DATABASE APPLICATIONS LABORATORY

Subject Code: 10CSL57	**I.A. Marks : 25**
Hours/Week : 03	**Exam Hours: 03**
Total Hours : 42	**Exam Marks: 50**

1. Consider the following relations:
   Student (*snum:* integer, *sname:* string, *major:* string, *level:* string, *age:* integer)
   Class (*name:* string, *meets at:* string, *room:* string, *d:* integer)
   Enrolled (*snum:* integer, *cname:* string)
   Faculty (f*id*: integer, *fname:* string, *deptid:* integer)
   The meaning of these relations is straightforward; for example, Enrolled has one record per student-class pair such that the student is enrolled in the class. Level is a two character code with 4 different values (example: Junior: JR etc)
   Write the following queries in SQL. No duplicates should be printed in any of the answers.
   i. Find the names of all Juniors (level = JR) who are enrolled in a class taught by Prof. Harshith
   ii. Find the names of all classes that either meet in room R128 or have five or more Students enrolled.
   iii. Find the names of all students who are enrolled in two classes that meet at the same time.
   iv. Find the names of faculty members who teach in every room in which some class is taught.
   v. Find the names of faculty members for whom the combined enrollment of the courses that they teach is less than five.

2. The following relations keep track of airline flight information:
   Flights (*no:* integer, *from:* string, *to:* string, *distance:* integer, *Departs:* time, *arrives:* time, *price:* real)
   Aircraft (*aid:* integer, *aname:* string, *cruisingrange:* integer)
   Certified (*eid:* integer, *aid:* integer)
   Employees (*eid:* integer, *ename:* string, *salary:* integer)
   Note that the Employees relation describes pilots and other kinds of employees as well; Every pilot is certified for some aircraft, and only pilots are certified to fly.
   Write each of the following queries in SQL.

   i. Find the names of aircraft such that all pilots certified to operate them have salaries more than Rs.80, 000.
   ii. For each pilot who is certified for more than three aircrafts, find the *eid* and the maximum *cruisingrange* of the aircraft for which she or he is certified.
   iii. Find the names of pilots whose *salary* is less than the price of the cheapest route from Bengaluru to Frankfurt.
   iv. For all aircraft with *cruisingrange* over 1000 Kms, .find the name of the aircraft and the average salary of all pilots certified for this aircraft.
   v. Find the names of pilots certified for some Boeing aircraft.
   vi. Find the *aid*s of all aircraft that can be used on routes from Bengaluru to New Delhi.

3. Consider the following database of student enrollment in courses & 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)
   i. Create the above tables by properly specifying the primary keys and the foreign keys.
   ii. Enter at least five tuples for each relation.

iii. Demonstrate how you add a new text book to the database and make this book be adopted by some department.
iv. 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.
v. List any department that has *all* its adopted books published by a specific publisher.
vi. Generate suitable reports.
vii. Create suitable front end for querying and displaying the results.

4. 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)
   i. Create the above tables by properly specifying the primary keys and the foreign keys.
   ii. Enter at least five tuples for each relation.
   iii. 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 average price of the books in the catalog and the year of publication is after 2000.
   iv. Find the author of the book which has maximum sales.
   v. Demonstrate how you increase the price of books published by a specific publisher by 10%.
   vi. Generate suitable reports.
   vii. Create suitable front end for querying and displaying the results.

5. Consider the following database for a banking enterprise
BRANCH(branch-name:string, branch-city:string, assets:real)
ACCOUNT(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)
   i. Create the above tables by properly specifying the primary keys and the foreign keys
   ii. Enter at least five tuples for each relation
   iii. Find all the customers who have at least two accounts at the *Main* branch.
   iv. Find all the customers who have an account at *all* the branches located in a specific city.
   v. Demonstrate how you delete all account tuples at every branch located in a specific city.
   vi. Generate suitable reports.
   vii. Create suitable front end for querying and displaying the results.

Instructions:

1. The exercises are to be solved in an RDBMS environment like Oracle or DB2.
2. Suitable tuples have to be entered so that queries are executed correctly.
3. Front end may be created using either VB or VAJ or any other similar tool.
4. The student need not create the front end in the examination. The results of the queries may be displayed directly.
5. Relevant queries other than the ones listed along with the exercises may also be asked in the examination.
6. Questions must be asked based on lots.

