1. **Consider the following schema for Insurance database:**

PERSON (dr_id, name, address);
CAR (reg_no, model, year);
ACCIDENT (report_no, acc_date, loc);
OWNS (dri_id, reg_no);
PARTICIPATED (dr_id, reg_no, rep_no, dm_amt);

CREATION OF TABLE:

```
CREATE table person
(dr_id number(10) primary key ,
name varchar(10),
address varchar(20));

CREATE table car
(reg_no varchar(10)  primary key,
model varchar(10),
year number(4));

CREATE table accident
(report_no number(2)  primary key,
acc_date date,
loc varchar(10));

CREATE table owns
(dr_id number(10) references person(dr_id),
reg_no varchar(10) references car(reg_no));

CREATE table participated
(dr_id number(10) references person(dr_id),
reg_no varchar(10) references car(reg_no),
rep_no number(2) references accident(report_no),
dm_amt number(10));
```

INSERTION OF VALUES:

```
INSERT into person values(111,'Rakesh ','Tumkur');
INSERT into person values(112,'Raju','Bangalore');
INSERT into person values(113,'Kiran','Mysore');
INSERT into person values(114,'Ramu','Mandya');
INSERT into person values(115,'Ravi','Ramanagara');
INSERT into person values(116,'Suresh','Karwar');
INSERT into person values(117,'Srinivas','Karwar');

INSERT into car values('KA06TK532','Qwid',2020);
INSERT into car values('KA02P1994','Thar',2023);
INSERT into car values('KA03SE008','Hyundai',2022);
INSERT into car values('KA04MC1995','Hyundai',2017);
INSERT into car values('KA05ES6789','Kia',2022);
INSERT into car values('KA04MD1076','Ford',2021);

INSERT into accident values(11,'01-Jan-21','Manglore');
INSERT into accident values(12,'18-Feb-23','Banglore');
INSERT into accident values(13,'20-May-22','Tumkur');
INSERT into accident values(14,'23-JUN-20','Manglore');
```

```
INSERT into accident values(15,'31-DEC-19','Hubli');
INSERT into owns values(111,'KA06TK532');
INSERT into owns values(112,'KA02P1994');
INSERT into owns values(113,'KA03SE008');
INSERT into owns values(114,'KA04MC1995');
INSERT into owns values(115,'KA05ES6789');
INSERT into owns values(116,'KA04MD1076');
INSERT into owns values(117,'KA04MD1076');

INSERT into participated values(111,'KA06TK532',11,75000);
INSERT into participated values(112,'KA02P1994',12,160000);
INSERT into participated values(113,'KA03SE008',13,30000);
INSERT into participated values(114,'KA04MC1995',14,125000);
INSERT into participated values(115,'KA05ES6789',15,100000);
```

QUERY:

1. Display the details of person in alphabetical order of name.
Ans:

```
SELECT *
FROM person
ORDER BY name;
```

O/P:

DR_ID	NAME	ADDRESS
113	Kiran	Mysore
112	Raju	Bangalore
111	Rakesh	Tumkur
114	Ramu	Mandya
115	Ravi	Ramanagara
117	Srinivas	Karwar
116	Suresh	Karwar

2. Display the details of accident in Banglore and damage amount more than 50000.
Ans:

```
SELECT *
FROM accident a, participated p
WHERE a.report_no=p.rep_no and
a.loc='Banglore' and
p.dm_amt >=50000;
```

O/P:

REPORT_NO	ACC_DATE	LOC	DR_ID	REG_NO	REP_NO	DM_AMT
12	18-FEB-23	Banglore	112	KA02P1994	12	160000

3. Display the details of accident of Tumkur car in Manglore location.
Ans:

```
SELECT *
FROM accident a, participated p
WHERE a.loc='Manglore' and
a.report_no=p.rep_no and
p.reg_no like 'KA06%';
```

O/P:

REPORT_NO	ACC_DATE	LOC	DR_ID	REG_NO	REP_NO	DM_AMT
11	01-JAN-21	Manglore	111	KA06TK532	11	75000

4. Display the maximum damage amount of specific car model.
Ans:

```
SELECT max(dm_amt)
FROM car c, participated p
WHERE c.reg_no=p.reg_no and
        c.model='&model';
```

