Chapter 5

**Querying Multiple Tables**

Exercise 5-1

SELECT e.emp\_id, e.fname, e.lname, b.name

FROM employee e INNER Join branch b

ON e.assigned\_branch\_id =b.vranch\_id

Exercise 5\_2

SELECT c.cust\_id, p.name , c.fed\_id

From account a INNER JOIN customer c

On a.cust.id=c.cust\_id

INNER JOIN product p

ON a.product\_cd=p.product\_cd

Where c.cust\_type\_cd=’I’;

Exercise 5\_3

SELECT e.emp\_id, e.fname, e.lname

FROM employee e INNER JOIN employee sp

ON e.superior\_emp\_id= sp.emp\_id;

**Working with sets**

Exercise 6-1

1.A union B ={L,M,N,P,O,Q,R,S,T}

2.A union all B={L,M,N,O,P,P,Q,R,S.T}

3. A intersect B={P}

4.A except B={L,M,N,O}

Exercise 6-2

SELECT fname,lname

FROM individual

UNION

SELECT fname, lnme

FROM employee;

**Join Revisited**

Exercise 10-1

SELECT p.name,a.account\_id

FROM product p LEFT JOIN account a

ON p.product\_cd=a.product\_cd

Exercise 10-2

SELECT p.name,a.account\_id

FROM account a RIGHT JOIN product p

ON p.product\_cd=a.product\_cd

Exercise 10-3

SELECT a.account\_id, a.product\_cd,i.fname,i.lname,b.name

FROM account LEFT JOIN individual i

ON a.cust\_id=i.cust\_id

LEFT JOIN business b

ON a.cust\_id=b.cust\_id

**Grouping and aggregates**

Exercise 8-1

SELECT count(\*)

FROM account;

Exercise 8-2

SELECT cust\_id, count(\*)

FROM account

Group by (cust\_id);

Exercise 8-3

SELECT cust\_id, count(\*)

FROM account

Group by (cust\_id)

Having count(\*)>=2;

Exercise 8-4

SELECT sum(avail\_balance), count(\*)

FROM account

GROUP BY (open\_branch\_id), (product\_cd)

HAVING count(\*) >= 1

ORDER BY sum(avail\_balance) DESC;

**Subqueries**

Exercise 9\_1

SELECT account\_id, product\_cd, cust\_id, avail\_balance

FROM account

WHERE IN (SELECT product\_type\_cd FROM product where product\_type\_cd=”LOAN”)

Exercise 9\_2

SELECT account\_id, product\_cd, cust\_id, avail\_balance

FROM account a

WHERE Exists (SELECT 1 from p.product\_cd=a.product\_cd

AND p.product\_type\_cd=”LOAN”)

Conditional logic

Exercise 11-1

SELECT emp\_id

CASE

WHEN title IN (“president”,”title”, “Treasurer”)

THEN “ management”

WHEN title IN (“operations Manger”, “Head Teller”)

THEN “OPERATIONS”

ELSE “unknow”

END

Exercise 11-2

-w3resource SQL Exercise

Boolean and relational operators

1. SELECT customer\_id, cust\_name, city, grade, and salesman\_id

FROM customer

where garde >100:

2. SELECT customer\_id, cust\_name, city, grade, and salesman\_id

FROM customer

where garde >100 AND city=‘New York’

3. SELECT customer\_id, cust\_name, city, grade, and salesman\_id

FROM customer

where garde >100 OR city=‘New York’;

4. SELECT customer\_id, cust\_name, city, grade, and salesman\_id

FROM customer

where city=‘New York’ OR NOT grade >100;

5. SELECT customer\_id, cust\_name, city, grade, and salesman\_id

FROM customer

where NOT (city=‘New York’ OR grade >100);

6. SELECT ord\_no, purch\_amt, ord\_date, customer\_id and salesman\_id

FROM orders

WHERE NOT ((ord\_date='2012-09-10' AND salesman\_id >5005) OR purch\_amt> 1000)

7. SELECT salesman\_id, name, city, and commission.

FROM salesman

WHERE commission 0.10 between 0.12;

8. SELECT ord\_no, purch\_amt, ord\_date, customer\_id, salesman\_id

FROM orders

WHERE purch\_amt< 200 OR NOT (ord\_date>= '2012-02-10' AND customer\_id> 3009)

9. SELECT ord\_no, purch\_amt, ord\_date, customer\_id, salesman\_id

FROM orders

WHERE NOT ((ord\_date>= '2012-02-10' OR customer\_id> 3005) AND purch\_amt< 1000);