COLLEGE DATABASE

```
Create table student2(
snum int primary key,
sname varchar(10) not null,
major varchar(10) not null,
lvl varchar(2) not null,
age int not null);

Create table faculty2(
fid int primary key,
fname varchar(15) not null,
dept_id int  not null);

Create table class2(
cname varchar(10) primary key,
meetsat varchar(10) not null,
room varchar(10) not null,
d int references faculty2(fid));

Create table enrolled2 (
snum int references student2(snum),
cname varchar(10) references class2(cname),
primary key(snum,cname));

Describe student;
```

Name	Null?	Type
SNUM	NOT NULL	NUMBER (38)
SNAME	NOT NULL	VARCHAR2 (10)
MAJOR	NOT NULL	VARCHAR2 (10)
LVL	NOT NULL	VARCHAR2 (2)
AGE	NOT NULL	NUMBER (38)

Describe faculty;

Name	Null?	Type
FID	NOT NULL	NUMBER(38)
FNAME	NOT NULL	VARCHAR2(15)
DEPT_ID	NOT NULL	NUMBER(38)

Describe class;

Name	Null?	Type
NAME	NOT NULL	VARCHAR (10)
MEETSAT	NOT NULL	VARCHAR (10)
ROOM	NOT NULL	VARCHAR (10)
FID	NOT NULL	NUMBER(38)

Describe enrolled;

Name	Null?	Type
SNUM	NOT NULL	NUMBER (38)
CNAME	NOT NULL	VARCHAR2 (10)

VALUES FOR TABLES:

Insert into student2 values ('111','john','cs','jr','19');
Insert into student2 values ('222','tarun','cse','sr','22');
Insert into student2 values ('333','mac','civil','jr','21');
Insert into student2 values ('444','ethan','mech','jr','20');
Insert into student2 values ('555','sonia','eee','sr','22');

SELECT * from student2;

SNUM	SNAME	MAJOR	LVL	AGE
111	john	cse	jr	19
222	tarun	cse	sr	22
333	mac	civil	jr	21
444	ethan	mech	jr	20
555	sonia	eee	sr	22

Insert into faculty2 values ('101','asthana','1');
Insert into faculty2 values ('102','sham','2');
Insert into faculty2 values ('103','abhishek','3');
Insert into faculty2 values ('104','vinay','4');
Insert into faculty2 values ('105','shekar','5');
Insert into faculty2 values ('106','harshith','6');

SELECT * from faculty2;

FID	FNAME	DEPT_ID
101	asthana	1
102	sham	2
103	abhishek	3
104	vinay	4
105	shekar	5
106	harshith	6

Insert into class2 values ('se','11am','r128', 106);
Insert into class2 values ('os','11am','r127', 102);
Insert into class2 values ('cn','2pm','r126', 106);
Insert into class2 values ('flat','8am','r125', 102);
Insert into class2 values ('dbms','11am','r127', 106);
Insert into class2 values ('ss','12pm','r128', 106);
Insert into class2 values ('ptw','8am','r128', 102);
Insert into class2 values ('oomd','11am','r127', 106);

SELECT * from class2;

CNAME	MEETSAT	ROOM	D
se	11am	r128	106
os	11am	r127	102
cn	2pm	r126	106
flat	8am	r125	102
dbms	11am	r127	106
ss	12pm	r128	106
oomd	11am	r127	106
ptw	8am	r128	102

```
Insert into enrolled2 values ('111','se');
Insert into enrolled2 values ('222,'dbms');
Insert into enrolled2 values ('111','dbms');
Insert into enrolled2 values ('333','dbms');
Insert into enrolled2 values ('444','dbms');
Insert into enrolled2 values ('555','dbms');
```

SELECT * from enrolled2;

