**Assignment 2**

**1)write a SQL query to find the salesperson and customer who reside in the same city.**

**Return Salesman, cust\_name and city**

select name,cust\_name,salesman.city from salesman inner join customer

on salesman.city=customer.city



**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 cust\_name,ord\_no,purch\_amt,city from customer inner join orders

on customer.customer\_id=orders.customer\_id

where purch\_amt between 500 and 2000



**3)write a SQL query to find the salesperson(s) and the customer(s) he represents.**

**Return Customer Name, city, Salesman, commission**

select name,cust\_name,salesman.city,commission from salesman inner join customer

on salesman.salesman\_id=customer.salesman\_id



**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 name,cust\_name,customer.city,commission from salesman inner join customer

on salesman.salesman\_id=customer.salesman\_id where commission>(0.12)



**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 salesman.city as sales\_person\_city,name,cust\_name,customer.city as customer\_city,commission from salesman inner join customer

on salesman.city!=customer.city and salesman.salesman\_id=customer.salesman\_id where commission>(0.12)



**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 ord\_no,ord\_date,purch\_amt,cust\_name,name,commission,grade

from orders inner join customer

on orders.customer\_id=customer.customer\_id

inner join salesman

on salesman.salesman\_id=orders.salesman\_id



**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

name as sales\_name

,salesman.city as sales\_city,commission,cust\_name,salesman.salesman\_id,customer.customer\_id

grade,ord\_no,ord\_date,purch\_amt from salesman join customer

on salesman.salesman\_id=customer.salesman\_id

join orders on orders.customer\_id=customer.customer\_id



**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 name as salesman\_name,cust\_name,customer.city,salesman.city,grade,customer\_id from salesman

full outer join customer

on salesman.salesman\_id=customer.salesman\_id

order BY customer\_id;



**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 name as salesman\_name,cust\_name,customer.city as cutomer\_city,salesman.city as salesman\_city,grade,customer\_id from salesman

full outer join customer

on salesman.salesman\_id=customer.salesman\_id

where customer.grade<300

order BY customer\_id ;



**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 cust\_name,city,ord\_no,ord\_date,purch\_amt as order\_amont from customer left outer join orders

on customer.customer\_id=orders.customer\_id

order by ord\_date



**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 cust\_name,customer.city,ord\_no,ord\_date,purch\_amt,name,commission

from customer left outer join orders

on customer.customer\_id=orders.customer\_id

left outer join salesman

on salesman.salesman\_id=orders.salesman\_id



**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**

/\* in this first column of customer join with salesman

and in join all rows of salesman is there and the rows which

dont match that are in output also but that particular row there

is NULL value in customer table rows\*/

SELECT cust\_name,customer.city,grade,

name AS Salesman, salesman.city

FROM customer

RIGHT OUTER JOIN salesman

ON salesman.salesman\_id=customer.salesman\_id

ORDER BY salesman.salesman\_id;

**13)write a SQL query to list all salespersons along with customer name, city, grade,**

**order number, date, and amount.**

select cust\_name,salesman.city,grade,ord\_no,ord\_date,purch\_amt,name as salesman

from salesman left outer join customer

on salesman.salesman\_id=customer.salesman\_id

left outer join orders

on orders.salesman\_id=salesman.salesman\_id



**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 cust\_name,customer.city,grade,

name AS salesman,

ord\_no, ord\_date, purch\_amt from customer right outer join salesman

on customer.salesman\_id=salesman.salesman\_id

left outer join orders on customer.customer\_id=orders.customer\_id where purch\_amt>2000

and grade is not null



**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 cust\_name,city,ord\_no,ord\_date,purch\_amt from customer

Full OUTER JOIN orders

on customer.customer\_id=orders.customer\_id where customer.grade is not null



**17)Write a SQL query to combine each row of the salesman table with each row of the**

**customer table**

select \* from salesman cross join customer



**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 salesman cross join customer where salesman.city is not null



**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 salesman cross join customer where salesman.city is not null

and customer.grade is not null



**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 salesman cross join customer where salesman.city!=customer.city

and customer.grade is not null