O/P:

```
Enter value for model: Hyundai

MAX(DM_AMT)
-----------
125000
```

5. Display the details of accident WHERE damage amount is more than the average damage amount of Hyundai model.
Ans:

```
SELECT *
FROM accident a, participated p
WHERE a.report_no=p.rep_no and
p.dm_amt < (SELECT avg(dm_amt)
              FROM car c, participated p
             WHERE c.reg_no=p.reg_no and
              c.model='Hyundai');
```

O/P:

REPORT_NO	ACC_DATE	LOC	DR_ID	REG_NO	REP_NO	DM_AMT
11	01-JAN-21	Manglore	111	KA06TK532	11	75000
13	20-MAY-22	Tumkur	113	KA03SE008	13	30000

6. Display the car details whose model is Kia and manufactured during 2022.
Ans:

```
SELECT *
FROM car
WHERE model='Kia' and year=2022;
```

O/P:

REG_NO	MODEL	YEAR
KA05ES6789	Kia	2022

7. Display the alphabetical order of person name, who are involved in accidents and the damage amount is more than 100000.
Ans:

```
SELECT name
FROM person p, participated pa
WHERE p.dr_id=pa.dr_id and
 pa.dm_amt>100000
ORDER BY p.name;
```

O/P:

```
NAME
----------
Raju
Ramu
```

8.Add new accident to the database.
Ans:

```
INSERT into accident values(16,'25-Dec-2021','Mandya');
1 row created.
INSERT into participated values (116,'KA04MD1076',16,34000);
1 row created
```

O/P:

SELECT * FROM accident;

REPORT_NO	ACC_DATE	LOC
11	01-JAN-21	Manglore
12	18-FEB-23	Banglore
13	20-MAY-22	Tumkur
14	23-JUN-20	Manglore
15	31-DEC-19	Hubli
16	25-DEC-21	Mandya

SELECT * FROM participated;

DR_ID	REG_NO	REP_NO	DM_AMT
111	KA06TK532	11	75000
112	KA02P1994	12	160000
113	KA03SE008	13	30000
114	KA04MC1995	14	125000
115	KA05ES6789	15	100000
116	KA04MD1076	16	34000

9. Update the damage amount to 40000 for the car with a specific register number in the accident with report number 15 to 30.
Ans:

```
UPDATE participated
SET dm_amt=40000
WHERE rep_no >15 and rep_no <30 and
reg_no='&reg_no';
```

O/P:

```
Enter value for reg_no: KA04MD1076
```

1 row updated.

SELECT * FROM participated;

DR_ID	REG_NO	REP_NO	DM_AMT
111	KA06TK532	11	75000
112	KA02P1994	12	160000
113	KA03SE008	13	30000
114	KA04MC1995	14	125000
115	KA05ES6789	15	100000
116	KA04MD1076	16	40000

10. Find the total number of people who owned cars that involved in accidents in 2022.
Ans:

```
SELECT count(*) as no_of_people
FROM owns o, accident a, participated p
WHERE o.dr_id = p.dr_id and
a.report_no = p.rep_no and
a.acc_date like '%22';
```

O/P:

NO_OF_PEOPLE
1

2. Consider the following database of student enrollment in courses & books adopted for each course.

STUDENT (usn, name, major, bdate).
COURSE (courseno, cname, dept)
TEXT (book_ISBN, book_title, publisher, author)
ENROLL (usn, courseno sem, marks)
BOOKADOPTION (courseno, sem, book_ISBN)

CREATION OF TABLE:

```
CREATE table student
(usn varchar(10) primary key,
name varchar(10),
major varchar(5),
bdate date);

CREATE table course
(courseno varchar(10) primary key,
cname varchar(10),
dept varchar(10));

CREATE table text
(book_ISBN number(15) primary key,
book_title varchar(25),
publisher varchar(25),
author varchar(25));

CREATE table enroll
```

```
(usn varchar(10) references student (usn),
courseno varchar(10) references course(courseno),
sem number(5),
marks number(3));

CREATE table book_adoption
(courseno varchar(10) references course(courseno),
sem number(5),
book_ISBN number(15) references text (book_ISBN));
```