SNUM	CNAME
111	se
333	dbms
444	dbms
555	dbms
111	dbms

i)Find the names of all juniors (level = JR) who are enrolled in a class taught by Prof. Harshith.

```
SELECT distinct sname
FROM student s,class c,enrolled e,faculty f
WHERE   lvl='jr' and
            s.snum=e.snum and
             c.cname=e.cname and
            fid=d and
             fname='harshith';
```

```
SNAME
------
ethan
john
mac
```

ii. Find the names of all classes that either meet in room R128 or have five or more Students enrolled.

```
SELECT cname
FROM class2
WHERE room='r128' or
            cname in (SELECT cname
                        FROM enrolled2
                        GROUP BY cname
                        HAVING count (snum)>=5);
```

```
CNAME
-----
se
ss
ptw
```

iii. Find the names of all students who are enrolled in two classes that meet at the same time.

```
SELECT sname
FROM student2
WHERE snum in (SELECT snum
        FROM enrolled2 e, class2 c
        WHERE   meetsat='11am' and
                c.cname=e.cname
        GROUP BY snum
        HAVING count (snum) =2);
```

```
SNAME
---------
John
```

iv. Find the names of faculty members who teach in every room in which some class is taught.

```
SELECT fname
FROM faculty2 f, class2 c,enrolled2 e
WHERE   fid=d and
          c.cname=e.cname
GROUP BY fname
HAVING count (distinct room) = (SELECT count (distinct room)
                  FROM class2 c,enrolled2 e
                  WHERE c.cname=e.cname);
```

```
Fname
-----
harshith
```

v. Find the names of faculty members for whom the combined enrollment of the courses that they teach is less than five.

```
SELECT fname
 FROM faculty2 f, class2 c
 WHERE   fid=d
```

```
GROUP BY fname
HAVING count (cname)<5;
```

```
FNAME
--------
Sham
```

AIRLINE DATABASE

```
create table aircraft2(
aid int primary key,
aname varchar(10),
cruisingrange int);

create table employees2(
eid int primary key,
ename varchar(10),
salary real);

create table certified2(
eid int,
aid int,
foreign key (eid) references employees2(eid),
foreign key (aid) references aircraft2(aid),
primary key(eid,aid));

create table flights2(
num int primary key ,
frome varchar(10),
too varchar(10),
distance int,
departs int,
arrives int,
price real);

Desc aircraft;
```

Name	Null?	Type
AID	NOT NULL	NUMBER(38)
ANAME		VARCHAR2(10)
CRUISINGRANGE		NUMBER(38)

Desc employees;

Name	Null?	Type
EID	NOT NULL	NUMBER(38)
ENAME		VARCHAR2(10)
SALARY		FLOAT(63)

Desc certified;

Name	Null?	Type
EID	NOT NULL	NUMBER(38)
AID	NOT NULL	NUMBER(38)

Desc flights;

Name	Null?	Type
NUM	NOT NULL	NUMBER(38)
FROME		VARCHAR2(10)
TOO		VARCHAR2(10)
DISTANCE		NUMBER(38)
DEPARTS		NUMBER(38)
ARRIVES		NUMBER(38)
PRICE		FLOAT(63)

VALUES FOR TABLES:

```
Insert into aircraft2 values(50,'kingfisher',20000);
Insert into aircraft2 values(51,'jet',500);
Insert into aircraft2 values(52,'IA',25000);
Insert into aircraft2 values(53,'Air india',30000);
Insert into aircraft2 values(54,'Boeing3',26000);
Insert into aircraft2 values(55,'Boeing5',25000);
```

Select * from aircraft;

```
AID ANAME      CRUISINGRANGE
---- ---------- -------------
  50 kingfisher        20000
  51 jet                500
  52 IA                25000
  53 Air india         30000
  54 boeing3           26000
  55 boeing5           25000
```

AID	ANAME	CRUISINGRANGE
50	kingfisher	20000
51	jet	500
52	IA	25000
53	Air india	30000
54	boeing3	26000
55	boeing5	25000

Insert into employees2 values(200,'manish',100000);
Insert into employees2 values(201,'ramesh',200000);
Insert into employees2 values(202,'ganesh',4000);
Insert into employees2 values(203,'suresh',85000);
Insert into employees2 values(204,'mahesh',80000);

Select * from employees;

```
EID   ENAME       SALARY
---   ----------  ----------
200    manish            100000
201   ramesh       200000
202   ganesh             4000
203    suresh           85000
204   mahesh            80000
```

EID	ENAME	SALARY
200	manish	100000
201	ramesh	200000
202	ganesh	4000

203	suresh	85000
204	mahesh	80000

```
Insert into certified2 values(200,53);
Insert into certified2 values(200,50);
Insert into certified2 values(200,51);
Insert into certified2 values(200,52);
Insert into certified2 values(202,54);
Insert into certified2 values(201,51);
Insert into certified2 values(202,52);
Insert into certified2 values(202,55);
Insert into certified2 values(201,50);
```

```
Select * from certified;
```

```
EID       AID
---- ----------
 200        53
 200        50
 200        51
 200        52
 202        54
 201        51
 202        52
 202        55
 201        50
```