Wildcard and special operators

1. SELECT(\*)

FROM salesman

WHERE 'Paris' City or 'Rome'

3. SELECT(\*)

FROM salesman

WHERE city NOT IN ('Paris','Rome' )

4. SELECT(\*)

FROM salesman

WHERE salesman\_id IN (3007,3008 ) OR (3009);

5.SELECT(\*)

FROM salesman

WHERE commission BETWEEN 0.12 AND 0.14;

6.SELECT(\*)

FROM orders

WHERE ord\_no BETWEEN 500 AND 4000 AND purch\_amt NOT IN ( 948.50, 1983.50);

7. SELECT(\*)

FROM orders

WHERE name BETWEEN A AND L;

9. SELECT(\*)

FROM orders

WHERE name LIKE “B%”

10. SELECT(\*)

FROM orders

WHERE name LIKE "%N"

11. SELECT(\*)

FROM orders

WHERE name LIKE "N--I%"

12. SELECT col1

FROM testtable

WHERE name LIKE "%\*\_%" ESCAPE "\*";

13. SELECT col1

FROM testtable

WHERE name NOT LIKE "%\*\_%" ESCAPE "\*";

14. SELECT col1

FROM testtable

WHERE name LIKE "%\*/%" ESCAPE "\*";

15. SELECT col1

FROM testtable

WHERE name NOT LIKE "%\*/%" ESCAPE "\*";

16. SELECT col1

FROM testtable

WHERE name LIKE "%\*\_/\*%" ESCAPE "\*";

18. SELECT col1

FROM testtable

WHERE name LIKE "%#%#%" ESCAPE "#";

19.SELECT col1

FROM testtable

WHERE name NOT LIKE "%#%#%" ESCAPE "#";

20. SELECT (\*)

FROM customer

WHERE grade IS NULL

21. SELECT (\*)

FROM customer

WHERE grade IS NOT NULL

22. SELECT (\*)

FROM emp\_details

WHERE emp\_lname LIKE "D%"

Aggregate function

1. SELECT sum(purch\_amt)

FROM orders

2. SELECT AVG(purch\_amt)

FROM orders

3. SELECT COUNT(DISTIC(salesman\_id))

FROM orders

4. SELECT COUNT(customer\_id)

FROM customer

5. SELECT COUNT(customer\_id)

FROM customer

WHERE garde >1

6. SELECT MAX(purch\_amt)

FROM order

7. SELECT MIN(purch\_amt)

FROM order;

8. SELECT MAX(grade)

FROM customer

GROUP BY city

9. SELECT MAX(purch\_amt),customer\_id,

FROM customer

order by customer\_id

10. SELECT MAX(purch\_amt),ord\_date

FROM customer

GROUP BY ord\_date,customer\_id

11. SELECT MAX(purch\_amt),salesman\_id

FROM customer

WHERE ord\_no='2012-08-17'

GROUP BY salesman\_id

12. SELECT MAX(purch\_amt),ord\_date,customer\_id

FROM customer

GROUP BY ord\_date,customer\_id

HAVING MAX(purch\_amt)>2000.00

13. SELECT MAX(purch\_amt),ord\_date,customer\_id

FROM customer

where MAX(purch\_amt) BETWEEN 2000 AND 6000

GROUP BY ord\_date,customer\_id

14. SELECT MAX(purch\_amt),ord\_date,customer\_id

FROM customer

where MAX(purch\_amt) IN(2000,3000,5760,6000)

GROUP BY ord\_date,customer\_id

15. SELECT MAX(purch\_amt),ord\_date,customer\_id

FROM customer

where customer\_id BETWEEN 3002 AND 3007

GROUP BY customer\_id

16. SELECT MAX(purch\_amt),ord\_date,customer\_id

FROM customer

where customer\_id BETWEEN 3002 AND 3007

GROUP BY customer\_id

HAVING MAX(purch\_amt)> 1000

17. SELECT MAX(purch\_amt),salesman\_id

FROM customer

GROUP BY salesman\_id

HAVING salesman\_id BETWEEN 3002 AND 3007

18. SELECT COUNT(ord\_no)

FROM customer

where sord\_date='2012-08-17';

19. SELECT COUNT(salesman\_id)

FROM customer

GROUP BY city

20. SELECT COUNT(ord\_no)

FROM customer

GROUP BY ord\_date, salesman\_id

21. SELECT AVG(PRO\_PRICE) AS avg\_product\_price

FROM item\_mast

22. SELECT COUNT(PRO\_ID)

FROM item\_mast

WHERE PRO\_PRICE > 350

23. SELECT AVG(PRO\_PRICE),PRO\_ID

FROM item\_mast

group by PRO\_COM

**Formatting output**

1. SELECT salesman\_id, city,name, commission AS "commission %"

FROM salesman

2.SELECT " for", ord\_date, "there are" COUNT(ord\_no) "orders"