INSERTION OF VALUES:

```
INSERT into student values('21CS051','Rani','CS','31-Jul-2003');
INSERT into student values('21CS061','Namyatha','CS','17-Jun-2004');
INSERT into student values('21EC062','Navya','EC','26-May-2003');
INSERT into student values('21EE062','Harshitha','EE','21-Nov-2003');
INSERT into student values('21IS061','Raji','IS','17-Jun-2005');
INSERT into student values('21IS051','Suresh','IS','27-Jan-2003');

INSERT into course values('CS5TH1','DS','CSE');
INSERT into course values('CS5TH2','OOMD','CSE');
INSERT into course values('EC5TH1','DC','ECE');
INSERT into course values('EC5TH2','DBMS','ECE');
INSERT into course values('EE5TH1','WT','EEE');
INSERT into course values('EE5TH2','Unix','EEE');
INSERT into course values('IS5TH1','Python','ISE');
INSERT into course values('IS5TH2','DIP','ISE');

INSERT into text values(11111,'Fundamentals of Web', 'Pearson', 'Richard Hoar');
INSERT into text values (11112,'Learning Bootstrap','PackT', 'Matt Lambert');
INSERT into text values (11113,'Java the reference','Tata McGraw Hill', 'Jim Keogh');
INSERT into text values (22221,'OOMD with UML','Pearson', 'Thamdhere');
INSERT into text values (33331,'Data communication','Pearson', 'Forouzan');
INSERT into text values (44441,'Database System','Pearson', 'Navathe');
INSERT into text values (55551,'Digital Image Processing','Pearson', 'Wood');
INSERT into text values (66661,'Fundamentals of Python','Cyber Plus', 'Halterman');
INSERT into text values (77771,'Introduction to Algorithm','Hall of India', 'Cormen');
INSERT into text values (77772,'Algorithm design','Pearson', 'Tardos');
INSERT into text values (88881,'Unix Design Pearson','Pearsonupadte', 'Rochkind');

INSERT into enroll values('21CS051','CS5TH1',5,92);
INSERT into enroll values('21CS051','CS5TH2',5,75);
INSERT into enroll values('21CS061','CS5TH1',5,85);
INSERT into enroll values('21EC062','EC5TH1',5,50);
INSERT into enroll values('21EE062','EE5TH1',5,60);
INSERT into enroll values('21IS061','IS5TH1',5,90);
INSERT into enroll values('21IS051','IS5TH2',5,78);

INSERT into book_adoption values('EE5TH1',5,11111);
INSERT into book_adoption values('EE5TH1',5,11112);
INSERT into book_adoption values('EE5TH1',5,11113);
INSERT into book_adoption values('CS5TH2',5,22221);
INSERT into book_adoption values('EC5TH1',5,33331);
INSERT into book_adoption values('EC5TH2',5,44441);
INSERT into book_adoption values('IS5TH2',5,55551);
INSERT into book_adoption values('IS5TH1',5,66661);
```

```
INSERT into book_adoption values('CS5TH1',5,77771);
INSERT into book_adoption values('CS5TH1',5,77772);
INSERT into book_adoption values('EE5TH2',5,88881);
```

QUERIES:

1. Display the details of all the students who born during 2004.

Ans:

```
SELECT *
FROM student
WHERE bdate like '%04';
```

O/P:

USN	NAME	MAJOR	BDATE
21CS061	Namyatha	CS	17-JAN-04

2. Display all the textbook details which is published by Pearson.

Ans:

```
SELECT *
FROM text
WHERE publisher='Pearson';
```

O/P:

BOOK_ISBN	BOOK_TITLE	PUBLISHER	AUTHOR
11111	Fundamentals of Web	Pearson	Richard Hoar
22221	OOMD with UML	Pearson	Thamdhere
33331	Data communication	Pearson	Forouzan
44441	Database System	Pearson	Navathe
55551	Digital Image Processing	Pearson	Wood
77772	Algorithm design	Pearson	Tardos
88881	Unix Design	Pearson	Rochkind