EID	AID
200	53
200	50
200	51
200	52
202	54
201	51
202	52
202	55
201	50

```
Insert into flights2 values(11,'bengaluru','frankfurt',2000,9,11,60000);
Insert into flights2 values(12,'bengaluru','frankfurt',2000,8,10,50000);
```

```
Insert into flights2 values(33,'bengaluru','newdelhi',1000,9,11,5000);
Insert into flights2 values(44,'bengaluru','newdelhi',800,12,2,6000);
Insert into flights2 values(55,'bengaluru','newdelhi',800,12,2,6000);

Select * from flights;
```

```
NUM FROME      TOO        DISTANCE   DEPARTS    ARRIVES      PRICE
---- ---------- ---------- ---------- ---------- ---------- ----------
  11 bengaluru  frankfurt       2000          9         11      60000
  12 bengaluru  frankfurt       2000          8         10      50000
  33 bengaluru  newdelhi         1000          9         11       5000
  44 bengaluru  newdelhi          800         12          2       6000
  55 engaluru  newdelhi          800         12          2       6000
```

NUM	FROME	TOO	DISTANCE	DEPARTS	ARRIVES	PRICE
11	bengaluru	frankfurt	2000	9	11	60000
12	bengaluru	frankfurt	2000	8	10	50000
33	bengaluru	newdelhi	1000	9	11	5000
44	bengaluru	newdelhi	800	12	2	6000
55	engaluru	newdelhi	800	12	2	6000

i) Find the names of aircraft such that all pilots certified to operate them have salaries more than Rs.80, 000.

```
SELECT aname, COUNT (eid)
FROM aircraft2 a, certified2 c
WHERE a.aid=c.aid
GROUP BY aname
INTERSECT
SELECT aname, COUNT (e.eid)
FROM aircraft2 a, certified2 c, employees2 e
WHERE a.aid=c.aid and
        c.eid=e.eid and
        salary>80000
GROUP BY aname;
```

```
ANAME      COUNT(EID)
----------  ----------
Air india           1
jet                 2
kingfisher          2
```

ii) For each pilot who is certified for more than three aircrafts, find the eid and the maximum cruisingrange of the aircraft for which she or he is certified.

```
SELECT eid, max(cruisingrange)
 FROM aircraft2 a, certified2 c
 WHERE  c.aid=a.aid
 GROUP BY eid
 HAVING COUNT(c.aid)>3;
```

```
   EID       MAX(CRUISINGRANGE)
------          ------------------
   200             30000
```

iii) Find the names of pilots whose salary is less than the price of the cheapest route from Bengaluru to Frankfurt.

```
SELECT distinct ename
FROM employees2 e,certified2 c
WHERE c.eid=e.eid and
            salary<(SELECT min(price)
                      FROM flights2
                    WHERE  frome='bengaluru' and
                                too='frankfurt');
```

```
ENAME
----------
ganesh
```

iv) For all aircraft with cruisingrange over 1000 Kms, .find the name of the aircraft and the average salary of all pilots certified for this aircraft.

```
SELECT aname,avg(salary)
FROM aircraft2 a, employees2 e, certified2 c
WHERE c.eid=e.eid and
            a.aid=c.aid and
            cruisingrange>1000
GROUP BY aname;
```

```
ANAME      AVG (SALARY)
---------- -----------
Air india          100000
IA                 52000
boeing3            4000
boeing5            4000
Kingfisher         150000
```

v) Find the names of pilots certified for some Boeing aircraft.

```
SELECT distinct ename
FROM aircraft2 a,certified2 c,employees2 e
WHERE c.eid=e.eid and
a.aid=c.aid and
aname like 'boeing%';
```

```
ENAME
-------
Ganesh
```

vi) Find the aids of all aircraft that can be used on routes from Bengaluru to New Delhi.

```
SELECT aid
FROM aircraft2
WHERE cruisingrange>=(SELECT min(distance)
 FROM flights2
 WHERE frome='bengaluru' and too='newdelhi');
```

```
AID
---
 50
 52
 53
 54
 55
```

