ROME	1990-03-10
9	3001	18.69	CISNER...	SANJO...	1990-03-10

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.
Code:

```
select s.salesman_id,s.name,c.cust_name,o.ord_no,o.purch_amt from salesman s left join customer c on c.salesman_id=s.salesman_id left join orders o on c.customer_id = o.customer_id where c.grade is not null and o.purch_amt>2000
```

Output:

	ord_no	purch_amt	cust_name	city	ord_date
1	3003	767.19	HOFFMAN	LONDON	1990-03-10
2	3008	4723	CLEMENS	LONDON	1990-05-10
3	3011	9891.88	CLEMENS	LONDON	1990-06-10
4	3012	900.45	CLEMENS	LONDON	1990-03-10
5	3005	5160.45	LIU	SANJOSE	1990-03-10
6	3007	75.75	GRASS	BERLIN	1990-04-10
7	3010	1309.95	GRASS	BERLIN	1990-06-10
8	3002	1900.1	PERIERA	ROME	1990-03-10
9	3001	18.69	CISNEROS	SANJOSE	1990-03-10
10	3006	1098.16	CISNEROS	SANJOSE	1990-03-10
11	3009	1713.23	GIOVANNI	ROME	1990-04-10

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.
Code:

```
select s.salesman_id,s.name,c.cust_name,o.ord_no,o.purch_amt from salesman s left join customer c on c.salesman_id=s.salesman_id left join orders o on c.customer_id = o.customer_id

where c.grade is not null and o.purch_amt>2000
```

Output:

	ord_no	purch_amt	cust_name	city	ord_date
1	3003	767.19	HOFFMAN	LONDON	1990-03-10
2	3008	4723	CLEMENS	LONDON	1990-05-10
3	3011	9891.88	CLEMENS	LONDON	1990-06-10
4	3012	900.45	CLEMENS	LONDON	1990-03-10
5	3005	5160.45	LIU	SANJOSE	1990-03-10
6	3007	75.75	GRASS	BERLIN	1990-04-10
7	3010	1309.95	GRASS	BERLIN	1990-06-10
8	3002	1900.1	PERIERA	ROME	1990-03-10
9	3001	18.69	CISNEROS	SANJOSE	1990-03-10
10	3006	1098.16	CISNEROS	SANJOSE	1990-03-10
11	3009	1713.23	GIOVANNI	ROME	1990-04-10

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.

Code:

```
select c.cust_name as "customer name" ,c.city,o.ord_no,o.ord_date,o.purch_amt from customer right join orders o on c.customer_id= o.customer_id;
```

Output:

	Customer Name	city	ord_no	ord_date	purch_amt
1	CISNEROS	SANJOSE	3001	1990-03-10	18.69
2	HOFFMAN	LONDON	3003	1990-03-10	767.19
3	PERIERA	ROME	3002	1990-03-10	1900.1
4	LIU	SANJOSE	3005	1990-03-10	5160.45
5	CISNEROS	SANJOSE	3006	1990-03-10	1098.16
6	GIOVANNI	ROME	3009	1990-04-10	1713.23
7	GRASS	BERLIN	3007	1990-04-10	75.75
8	CLEMENS	LONDON	3008	1990-05-10	4723
9	GRASS	BERLIN	3010	1990-06-10	1309.95
10	CLEMENS	LONDON	3011	1990-06-10	9891.88
11	CLEMENS	LONDON	3012	1990-03-10	900.45

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

Code:

```
select * from salesman full outer join customer on salesman.salesman_id  = customer.salesman_id
```

Output:

	salesman_id	name	city	commission	customer_id	cust_name	city	grade	salesman_id
1	1001	PEEL	LONDON	0.12	2001	HOFFMAN	LONDON	100	1001
2	1001	PEEL	LONDON	0.12	2006	CLEMENS	LONDON	100	1001
3	1002	SERRES	SANJOSE	0.13	2003	LIU	SANJOSE	200	1002
4	1002	SERRES	SANJOSE	0.13	2004	GRASS	BERLIN	300	1002
5	1004	NOTIKA	LONDON	0.11	2007	PERIERA	ROME	100	1004
6	1007	RIFKIN	BARCELONA	0.15	2008	CISNEROS	SANJOSE	300	1007
7	1003	AXELROD	NEWYORK	0.1	2002	GIOVANNI	ROME	200	1003

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

Code:

```
select * from salesman s cross join customer c where s.city=c.city
```

Output:

	salesman_id	name	city	commission	customer_id	cust_name	city	grade	salesman_id
1	1001	PEEL	LONDON	0.12	2001	HOFFMAN	LONDON	100	1001
2	1001	PEEL	LONDON	0.12	2006	CLEMENS	LONDON	100	1001
3	1002	SERRES	SANJOSE	0.13	2003	LIU	SANJOSE	200	1002
4	1002	SERRES	SANJOSE	0.13	2008	CISNEROS	SANJOSE	300	1007
5	1004	NOTIKA	LONDON	0.11	2001	HOFFMAN	LONDON	100	1001
6	1004	NOTIKA	LONDON	0.11	2006	CLEMENS	LONDON	100	1001

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

Code:

```
select * from salesman s cross join customer c where s.city is not null and c.grade is not null
```

Output:

	salesman_id	name	city	commission	customer_id	cust_name	city	grade	salesman_id
1	1001	PEEL	LONDON	0.12	2001	HOFFMAN	LONDON	100	1001
2	1001	PEEL	LONDON	0.12	2002	GIOVANNI	ROME	200	1003
3	1001	PEEL	LONDON	0.12	2003	LIU	SANJOSE	200	1002
4	1001	PEEL	LONDON	0.12	2004	GRASS	BERLIN	300	1002
5	1001	PEEL	LONDON	0.12	2006	CLEMENS	LONDON	100	1001
6	1001	PEEL	LONDON	0.12	2008	CISNEROS	SANJOSE	300	1007
7	1001	PEEL	LONDON	0.12	2007	PERIERA	ROME	100	1004
8	1002	SERRES	SANJOSE	0.13	2001	HOFFMAN	LONDON	100	1001
9	1002	SERRES	SANJOSE	0.13	2002	GIOVANNI	ROME	200	1003
10	1002	SERRES	SANJOSE	0.13	2003	LIU	SANJOSE	200	1002
11	1002	SERRES	SANJOSE	0.13	2004	GRASS	BERLIN	300	1002
12	1002	SERRES	SANJOSE	0.13	2006	CLEMENS	LONDON	100	1001
13	1002	SERRES	SANJOSE	0.13	2008	CISNEROS	SANJOSE	300	1007
14	1002	SERRES	SANJOSE	0.13	2007	PERIERA	ROME	100	1004
15	1004	NOTIKA	LONDON	0.11	2001	HOFFMAN	LONDON	100	1001

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

Code:

```
select * from salesman s cross join customer c where s.city is not null and c.grade is not null and s.city=c.city
```

Output:

	salesman_id	name	city	commission	customer_id	cust_name	city	grade	salesman_id
1	1001	PEEL	LONDON	0.12	2001	HOFFMAN	LONDON	100	1001
2	1001	PEEL	LONDON	0.12	2006	CLEMENS	LONDON	100	1001
3	1002	SERRES	SANJOSE	0.13	2003	LIU	SANJOSE	200	1002
4	1002	SERRES	SANJOSE	0.13	2008	CISNEROS	SANJOSE	300	1007
5	1004	NOTIKA	LONDON	0.11	2001	HOFFMAN	LONDON	100	1001
6	1004	NOTIKA	LONDON	0.11	2006	CLEMENS	LONDON	100	1001