3. Display course number and course name which has adopted more than 2 books.

Ans:

```
SELECT c.courseno, c.cname
FROM course c, book_adoption b
WHERE c.courseno=b.courseno
GROUP BY c.courseno, c.cname having
count(*) >2;
```

O/P:

COURSENO	CNAME
EE5TH1	WT

4. Display the students who have scored more than 80 and enrolled for DS course.

Ans:

```
SELECT *
FROM enroll e, student s
WHERE e.usn=s.usn and e.courseno='CS5TH1' and
e.marks >80;
```

O/P:

USN	COURSENO	SEM	MARKS	USN	NAME	MAJOR	BDATE
21CS051	CS5TH1	5	92	21CS051	Rani	CS	31-JUL-03
21CS061	CS5TH1	5	85	21CS061	Namyatha	CS	17-JAN-04

5.Display the number of students in each department.
Ans:

```
SELECT dept, count(*) as total_student
FROM course
GROUP BY dept;
```

O/P:

DEPT	TOTAL_STUDENT
CSE	2
ECE	2
EEE	2
ISE	2

6. Display all the textbook written by author Thamdhere.
Ans:

```
SELECT book_title
FROM text
WHERE author='Thamdhere';
```

O/P:

BOOK_TITLE
OOMD with UML

7. Demonstrate how you add new textbook to the database and make this book adopted by some department.
Ans:

```
Insert into text values (77773,'Internet of Thinking','Pearson','David');
1 row created.
Insert into book_adoption values('CS5TH1',5,77773);
 1 row created.
```

O/P:

SELECT * FROM text;

BOOK_ISBN	BOOK_TITLE	PUBLISHER	AUTHOR
11111	Fundamentals of Web	Pearson	Richard Hoar
11112	Learning Bootstrap	PackT	Matt Lambert
11113	Java the reference	Tata McGraw Hill	Jim Keogh
22221	OOMD with UML	Pearson	Thamdhere
33331	Data communication	Pearson	Forouzan
44441	Database System	Pearson	Navathe
55551	Digital Image Processing	Pearson	Wood
66661	Fundamentals of Python	Cyber Plus	Halterman

BOOK_ISBN	BOOK_TITLE	PUBLISHER	AUTHOR
77771	Introduction to Algorithm	Hall of India	Cormen
77772	Algorithm design	Pearson	Tardos
88881	Unix Design	Pearson	Rochkind
77773	Internet of Thinking	Pearson	David

SELECT * FROM book_adoption;

COURSENO	SEM	BOOK_ISBN
EE5TH1	5	11111
EE5TH1	5	11112
EE5TH1	5	11113
CS5TH2	5	22221
EC5TH1	5	33331
EC5TH2	5	44441
IS5TH2	5	55551
IS5TH1	5	66661
CS5TH1	5	77771
CS5TH1	5	77772
EE5TH2	5	88881
CS5TH1	5	77773

8.List any department that has all it's adopted book published by specific publisher.
Ans:

```
SELECT c.dept
FROM text t, book_adoption b, course c
WHERE c.courseno=b.courseno and
t.book_ISBN=b.book_ISBN and
t.publisher='Pearson'   MINUS
(SELECT c.dept
FROM text t, book_adoption b, course c
WHERE c.courseno=b.courseno and
t.book_ISBN=b.book_ISBN and
t.publisher!='Pearson');
```

O/P:

DEPT
ECE

9.Produce a list of textbook like course number, book_ISBN, title in alphabetical order of courses offered by CSE department that use more than 2 books.
Ans:

```
SELECT c.courseno, t.book_ISBN, t.book_title, count(*)
FROM course c, text t, book_adoption b
WHERE c.courseno=b.courseno and
b.book_ISBN=t.book_ISBN and
dept='CSE' and
c.courseno IN (SELECT ba.courseno
                FROM book_adoption ba
               GROUP BY ba.courseno
               HAVING count(courseno)>=2)
GROUP BY c.courseno,t.book_ISBN, t.book_title;
```

O/P:

COURSENO	BOOK_ISBN	BOOK_TITLE	COUNT(*)
CS5TH1	77771	Introduction to Algorithm	1
CS5TH1	77772	Algorithm design	1
CS5TH1	77773	Internet of Thinking	1