STUDENT ENROLLMENT DATABASE

```
Create table Student3 (
Regno varchar(10),
Name varchar(10) not null,
Major varchar(10) not null,
Bdate date,
primary key (Regno));


Create table Course3 (
Course_id integer,
Cname varchar(10) not null,
Dept varchar(10) not null,
primary key (Course_id));

 Create table Enroll3 (
 Regno varchar(10),
 Course_id integer,
 Sem integer not null,
 Marks integer,
 primary key (Regno,Course_id),
 foreign key (Regno) references Student3(Regno),
 foreign key (Course_id) references Course3(Course_id));

Create table Text3(
ISBN integer,
Booktitle varchar(20) not null,
Publisher varchar(20),
Author varchar(15),
primary key (ISBN));

Create table Book_adoption3(
Courseno integer,
Sem integer,
ISBN integer,
primary key (Courseno,Sem),
foreign key (Courseno) references Course3(Course_id),
foreign key (ISBN) references Text3(ISBN));
```

Desc student;

Name	Null?	Type
REGNO	NOT NULL	VARCHAR2 (10)
NAME	NOT NULL	VARCHAR2 (10)
MAJOR	NOT NULL	VARCHAR2 (10)
BDATE	DATE	

Desc course;

Name	Null?	Type
COURSE_ID	NOT NULL	NUMBER(4)
CNAME		VARCHAR2(10)
DEPT		VARCHAR2(5)

Desc Enroll;

Name	Null?	Type
REGNO	NOT NULL	VARCHAR2(5)
COURSE_ID	NOT NULL	NUMBER(4)
SEM		NUMBER(2)
MARKS		NUMBER(3)

Desc Text;

Name	Null?	Type
ISBN	NOT NULL	NUMBER(38)
BOOKTITLE	NOT NULL	VARCHAR2(20)
PUBLISHER		VARCHAR2(20)
AUTHOR		VARCHAR2(15)

Desc book_adoption;

Name	Null?	Type
COURSENO	NOT NULL	NUMBER(38)
SEM	NOT NULL	NUMBER(38)
ISBN		NUMBER(38)

Insert into Student3 values('1BI02CS010','Karan','CSE','02-Jan-1984');
Insert into Student3 values('1BI02EE015','Jack','EEE','15-Apr-1983');
Insert into Student3 values('1BI00CS010','Adi','CSE','02-Jan-1982');
Insert into Student3 values('1BI01EC089','Rahul','ECE','01-Dec-1983');
Insert into Student3 values('1BI01ME075','Sachin','MECH','18-Jul-1983');

Select * from Student3;

REGNO	NAME	MAJOR	BDATE
1BI01ME075	Sachin	MECH	18-JUL-83
1BI02CS010	Karan	CSE	02-JAN-84
1BI02EE015	Jack	EEE	15-APR-83
1BI00CS010	Adi	CSE	02-JAN-82
1BI01EC089	Rahul	ECE	01-DEC-83

REGNO	NAME	MAJOR	BDATE
1BI01ME075	Sachin	MECH	18-JUL-83
1BI02CS010	Karan	CSE	02-JAN-84
1BI02EE015	Jack	EEE	15-APR-83
1BI00CS010	Adi	CSE	02-JAN-82
1BI01EC089	Rahul	ECE	01-DEC-83

COURSE

Insert into course3 values(11,'DSC','CSE');
Insert into course3 values(22,'ADA','CSE');
Insert into course3 values(33,'CN','EC');
Insert into course3 values(44,'TD','MECH');
Insert into course3 values(55,'MP','EC');

Select * from course3;

COURSENO	CNAME	DEPT
11	DSC	CSE
22	ADA	CSE
33	CN	EC
44	TD	MECH
55	MP	EC

COURSENO	CNAME	DEPT
11	DSC	CSE
22	ADA	CSE
33	CN	EC
44	TD	MECH
55	MP	EC

ENROLL

Insert into enroll3 values('1BI02CS010',22,5,72);
Insert into enroll3 values('1BI00CS010',11,3,90);
Insert into enroll3 values('1BI01EC089',33,6,52);
Insert into enroll3 values('1BI01ME075',44,4,85);
Insert into enroll3 values('1BI02EE015',22,5,75);

Select * from enroll3;