FROM order

GROUP BY ord\_date;

3. SELECT \*

FROM order

ORDER by (ord\_no)

4. SELECT \*

FROM order

ORDER by (ord\_date) DESC

5. SELECT \*

FROM order

ORDER by (ord\_date,purch\_amt) DESC

6. SELECT cust\_name,city,grade

FROM customer

ORDER by (customer\_id)

7. SELECT MAX(purch\_amt),ord\_date,salesman\_id

FROM salesman

GROUP BY salesman\_id

ORDER by (salesman\_id,ord\_date);

8. SELECT cust\_name, city, grade

FROM customer

ORDER by 3 DESC

9. SELECT customer\_id, DISC(ord\_no),MAX(purch\_amt)

FROM orders

GROUP BY customer\_id

ORDER BY 2 DESC

**Query Multiple Tables**

1. SELECT c.cust\_name,s.name,s.city

FROM salesman s , customer c

WHERE s.city=c.city;

2. SELECT c.cust\_name,s.name,

FROM salesman s , customer c

WHERE s.salesman\_id=c.salesman\_id

3. SELECT c.cust\_name,s.name,r.ord-name,r.customer\_id,r.salesman\_id

FROM salesman s , customer c,orders r

WHERE s.city<>c.city

AND s.salesman\_id=c.salesman\_id

AND c.salesman\_id=r.salesman\_id

4. SELECT c.cust\_name,r.ord-name

FROM s customer c,orders r

WHERE c.salesman\_id=r.salesman\_id

5. SELECT c.cust\_name,s.name,r.ord-name,r.customer\_id,r.salesman\_id

FROM salesman s , customer c,orders r

WHERE c.grade is NOT NULL AND c.city is NOT NULL

AND s.salesman\_id=c.salesman\_id

AND c.salesman\_id=r.salesman\_id

6. SELECT c.cust\_name AS "Customer",s.city AS "City"

FROM customer c, salesman S

WHERE s.salesman\_id=c.salesman\_id

AND s.commission BETWEEN 0.12 AND 0.14;

7. SELECT R.purch\_amt\*S.commission AS "Commission",c.cust\_name,s.commission AS "commission%"

FROM customer C, salesman S, R orders

WHERE c.grade>=200

AND c.salesman\_id=s.salesman\_id

AND s.salesman\_id =R.salesman\_id

8. SELECT c.customer\_id,c.cust\_name,c.city,c.grade,s.salesman\_id,R.ord\_no,

r.purch\_amt,r.ord\_date,s.salesman\_id

FROM customer C, salesman S, R orders

WHERE r.ord\_date="2012-10-05"

AND c.salesman\_id=s.salesman\_id

AND s.salesman\_id =R.salesman\_id

SQL JOINT

1. SELECT S.salesman\_id,C.cust\_name,S.city

FROM salesman S,customer C

WHERE S.salesman\_id=C.customer

2. SELECT R.ord\_no,R.purch\_amt,C.cust\_name,C.city

FROM orders R,customer C

WHERE R.purch\_amt BETWEEN 500 AND 2000

AND R.salesman\_id=C.salesman\_id

3. SELECT S.salesman\_id, S.commission,C.cust\_name,C.city

FROM salesman INNER join S,customer C

ON C.salesman\_id=C.salesman\_id

4. SELECT S.salesman\_id, S.commission,C.cust\_name,C.city

FROM salesman INNER join S,customer C

ON C.salesman\_id=C.salesman\_id

WHERE C.commission >0.12;

5. SELECT S.salesman\_id, S.commission,C.cust\_name,C.city

FROM salesman INNER join S,customer C

ON C.salesman\_id=C.salesman\_id

WHERE C.commission >0.12 AND S.city<>C.city;

6. ELECT R.ord\_no,R.ord\_date,C.grade,S.salesman\_id, S.commission,C.cust\_name

FROM salesman INNER JOIN S,customer C

ON C.salesman\_id=C.salesman\_id.orders R

INNER JOIN

ON R.salesman\_id=S.salesman\_id

WHERE C.commission >0.12 AND S.ci

7. SELECT (\*)

FROM salesman

NATURAL JOIN customer

NATURAL JOIN orders

8. SELECT C.cust\_name,C.city,C.grade,S.name,S.city

FROM salesman S

LEFT JOIN customer

ON S.salesman\_id=S.salesman\_id

ORDER BY customer\_id

9. SELECT C.cust\_name,C.city,C.grade,S.name,S.city

FROM salesman S

LEFT JOIN customer C

ON S.salesman\_id=S.salesman\_id

WHERE C.grade < 300

ORDER BY C.customer\_id;

