- *** PROGRAM 1: INSURANCE DATABASE**

I. Consider the Insurance database given below. The primary keys are underlined and the data types are specified.

PERSON (driver – id #: String, name: String, address: String)
CAR (Regno: String, model: String, year: int)
ACCIDENT (report-number: int, accdate: date, location: String)
OWNS (driver-id #: String, Regno: String)
PARTICIPATED (driver-id#: String, Regno: String, report-number: int, damage-amount: int)

SQL>create table PERSON(d_id varchar(12),name varchar(12),address varchar(33),primary key(d_id));

SQL>create table OWNS(d_id varchar(12) references person(d_id),
regno varchar(12) references car(regno),primary key(regno,d_id));

SQL> insert into PERSON values('a11','abc','bangalore');

SQL> select * from person;

```
DRIVERID   NAME       ADDRESS
---------- ---------- ----------
1          aa         bangalore
2          bb         bangalore
3          cc         mysore
```

SQL> select * from car;

```
REGNO      MODEL      YEAR
---------- ---------- ---------
11         bmw        12-JAN-10
12         benz       12-MAR-08
13         benz       14-JUN-08
```

SQL> select * from owns;

```
DRIVERID   REGNO
---------- ----------
1          11
2          12
3          13
```

SQL> select * from accident;

```
 REPORTNO ACCIDENTD LOCATION
---------- --------- ----------
       111 12-JAN-11 bangalore
       222 12-MAR-11 mysore
       333 12-MAR-11  mysore
       444 12-JAN-08 bangalore
```

SQL> select * from participated;

DRIVERID	REGNO	REPORTNO	DAMAGE
1	11	111	20000
2	12	222	4000
3	13	333	20000

iii) Demonstrate how you : a). Update the damage amount to 25000 for the car with a specific Regno in the accident with report number 12.

```
SQL>  UPDATE PARTICIPATED
      SET damagedamount='25000'
      WHERE (reportno='12' and regno='ka02e1231');
```

b). Add a new accident to the database.

b). SQL>insert into ACCIDENT values('33','01/sep/2002','bangalore');

QUERY 4: Find the total number of people who owned cars that were involved in accidents in 2008.

```
select count(*) from person p,accident ac,participated pa where (p.driverid=pa.driverid) and
(ac.reportno=pa.reportno) and (accidentdate like '%08');

 count(*)
----------
         1
```

QUERY 5: Find the number of accidents in which cars belonging to a specific model were involved.

```
select count(*) from car c,accident ac,participated pa where (c.regno=pa.regno)
 and (ac.reportno=pa.reportno) and c.model='bmw'


COUNT(*)
------
     1
```

PROGRAM 2: ORDER PROCESSING DATABASE

II. Consider the following relations for an Order Processing database application in a company.

CUSTOMER (CUST #: int, cname: String, city: String)
ORDER (order #: int, odate: date, cust #: int, ord-Amt: int)
ITEM (item #: int, unit-price: int)
ORDER-ITEM (order #: int, item #: int, qty: int)
WAREHOUSE (warehouse #: int, city: String)
SHIPMENT (order #: int, warehouse #: int, ship-date: date)

```
CREATE TABLE ORDER_ITEM(ORDERNO number(10) references orders(orderno),
ITEMNO number(10) references item(itemno)  on delete set null ,
QTY number(10));
```

SQL> select * from customers;

CUSTNO	CNAME	CITY
11	shobha	BANGLORE
12	shilpa	MANDYA
13	sameera	MYSORE
14	swetha	DHARWAD
15	shubha	GULBARGA
16	Sunitha	Hubbli

SQL> select * from orders;

ORDERNO	ODATE	CUSTNO	ORD_AMT
21	12-JAN-02	11	1000
22	12-JAN-02	12	2000
23	02-JAN-02	13	3000
24	12-FEB-02	14	4000
25	12-JAN-02	15	5000

SQL> select * from item;

ITEMNO	UNITPRICE
33	100
10	150
3	140
11	100
22	101

SQL> select * from order_item;

ORDERNO	ITEMNO	QTY
21	33	3
22	10	3
23	3	3
24	11	3
25	22	3

SQL> select * from warehouse;