REGNO	COURSENO	SEM	MARKS
1BI02CS010	22	5	72
1BI00CS010	11	3	90
1BI01EC089	33	6	52
1BI01ME075	44	4	85
1BI02EE015	22	5	75

REGNO	COURSENO	SEM	MARKS
1BI02CS010	22	5	72
1BI00CS010	11	3	90
1BI01EC089	33	6	52

1BI01ME075	44	4	85
1BI02EE015	22	5	75

TEXT

```
Insert into text3 values(7722,'VB6','Dreamtech','Holzner');
Insert into text3 values(1144,'DS with C','Sapna','Nandagopalan');
Insert into text3 values(4400,'C Programming','TMH','Balaguruswamy');
Insert into text3 values(5566,'Computer Nw','PHI','Tennenbaum');
Insert into text3 values(3388,'MP','PHI','Brey');

Select * from text3;
```

ISBN	BOOKTITLE	PUBLISHER	AUTHOR
7722	VB6	Dreamtech	Holzner
1144	DS with C	Sapna	Nandagopalan
4400	C Programming	TMH	Balaguruswamy
5566	Computer Nw	PHI	Tennenbaum
3388	MP	PHI	Brey

BOOK_ADOPTION

```
Insert into book_adoption3 values(11,3,7722);
Insert into book_adoption3 values(22,4,7722);
Insert into book_adoption3 values(11,5,4400);
Insert into book_adoption3 values(11,8,5566);
Insert into book_adoption3 values(55,4,3388);
Insert into book_adoption3 values(44,4,5566);
Insert into book_adoption3 values(44,7,3388);
```

Select * from book_adoption3;

COURSENO	SEM	ISBN
11	3	7722
22	4	7722
11	5	4400
11	8	5566
55	4	3388
44	4	5566
44	7	3388

COURSENO	SEM	ISBN
11	3	7722
22	4	7722
11	5	4400
11	8	5566
55	4	3388
44	4	5566
44	7	3388

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

Insert into text3 values(1234,'Elec.Circuits','Sapna','Giridhar');

1 row created.

Insert into book_adoption3 values(55,3,1234);

1 row created.

Select * from text3;

ISBN	BOOKTITLE	PUBLISHER	AUTHOR
7722	VB6	Dreamtech	Holzner

1144	DS with C	Sapna	Nandagopalan
4400	C Programming	TMH	Balaguruswamy
5566	Computer Nw	PHI	Tennenbaum
3388	MP	PHI	Brey
1234	Elec.Circuits	Sapna	Giridhar

6 rows selected.

ISBN	BOOKTITLE	PUBLISHER	AUTHOR
7722	VB6	Dreamtech	Holzner
1144	DS with C	Sapna	Nandagopalan
4400	C Programming	TMH	Balaguruswamy
5566	Computer Nw	PHI	Tennenbaum
3388	MP	PHI	Brey
1234	Elec.Circuits	Sapna	Giridhar

Select * from book_adoption3;

COURSENO	SEM	ISBN
11	3	7722
22	4	7722
11	5	4400
11	8	5566
55	4	3388
44	4	5566
44	7	3388
55	3	1234

8 rows selected.

COURSENO	SEM	ISBN
11	3	7722
22	4	7722
11	5	4400
11	8	5566
55	4	3388

44	4	5566
44	7	3388
55	3	1234

iv. Produce a list of text books in alphabetical order for courses offered by CS department that use more than two books.

```
SELECT c.Course_id, b.ISBN, Booktitle
   FROM Course3 c,Book_adoption3 b,Text3 t
  WHERE c.Course_id=b.Courseno and
              t.ISBN=b.ISBN and
c.Dept='CSE' and
c.Course_id in (select courseno
from book_adoption
   GROUP BY Courseno
Having count(*)>=2)
Order by c.Course_id, Booktitle;
```

```
 COURSENO      ISBN    BOOKTITLE
-------------------  ---------- --------------------
       11             4400    C Programming
       11             5566    Computer Nw
       11             7722          VB6
```

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

```
Select distinct C.Dept
    from Course C, Book_adoption A,Text T
  where C.Courseno=A.Courseno and
        A.ISBN=T.ISBN and
   not exists (( select Y.ISBN
              from Course X,Book_Adoption Y
             where X.Courseno=Y.Courseno
                  and X.Dept=C.Dept)
             minus
             (select ISBN
```

```
 from Text
where publisher='PHI'));
```

```
DEPT
----------
MECH
```