10. SELECT C.cust\_name,C.city,C.grade,S.name,S.city

FROM customer C

LEFT JOIN orders R

ON R.salesman\_id=C.salesman\_id

ORDER BY purch\_amt;

11. SELECT C.cust\_name,C.city,R.ord\_no,R.ord\_date,R.purch\_amt,S.name,

S.commission

FROM customer C

LEFT JOIN orders R

ON R.salesman\_id=C.salesman\_id

LEFT JOIN salesman S

ON S.salesman\_id=R.salesman\_id

12. SELECT C.cust\_name,C.city,R.ord\_no,R.ord\_date,R.purch\_amt,S.name,

S.commission

FROM salesman\_id

LEFT JOIN salesman S

ON S.salesman\_id=R.salesman\_id

ORDER BY customer

**Subqueries**

1. SELECT (\*)SELECT (\*)

FROM orders

WHERE Ssalesman\_id(SELECT city

FROM salesman

WHERE city='London')

2.FROM orders

WHERE salesman\_id(SELECT salesman\_id

FROM salesman

WHERE name='Paul Adam')

3. SELECT (\*)

FROM orders

WHERE salesman\_id(SELECT customer\_id

FROM orders

WHERE customer\_id=3007)

4. SELECT (\*)

FROM orders

WHERE purch\_amt(SELECT AVG(purch\_amt)

FROM orders

WHERE ord\_date="2012-10-10")

5. SELECT (\*)

FROM orders

WHERE salesman\_id(SELECT city

FROM salesman

WHERE city="New York")

6. SELECT commission

FROM salesman

WHERE city="Paris")

8. SELECT count(\*), grade

FROM customer

GROUP BY grade > (select AVG(grade)

FROM customer

WHERE city="New York")

9. SELECT (\*)

FROM orders

WHERE salesman\_id =(SELECT MAX (commission)

FROM salesman

WHERE commission=MAX(comission)

10. SELECT O.ord\_no, O.purch\_mat,O.ord\_date,O.customer\_id, O.salesman\_id,C.cust\_name

FROM orders O, customer C

WHERE customer\_id =(SELECT customer\_id)

FROM C

WHERE customer\_id=(SELECT customer\_id

FROM O

WHERE O.ord\_date=2012-08-17)

11. SELECT salesman\_id,name

FROM salesman

GROUP BY salesman\_id

HAVEING 1>(SELECT salesman\_id

FROM customers)

12. SELECT (\*)

FROM orders a

WHERE purch\_amt>(SELECT AVG(purch\_amt)

FROM orders b

WHERE a.customer\_id=b.customer\_id)

13. SELECT (\*)

FROM orders a

WHERE purch\_amt>=(SELECT AVG(purch\_amt)

FROM orders b

WHERE a.customer\_id=b.customer\_id)

14. SELECT SUM (purch\_amt)

FROM orders a

GROUP BY ord\_date

HAVING SUM(purch\_amt)>(select MAX(purch\_amt)+1000

FROM orders b

WHERE a.ord\_date=b.ord\_date)

15. SELECT count(\*)

FROM customer a

GROUP BY customer\_id

HAVING 1<(SELECT customer\_id

FROM customer b

WHERE city="London")

16. SELECT count(\*)

FROM salesman

WHERE salesman\_id= (SELECT DISTICINCT(salesman\_id)

FROM customer

WHERE customer\_id>1)

17. SEECT (\*)

FROM SALESMAN

WHERE EXIEST salesman\_id(count(\*)

FROM customer

where customer\_id(count(\*)=1)

18. SEECT (\*)

FROM SALESMAN

WHERE salesman\_id IN(SELECT salesman\_id

FROM CUSTOMER

WHERE customer\_id(SELECT customer\_id

FROM orders

WHERE count(customer\_id)>1)

19. SEECT (\*)

FROM SALESMAN

WHERE salesman\_id IN (SELECT salesman\_id

FROM customer

WHERE salesman.city =salesman.city)

20. SEECT (\*)

FROM SALESMAN

WHERE salesman\_id ANY(SELECT salesman\_id

FROM customer

WHERE salesman.city =salesman.city)

21.