10. Display the maximum and minimum number of students who are registered for any course.
Ans:

```
SELECT max(count(*)) as max,  min(count(*)) as min
FROM student s, course c,enroll e
WHERE s.usn=e.usn and c.courseno=e.courseno
GROUP BY c.cname;
```

O/P:

MAX	MIN
2	1

3.Consider the following schema for a Library Database:

BOOK (*Book_id, Title, Publisher_Name, Pub_Year*)
BOOK_AUTHORS (Book_id, Author_*Name*)
PUBLISHER (*Name, Address, Phone*)
BOOK_COPIES (*Book_id, Branch_id, No-of_Copies*)
BOOK_LENDING (*Book_id, Branch_id, Card_No, Date_Out, Due_Date*)
LIBRARY_BRANCH (*Branch_id, Branch_Name, Address*)

CREATION OF TABLE:

```
CREATE table book
(book_id number (10) primary key,
title varchar (20),
publisher_name varchar (20),
pub_year number (20));

CREATE table book_authors
(book_id number (10) references book (book_id) on delete cascade ,
author_name varchar (20),
primary key (book_id, author_name));

CREATE table publisher
(name varchar (20) ,
address varchar (20),
phone number (10));

CREATE table library_branch
(branch_id number (10) primary key ,
branch_name varchar (50) ,
address varchar (50));


CREATE table book_copies
(book_id number (10) references book (book_id) on delete cascade ,
```

```
branch_id number (10) references library_branch(branch_id),
no_of_copies number (10),
primary key (book_id, branch_id));

CREATE table card
(card_no number (10) primary key);

CREATE table book_lending
(book_id number (10) references book (book_id) on delete cascade,
branch_id number (10) references library_branch(branch_id) ,
card_no number (10) references card (card_no),
date_out date,
due_date date,
primary key (book_id, branch_id, card_no));
```

INSERTION OF VALUES:

```
INSERT into book values (1,'DBMS', 'Mcgraw-Hill',2017);
INSERT into book values (2,'ADBMS', 'Mcgraw-Hill',2016);
INSERT into book values (3,'CN', 'Pearson',2016);
INSERT into book values (4,'CG', 'Grupo Planeta',2015);
INSERT into book values (5,'OS', 'Pearson',2016);

INSERT into book_authors values (1,'Navathe');
INSERT into book_authors values (2,'Navathe');
INSERT into book_authors values (3,'Tanenbaum');
INSERT into book_authors values (4,'Edward Angel');
INSERT into book_authors values (5,'Galvin');

INSERT into publisher values ('Mcgraw-Hill', 'Bangalore', 9989076587);
INSERT into publisher values ('Pearson', 'Newdelhi', 9889076565);
INSERT into publisher values ('Random House', 'Hyderabad', 7455679345);
INSERT into publisher values ('Hachette Livre', 'Chennai', 8970862340);
INSERT into publisher values ('Grupo Planeta', 'Bangalore', 7756120238);

INSERT into library_branch values (10,'RR Nagar', 'Bangalore');
INSERT into library_branch values (11,'RNSIT','Bangalore');
INSERT into library_branch values (12,'Rajaji Nagar', 'Bangalore');
INSERT into library_branch values (13,'NITTE','Mangalore');
INSERT into library_branch values (14,'Manipal','Udupi');

INSERT into book_copies values (1, 10, 10);
INSERT into book_copies values (1, 11, 5);
INSERT into book_copies values (2, 12, 2);
INSERT into book_copies values (2, 13, 5);
INSERT into book_copies values (3, 14, 7);


INSERT into card values (101);
INSERT into card values (102);
INSERT into card values (103);
INSERT into card values (104);
INSERT into card values (105);

INSERT into book_lending values (1, 10, 101,'01-Jan-23','01-Jun-23');
```

```
INSERT into book_lending values ( 3, 14, 101,'11-Jan-23','11-Mar-23');
INSERT into book_lending values ( 2, 13, 101,'21-Feb-23','21-Apr-23');
INSERT into book_lending values (2, 12, 101,'15-Mar-23','15-Jul-23');
INSERT into book_lending values (1, 11, 104, '12-Apr-23','12-May-23');
```