BOOK DEALER DATABASE

```
Create table Author4(
 Authorid integer,
Aname varchar(15),
Acity varchar(15),
Acountry varchar(15),
primary key (Authorid));

Create table Publisher4(
 Publisherid integer,
 Pname varchar(15),
 Pcity varchar(15),
 Pcountry varchar(15),
 primary key (Publisherid));

Create table Category4(
Categoryid integer,
Description varchar(20),
primary key (Categoryid));

Create table Catalog4(
Bookid integer,
Title varchar(20),
Authorid integer,
Publisherid integer,
Categoryid integer,
Year integer,
Price integer,
   primary key (Bookid),
   foreign key (Authorid) references Author4(Authorid),
   foreign key (Publisherid) references Publisher4(Publisherid),
   foreign key (Categoryid) references Category4(Categoryid));

Create table Order_details4(
Orderno integer,
Bookid integer,
Quantity integer,
primary key (Orderno,Bookid),
foreign key (Bookid) references Catalog4(Bookid));
```

desc author;

Name	Null?	Type
AUTHORID	NOT NULL	NUMBER(38)
ANAME		VARCHAR2(15)
ACITY		VARCHAR2(15)
ACOUNTRY		VARCHAR2(15)

Desc publisher;

Name	Null?	Type
PUBLISHERID	NOT NULL	NUMBER(38)
PNAME		VARCHAR2(15)
PCITY		VARCHAR2(15)
PCOUNTRY		VARCHAR2(15)

Desc catalog;

Name	Null?	Type
BOOK_ID	NOT NULL	NUMBER(38)
TITLE		VARCHAR2(20)
AUTHOR_ID		NUMBER(38)
PUBLISHER_ID		NUMBER(38)
CATEGORY_ID		NUMBER(38)
YEAR		NUMBER(38)
PRICE		NUMBER(38)

Desc category;

Name	Null?	Type
CATEGORY_ID	NOT NULL	NUMBER(38)
DESCRIPTION		VARCHAR2(20)

Desc order_details;

Name	Null?	Type
ORDER_NO	NOT NULL	NUMBER(38)
BOOK_ID	NOT NULL	NUMBER(38)
QUANTITY		NUMBER(38)

AUTHOR

```
Insert into Author4 values(1000,'Nandagopalan','Bangalore','India');
Insert into Author4 values(2000,'Tony','Haywood','USA');
Insert into Author4 values(3000,'Holzner','New York','USA');
Insert into Author4 values(4000,'Tennenbaum','London','UK');
Insert into Author4 values(5000,'Balaguruswamy','Chennai','India');

 select * from Author4;
```

AUTHORID	ANAME	ACITY	ACITY
1000	Nandagopalan	Bangalore	India
2000	Tony	Haywood	USA
3000	Holzner	New York	USA
4000	Tennenbaum	London	UK
5000	Balaguruswamy	Chennai	India

AUTHORID	ANAME	ACITY	ACITY
1000	Nandagopalan	Bangalore	India
2000	Tony	Haywood	USA
3000	Holzner	New York	USA
4000	Tennenbaum	London	UK
5000	Balaguruswamy	Chennai	India

PUBLISHER

```
Insert into publisher4 values(11,'Wiely','NewDelhi','India');
Insert into publisher4 values(22,'PHI','California','USA');
Insert into publisher4 values(33,'Sapna','Bangalore','India');
Insert into publisher4 values(44,'TMH','NewYork','USA');
Insert into publisher4 values(55,'Wrox','Texas','USA');
```

Select * from publisher4;

PUBLISHERID	PNAME	PCITY	PCOUNTRY
11	Wiely	NewDelhi	India
22	PHI	California	USA
33	Sapna	Bangalore	India
44	TMH	NewYork	USA
55	Wrox	Texas	USA

PUBLISHERID	PNAME	PCITY	PCOUNTRY
11	Wiely	NewDelhi	India
22	PHI	California	USA
33	Sapna	Bangalore	India
44	TMH	NewYork	USA
55	Wrox	Texas	USA

CATEGORY

Insert into category4 values(1,'OS');
Insert into category4 values(2,'Languages');
Insert into category4 values(3,'Hardware');
Insert into category4 values(4,'Algorithms');
Insert into category4 values(5,'Internet');

Select * from category4;