WAREHOUSENO	CITY
1	MYSORE
2	MYSORE
3	MYSORE
4	BANGL
5	BANGL

SQL> select * from shipment;

```
 ORDERNO WAREHOUSENO SHIP_DATE
---------- ---------- ---------
        21          1 04-JAN-02
        22          1 07-JAN-02
        23          1 08-JAN-02
        24          2 09-JAN-02
        25          2 02-JAN-02
```

i) Produce a listing: CUSTNAME, #oforders, AVG_ORDER_AMT, where the middle column is the total numbers of orders by the customer and the last column is the average order amount for that customer.

```
select c.cname,count(*) as total_no_OFORDERS,avg(o.ord_amt)
   from customers c,orders o
where (c.custno=o.custno) group by cname;
```

```
CNAME      TOTAL_NO_OFORDERS AVG(O.ORD_AMT)
---------- ----------------- --------------
sameera                    1           3000
shilpa                     1           2000
shobha                     1           1000
shubha                     1           5000
swetha                     1           4000
```

ii) List the order# for orders that were shipped from all warehouses that the company has in a specific city.

```
select * from orders where orderno in(
      select orderno from shipment where warehouseno in
( select warehouseno from warehouse where city='MYSORE'));
```

```
DERNO ODATE       CUSTNO    ORD_AMT
----- --------- ---------- ----------
   21 12-JAN-02         11       1000
   22 12-JAN-02         12       2000
   23 02-JAN-02         13       3000
   24 12-FEB-02         14       4000
   25 12-JAN-02         15       5000
```

iii) Demonstrate how you delete item# 10 from the ITEM table and make that field *null* in the ORDER_ITEM table.

```
SQL> delete from items where itemno=5001;
```

PROGRAM 3: STUDENT ENROLLMENT DATABASE

III. Consider the following database of student enrollment in courses and books adopted for each course.

STUDENT (regno: String, name: String, major: String, bdate: date)
COURSE (course #: int, cname: String, dept: String)
ENROLL (regno: String, course#: int, sem: int, marks: int)
BOOK_ADOPTION (course #: int, sem: int, book-ISBN: int)
TEXT(book-ISBN:int, book-title:String, publisher:String, author:String)

SQL> select * from student;

REGNO	NAME	MAJOR	BDATE
11	NIKIL	CSE	12-JAN-02
12	DERU	CSE	17-JAN-02
13	TARUN	ISE	13-JAN-02
14	NAKUL	ECE	14-JAN-02
15	SRIRAM	EEE	15-JAN-02

SQL> select * from course;

COURSENO	CNAME	DEPT
31	ADA	CS
32	FAFL	CS
33	GRAPHICS	MECH
34	MATHS	MATHD
35	MP	EC

SQL> select * from enroll;

REGNO	COURSENO	SEM	MARKS
11	31	2	24
12	32	4	25
13	33	6	23
14	34	5	21
15	35	4	25

SQL> select * from book_adoption;

COURSENO	SEM	BOOK_ISBN
31	2	244
32	4	255
33	6	233
34	5	277
35	4	265
31	1	244
32	5	255
33	8	233
31	2	7

SQL> select * from text;

BOOK_ISBN	BOOK_TITLE	PUBLISHER	AUTHOR
244	C	PEARSON	BALAGURU
255	C++	TATA	PADMA
233	DSC	ELITE	PADMA
277	ADA	HIMALAYA	LEVITIN
265	EC	EXCELLENT	MVRAO
7	DBMS Intro	Pearson	Godse

1. Demonstrate how you add a new text book to the database and make this book be adopted by some department

```
insert into text values(7,'DBMS Intro','Pearson','Godse');

insert into book_adoption values(55,2,7);
```

2. **Produce a list of text books (include Course #, Book-ISBN, Book-title) in the alphabetical order for courses offered by the 'CS' department that use more than two books.**

```
select c.courseno,t.book_title,t.book_isbn  from course c,book_adoption b,text t
   where  (c.courseno=b.courseno) and  (t.book_isbn=b.book_isbn) and
b.courseno in (select b.courseno from book_adoption b where c.dept='CS' group by b.courseno
  having count(*)>=2)    order by t.book_title
```

```
 COURSENO BOOK_TITLE  BOOK_ISBN
--------- ---------- ----------
       31 C              244
       31 C              244
       32 C++            255
       32 C++            255
       31 DBMS Intro       7
```

3. **List any department that has *all* its adopted books published by a specific publisher.**

```
  1  select c.dept,c.cname,t.book_title,t.publisher  from text t,book_adoption b,course c
  3     where (c.courseno=b.courseno) and (b.book_isbn=t.book_isbn) and c.dept='CS' and t.publisher='TATA'
```

```
DEPT      CNAME     BOOK_TITLE PUBLISHER
---------- ---------- ---------- ----------
CS        FAFL      C++        TATA
CS        FAFL      C++        TATA
```

PROGRAM 4: BOOK DEALER DATABASE

IV. The following tables are maintained by a book dealer:

AUTHOR(author-id: int, name: String, city: String, country: String)
PUBLISHER(publisher-id: int, name: String, city: String, country: String)
CATALOG(book-id: int, title: String, author-id: int, publisher-id: int, category-id: int, year: int, price: int)
CATEGORY(category-id: int, description: String)
ORDER-DETAILS(order-no: int, book-id: int, quantity: int)

```
select * from author;
```

```
     AID NAME       CITY       COUNTRY
--------- ---------- ---------- ----------
       11 nandagopal banglore   india
       12 yogish     mandya     india
       13 ian        london     england
       14 padmareddy dharwad    india
       15 albert     newyork    usa
```

select * from publisher;