QUERIES:

1. Retrieve the book_id and title of author Navathe.

Ans:

```
SELECT b.book_id,b.title
FROM book b, book_authors ba
WHERE b.book_id=ba.book_id and
  ba.author_name='Navathe';
```

O/P:

BOOK_ID	TITLE
1	DBMS
2	ADBMS

2. Retrieve name and address of all publisher along with book_title.

Ans:

```
SELECT p.name,p.address,b.title
FROM book b,publisher p
WHERE p.name=b.publisher_name;
```

O/P:

NAME	ADDRESS	TITLE
Mcgraw-Hill	Bangalore	DBMS
Mcgraw-Hill	Bangalore	ADBMS
Pearson	Newdelhi	CN
Pearson	Newdelhi	OS
Grupo Planeta	Bangalore	CG

3. Retrieve branch name, branch id and number of copies in each branch.

Ans:

```
SELECT l.branch_name,l.branch_id,b.no_of_copies
FROM library_branch l,book_copies b
WHERE l.branch_id=b.branch_id;
```

O/P:

BRANCH_NAME	BRANCH_ID	NO_OF_COPIES
RR Nagar	10	10
RNSIT	11	5
Rajaji Nagar	12	2
NITTE	13	5
Manipal	14	7

4. Retrieve the details of all books in the library like book ID, Book title, name of the publisher, author name, number of copies in each branch.

Ans:

```
SELECT b.book_id, b.title, b.publisher_name, a.author_name, c.no_of_copies, l.branch_id
  FROM book b, book_authors a, book_copies c, library_branch l
  WHERE b.book_id=a.book_id and b.book_id=c.book_id and l.branch_id=c.branch_id;
```

O/P:

BOOK_ID	TITLE	PUBLISHER_NAME	AUTHOR_NAME	NO_OF_COPIES	BRANCH_ID
1	DBMS	Mcgraw-Hill	Navathe	10	10
1	DBMS	Mcgraw-Hill	Navathe	5	11
2	ADBMS	Mcgraw-Hill	Navathe	2	12
2	ADBMS	Mcgraw-Hill	Navathe	5	13
3	CN	Pearson	Tanenbaum	7	14

5. Retrieve the name of author who has written more than two books.

Ans:

```
SELECT author_name
 FROM book_authors
GROUP BY author_name
 HAVING count (*)>1;
```

O/P:

```
AUTHOR_NAME
--------------------
Navathe
```

6. Get the particular of a borrower or who have borrowed more than three books FROM Jan 2023 to Jun 2023.

Ans:

```
SELECT card_no
 FROM book_lending
 WHERE date_out between '01-Jan-23' and '01-Jun-23'
 GROUP BY card_no
  HAVING count(*) >3;
```

O/P:

```
 CARD_NO
 ----------
 101
```

7. Retrieve the publisher name who has published maximum number of books.

Ans:

```
SELECT publisher_name,count(*)
FROM book
GROUP BY publisher_name
HAVING count (*) >= (SELECT max(count(*))
                     FROM book
                     GROUP BY publisher_name);
```

O/P:

PUBLISHER_NAME	COUNT (*)
Mcgraw-Hill	2
Pearson	2

8. Change the number of copy is to 100 for DBMS books.

Ans:

```
UPDATE  book_copies
SET no_of_copies=100
WHERE book_id=3;
```

O/P:

1 row updated.

```
SELECT * FROM book_copies;
```

BOOK_ID	BRANCH_ID	NO_OF_COPIES
1	10	10
1	11	5
2	12	2
2	13	5
3	14	100

9. Delete a book in book table update the content of other table to reflect the data manipulation operation.

Ans:

```
 DELETE FROM book
WHERE book_id=3;
```

O/P:

```
SELECT * FROM book;
```

BOOK_ID	TITLE	PUBLISHER_NAME	PUB_YEAR
1	DBMS	Mcgraw-Hill	2017
2	ADBMS	Mcgraw-Hill	2016
4	CG	Grupo Planeta	2015
5	OS	Pearson	2016

```
SELECT * FROM book_authors;
```

