**Lab 6**

Using the university schema, answer the following:

1. Retrieve the name of all instructors along with their department names and department building name.
2. Find all courses taught in the 5th semester in 2010
3. Find all courses taught in the 5th semester in 2017

Use Union, Union All, Intersect, intersect all, except

Select a1 from A

Union

Select b1 from B

Here b1 of B and a1 of A must be compatible (same column with same type)

1. Find all the courses taught in 5th semester 2010 or in 5th semester 2017 or both
2. Find all the courses taught in 5th semester 2010 or in 5th semester 2017 or both (retain duplicates)
3. Find all the courses taught in 5th semester 2010 as well as in 5th semester 2017.
4. Find all the courses taught in 5th semester 2010 but not in 5th semester 2017.
5. Find the total number of instructors who teach a course in the 5th semester 2017
6. Find the average salary of each department.
7. Find the number of instructors in each department who teach a course in the 5th semester 2017
8. Find the average salary of each department where the average salary is more than 45000

**Part 2**

create table salesman

( salesman\_id int primary key,name varchar(50),city varchar(40),commission numeric (3,2));

insert into salesman values(5001,'James Hoog','New York',0.15)

insert into salesman values(5002,'Nail Knite','Paris',0.13)

insert into salesman values(5005,'Pit Alex','London',0.11)

insert into salesman values(5006,'Mc Lyon','Paris',0.14)

insert into salesman values(5003,'Lauson Hen',NULL,0.12)

insert into salesman values(5007,'Paul Adam','Rome',0.13)

select \* from salesman

create table customer1

(customer\_id int primary key,cust\_name varchar(80),city varchar(80),grade int, salesman\_id int references salesman);

insert into customer1 values(3002,'Nick Rimando','New York',100,5001)

insert into customer1 values(3005,'Graham Zusi','California',200,5002)

insert into customer1 values(3001,'Brad Guzan','London',NULL,5005)

insert into customer1 values(3004,'Fabian Johns','Paris',300,5006)

insert into customer1 values(3007,'Brad Davis','New York',200,5001)

insert into customer1 values(3009,'Geoff Camero','Berlin',100,5003)

insert into customer1 values(3008,'Julian Green','London',300,5002)

insert into customer1 values(3003,'Jozy Altidor','Moncow',200,5007)

select \* from customer1

create table orders1

(ord\_no int primary key,purch\_amt numeric(10,2),ord\_date date,customer\_id int references customer1,salesman\_id int references salesman);

insert into orders1 values(70001,150.5,'2012-10-05',3005,5002)

insert into orders1 values(70009,270.65,'2012-09-10',3001,5005)

insert into orders1 values (70002,65.26,'2012-10-05',3002,5001)

insert into orders1 values (70004,110.5,'2012-08-17',3009,5003)

insert into orders1 values (70007,948.5,'2012-09-10',3005,5002)

insert into orders1 values (70005,2400.6,'2012-07-27',3007,5001)

insert into orders1 values (70008,5760,'2012-09-10',3002,5001)

insert into orders1 values (70010,1983.43,'2012-10-10',3004,5006)

insert into orders1 values (70003,2480.4,'2012-10-10',3009,5003)

insert into orders1 values (70012,250.45,'2012-06-27',3008,5002)

insert into orders1 values (70011,75.29,'2012-08-17',3003,5007)

insert into orders1 values (70013,3045.6,'2012-04-25',3002,5001)

select \* from orders1

1. Write a SQL statement to know which salesman are working for which customer.
2. Write a SQL statement to find the list of customers who appointed a salesman for their jobs who gets more than 12% commission from the company.
3. Write a SQL statement to find the list of customers who appointed a salesman for their jobs who does not live in same city where there customer lives, and gets a commission above 12%.
4. Write a SQL statement to find the details of order i.e. order number, order date, amount of order, which customer gives the order and which salesman works for that customer and how much commission he gets for an order.
5. Write a SQL statement to make a list in ascending order for the customer who works either through a salesman or by own.
6. Write a SQL statement to make a list in ascending order for the customer who holds a grade less than 300 and works either through a salesman or by own.
7. 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 find that either any of the existing customer have not placed any order or placed an order.
8. Write a SQL statement to make a report with customer name, city, order number, order date, order amount, salesman name and commission to find that either any of the existing customer have placed no order or placed an order by their salesman or by own
9. Write a SQL statement to make a list in ascending order for the salesmen who works either for one or more customer or not yet joined under any of the customer
10. Write a SQL statement to make a list for the salesmen who works either for one or more customer or not yet join under any of the customer who placed either an order or no order.
11. Write a SQL statement to make a list for the salesmen who either work for one or more customer or yet to join any of the customer. The customer, may have placed, either an order of or above order amount 2000 and must have a grade, or he may not have placed any order.
12. Write a SQL statement to make a cartesian product between salesman and customer i.e. each salesman will appear for all customer and vice versa
13. Write a SQL statement to make a cartesian product between salesman and customer i.e. each salesman will appear for all customer and vice versa for those customer who belongs to the same city.
14. Write a SQL statement to make a cartesian product between salesman and customer i.e. each salesman will appear for all customer and vice versa for those salesmen who belongs to the same city and the customers who must have a grade