PID	NAME	CITY	COUNTRY
21	sapna	bangalore	india
22	ep	mysore	india
23	pearl	tokyo	japan
24	vvv	sydney	austr
25	tatamg	washington	usa

select * from category;

CID	DESCRIPTIO
31	dbms
32	unix
33	computer
35	vb
36	science

select * from catalog1;

BID	TITLE	AID	PID	CID	YEAR	PRICE
1	c	11	21	31	2001	1000
2	c++	12	22	32	2006	1500
3	c#	13	23	33	2002	1450
4	ansic	14	24	36	2003	1560
5	pointers	15	25	35	2005	1230
6	linux	13	25	35	2003	1830

select * from order_details;

ONO	BID	QUANTITY
41	1	12
42	2	16
43	3	23
44	4	21
45	5	33
46	6	9

1. **Give the details of the authors who have 2 or more books in the catalog and the price of the books is greater than the avg price of the books in the catalog and the year of publication is after 2000.**

```
1  select * from author where aid in(select aid from catalog1 where year>2000 and
2  price > (select avg(price) from catalog1)
3* group by aid having count(aid) >=2)
```

AID	NAME	CITY	COUNTRY
13	ian	london	England

2. **Find the author of the book which has maximum sales.**

```
1  select name from author where aid in(select aid from catalog1 where bid in
3  (select bid from order_details where quantity=(select max(quantity) from order_details)))
```

```
NAME
----------
Albert
```

3. Demonstrate how you increase the price of books published by a specific publisher by 10%.

```
update catalog1 set price=price*1.1  where pid=21;
```

or

```
 select c.title, p.pid,(0.10*price)+price as incprice from catalog1 c, publisher p
   where p.pid=c.pid and p.name='sapna';

TITLE           PID   INCPRICE
---------- ---------- ----------
c               21       1331
```

PROGRAM 5: BANKING ENTERPRISE DATABASE Consider the following database for a banking enterprise.

BRANCH (branch-name: String, branch-city: String, assets: real)
ACCOUNTS (accno: int, branch-name: String, balance: real)
DEPOSITOR (customer-name: String, accno: int)
customer(customer-name:String,customer-street:String,customer-city: String)
LOAN (loan-number: int, branch-name: String, amount: real)
BORROWER (customer-name: String, loan-number: int)

```
SQL> select * from branch;

BNAME      BCITY         ASSETS
---------- ---------- ----------
rtnagar    bang         12000000
yelahanka  bang        100000000
vnagar     mysore       23000000
krnagar    mandya       21000000
hebbal     bang         33000000

SQL> select * from account;

    ACCNO BNAME        BALANCE
---------- ---------- ----------
         1 rtnagar        123450
         2 yelahanka      254310
         3 vnagar         154730
         4 krnagar        564440
         5 hebbal         342110
         6 rtnagar        223450

SQL> select * from customer;

CNAME      CSTREET    CCITY
---------- ---------- ----------
kiran      a1         bang
vijay      b2         bang
barath     d5         mysore
chandru    t4         mandya
dinesh     h9         bang
```

SQL> select * from depositor;

```
CNAME          ACCNO
---------- ----------
kiran               1
vijay               2
barath              3
chandru             4
dinesh              5
kiran               6
```

SQL> select * from loan;

```
   LOANNO BNAME         AMOUNT
---------- ---------- ----------
        21 hebbal         110000
        22 yelahanka      120000
        23 vnagar          14000
        24 krnagar        480000
        25 hebbal         280000
```

SQL> select * from borrower;

```
CNAME         LOANNO
---------- ----------
kiran              21
vijay              22
barath             23
chandru            24
dinesh             25
```

QUERY 3: Find all the customers who have at least two accounts at the *Main* branch.

```
  1  select distinct(cname) from depositor where accno in(select accno from account where
  2* bname='rtnagar') group by cname having count(cname)>=1
```

```
CNAME
----------
Kiran
```

QUERY 4: Find all the customers who have an account at *all* the branches located in a specific city.

```
select distinct(cname) from depositor  where accno in ( select accno from account
where bname in (select bname from branch where bcity='bang'))
```

```
CNAME
----------
dinesh
kiran
vijay
```

QUERY 5: Demonstrate how you delete all account tuples at every branch located in a specific city.

SQL> delete from account where bname in (select bname from branch where bcity='mandya');

1 row deleted.

Select * from account;