BOOK_ID	AUTHOR_NAME
1	Navathe
2	Navathe
4	Edward Angel
5	Galvin

```
SELECT * FROM book_copies;
```

BOOK_ID	BRANCH_ID	NO_OF_COPIES
1	11	5
2	12	2
2	13	5
1	10	10

SELECT * FROM book_lending;

BOOK_ID	BRANCH_ID	CARD_NO	DATE_OUT	DUE_DATE
1	10	101	01-JAN-23	01-JUN-23
2	13	101	21-FEB-23	21-APR-23
1	11	104	12-APR-23	12-MAY-23
2	12	101	15-MAR-23	15-JUL-23

4. Consider the following schema for Order Database:

SALESMAN (*Salesman_id, Name, City, Commission*)
CUSTOMER1 (*Customer_id, Cust_Name, City, Grade, Salesman_id*)
ORDERS (*Ord_No, Purchase_Amt, Ord_Date, Customer_id, Salesman_id*)

CREATION OF TABLE:

```
CREATE table salesman
(salesman_id number (10) primary key,
name varchar (20),
city varchar (20),
commission number (20));

CREATE table customer
(customer_id number (10) primary key,
cust_name varchar (20),
city varchar (20),
grade varchar(10),
salesman_id number (10) references salesman (salesman_id) on delete set null);

CREATE table orders
(ord_no number (15),
purchase_amt number (15),
ord_date date,
customer_id number(10) references customer(customer_id),
salesman_id number (10) references salesman (salesman_id) on delete cascade);
```

INSERTION OF VALUES:

```
INSERT into salesman values (1000, ‘Rani’,'Bangalore’,20);
INSERT into salesman values (1001, ‘Raji’,'Mangalore’,40);
INSERT into salesman values (1002, ‘Ramesh’,'Magadi’,45);
INSERT into salesman values (1003, ‘Suresh’,'Bangalore’,54);
INSERT into salesman values (1004, ‘Raj’,'Ramanagara’,35);

INSERT into customer values(101,'Harsha','Mangalore','VIP',1000);
INSERT into customer values(102 ,'Jaya','Bangalore','VVIP',1003);
```

```
INSERT into customer values(103,'Sumi','Banglore','VVIP',1001);
INSERT into customer values(104,'Pavi','Ramanagara','Public',1004);
INSERT into customer values (105,'Dimpana','Magadi','VVIP', 1002);
INSERT into customer values(106,'Bhavya','Banglore','Public',1000);

INSERT into orders values(25,55000,'23-Nov-2023',101,1000);
INSERT into orders values(43,47000,'20-Dec-2023',102,1003);
INSERT into orders values(55,25000,'22-Aug-2023',103,1001);
INSERT into orders values(47,72000,'31-Jul-2023',104,1004);
INSERT into orders values(72,50000,'17-Jun-2023',105,1002);
INSERT into orders values(32,12000,'07-Mar-2023',106,1000);
```

QUERY:

1.Display the details of all salesmen who are living in Bangalore city .
Ans:

```
SELECT *
FROM salesman
WHERE city='Bangalore';
```

O/P:

```
SALESMAN_ID NAME                 CITY                 COMMISSION
----------- -------------------- -------------------- ----------
       1000 Rani                 Bangalore                    20
       1003 Suresh               Bangalore                    54
```

2. Display the salesman name, customer name ,purchase amount WHERE purchase amount is more than 50,000 .
Ans:

```
SELECT name,cust_name,purchase_amt
FROM salesman s, customer c,orders o
WHERE s.salesman_id=o.salesman_id and
c.customer_id=o.customer_id and
o.purchase_amt>50000;
```

O/P:

```
NAME                 CUST_NAME            PURCHASE_AMT
-------------------- -------------------- ------------
Rani                 Harsha                      55000
Raj                  Pavi                        72000
```

3.Display the details of all the customer who belongs to VIP grade.
Ans:

```
SELECT *
FROM customer
WHERE grade='VIP';
```

O/P:

```
CUSTOMER_ID CUST_NAME            CITY                 GRADE      SALESMAN_ID
----------- -------------------- -------------------- ---------- -----------
        101 Harsha               Mangalore            VIP              1000
```

4. Display the name of salesman who have highests Commission.

Ans:

```
SELECT name
FROM salesman
WHERE commission = (SELECT max(commission)
FROM salesman);
```

O/P:

```
NAME
-------------
Suresh
```

5. Display the details of order by the customer Harsha.

Ans:

```
SELECT *
FROM orders
WHERE customer_id = (SELECT customer_id
FROM customer
WHERE cust_name='Harsha');
```

O/P:

```
ORD_NO    PURCHASE_AMT    ORD_DATE   CUSTOMER_ID   SALESMAN_ID
 ----------      ------------      ---------      -----------      -----------
      25            55000            23-NOV-23        101               1000
```

6. Count the customer with purchase amount above the average purchase amount.

Ans:

```
SELECT count(*) as no_of_cust
FROM orders
WHERE purchase_amt > (SELECT avg(purchase_amt)
FROM orders);
```

O/P:

```
NO_OF_CUST
    ----------
       4
```

7. Find the name and number of all salesman who have more than one customer.

Ans:

```
SELECT name,salesman_id
FROM salesman s
WHERE 1 < (SELECT count (*)
FROM customer
WHERE salesman_id=s.salesman_id);
```

O/P:

```
NAME         SALESMAN_ID
------------         -----------
Rani                  1000
```

8. List all salesman and indicate those who have and don’t have customer in their cities (use union operator).

Ans:

Ans:

```
(SELECT s.salesman_id,name,cust_name,commission
FROM salesman s, customer
WHERE s.salesman_id=customer.salesman_id) union
(SELECT s.salesman_id, name,'NO MATCH',commission
FROM salesman s,customer c WHERE s.salesman_id=c.salesman_id and s.city!=c.city)
order by 2 desc;
```

O/P:

SALESMAN_ID	NAME	CUST_NAME	COMMISSION
1003	Suresh	Jaya	54
1000	Rani	Bhavya	20
1000	Rani	Harsha	20
1000	Rani	NO MATCH	20
1002	Ramesh	Dimpana	45
1001	Raji	NO MATCH	40
1001	Raji	Sumi	40
1004	Raj	Pavi	35

9. Create a view that finds the salesman who has the customer with the highest order of a day.

Ans:

```
CREATE view sm as
SELECT o.ord_date,s.salesman_id,s.name
FROM salesman s,orders o
WHERE s.salesman_id=o.salesman_id and
o.purchase_amt = (SELECT max (purchase_amt)
FROM orders c
WHERE c.ord_date=o.ord_date);
```

View created.

SELECT * from sm;

ORD_DATE	SALESMAN_ID	NAME
23-NOV-23	1000	Rani
20-DEC-23	1003	Suresh
22-AUG-23	1001	Raji
31-JUL-23	1004	Raj
17-JUN-23	1002	Ramesh
07-MAR-23	1000	Rani

10. Demonstrate the delete operation by removing salesman with ID 1000 and all his order must be deleted.

Ans:

```
DELETE  from salesman
WHERE salesman_id=1000;
```

1 row deleted.

SQL:**SELECT * from salesman;**

SALESMAN_ID	NAME	CITY	COMMISSION
1001	Raji	Mangalore	40
1002	Ramesh	Magadi	45
1003	Suresh	Bangalore	54
1004	Raj	Ramanagara	35

SQL> **select * from customer;**

CUSTOMER_ID	CUST_NAME	CITY	GRADE	SALESMAN_ID
101	Harsha	Mangalore	VIP	
102	Jaya	Bangalore	VVIP	1003
103	Sumi	Banglore	VVIP	1001
104	Pavi	Ramanagara	Public	1004
105	Dimpana	Magadi	VVIP	1002
106	Bhavya	Banglore	Public	

6 rows selected.

SQL> **select * from orders;**

ORD_NO	PURCHASE_AMT	ORD_DATE	CUSTOMER_ID	SALESMAN_ID
43	47000	20-DEC-23	102	1003
55	25000	22-AUG-23	103	1001
47	72000	31-JUL-23	104	1004
72	50000	17-JUN-23	105	1002