CATEGORYID	DESCRIPTION
1	OS
2	Languages
3	Hardware
4	Algorithms
5	Internet

CATEGORYID	DESCRIPTION
1	OS
2	Languages

3	Hardware
4	Algorithms
5	Internet

CATALOG

```
Insert into catalog4 values(222,'DSC',1000,33,2,2000,32);
Insert into catalog4 values(444,'Networks',4000,44,4,2002,365);
Insert into catalog4 values(555,'VB6',2000,11,2,2000,300);
Insert into catalog4 values(333,'Frontpage2002',4000,44,5,2003,500);
Insert into catalog4 values(111,'ADA',1000,33,4,2001,35);

Select * from catalog4;
```

BOOKID TITLE AUTHORID PUBLISHERID CATEGORYID YEAR PRICE

222 DSC 1000 33 2 2000 32

444 Networks 4000 44 4 2002 365

555 VB6 2000 11 2 2000 300

333 Frontpage2002 4000 44 5 2003 500

111 ADA 1000 33 4 2001 35

BOOKID	TITLE	AUTHORID	PUBLISHERID	CATEGORYID	YEAR	PRICE
222	DSC	1000	33	2	2000	32
444	Networks	4000	44	4	2002	365
555	VB6	2000	11	2	2000	300
333	Frontpage2002	4000	44	5	2003	500

111	ADA	1000	33	4	2001	35

ORDER_DETAILS

```
Insert into order_details4 values(112,222,100);
Insert into order_details4 values(113,333,20);
Insert into order_details4 values(114,555,50);
Insert into order_details4 values(115,222,500);
Insert into order_details4 values(116,444,8);

Select * from order_details4;
```

```
  ORDERNO     BOOKID    QUANTITY
 ---------- ---------- ----------
     112                222          100
     113                333              20
     114            555              50
     115              222              500
     116              444              8
```

ORDERNO	BOOKID	QUANTITY112
112	222	100
113	333	20
114	555	50
115	222	500
116	444	8

iii) 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 average price of the books in the catalog and the year of publication is after 2000.

```
select C.Authorid,a.aname,a.acountry,a.acity
from Catalog4 C,Author4 a
where A.Authorid=C.Authorid
and C.Year>2000
and C.Price>(Select Avg(Price)
                from Catalog4)
```

```
group by C.Authorid,a.aname,a.acountry,a.acity
having count(C.Authorid)>=2;
```

```
AUTHOR_ID NAME                 COUNTRY              CITY
---------- -------------------- -------------------- --------------------
      4000 Tennenbaum           UK                   London
```

(iv) Find the author of the book which has maximum sales.

```
Select a.aname,c.authorid,c.bookid
from author4 a,catalog4 c,order_details4 o
where c.bookid=o.bookid
and c.authorid=a.authorid
and o.quantity=(select max(quantity)
                from order_details4);
```

```
NAME                 AUTHOR_ID    BOOK_ID
-------------------- ---------- ----------
Nandagopal                 1000        222
```

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

```
UPDATE catalog4
 Set price=price+price*0.1
 where publisherid  in  ( SELECT publisherid
                          from publisher4
                          where pname='Sapna');
```

```
2 rows updated.
```

or

```
update catalog
set price=price*1.10
where publisherid=33;
```

```
2 rows updated.
```

5.BANKING ENTERPRISE

```
Create table branch5(
branchname varchar(20),
branchcity varchar(20),
assets real,
primary key(branchname));


Create table account5(
accno int primary key,
branchname varchar(20) references branch5(branchname),
balance real);


Create table customer5(
customername varchar(20),
customerstreet varchar(20),
customercity varchar(20),
primary key(customername));

Create table depositor5(
customername varchar(20),
accno int,
primary key(customername,accno),
foreign key(customername) references customer5(customername),
foreign key(accno) references account5(accno)on delete cascade);

Create table loan5(
loanno int,
branchname varchar(20),
amount real,
primary key(loanno),
foreign key(branchname) references branch5(branchname));


Create table borrower5(
customername varchar(20),
loanno int,
```

```
primary key(customername,loanno),
foreign key(customername) references customer5(customername),
foreign key(loanno) references loan5(loanno));
```

DESC branch;

```
 Name                           Null?            Type
 ------------------------------- ---------------- ----------------------------
 BRANCHNAME              NOT NULL     VARCHAR2(15)
 BRANCHCITY                            VARCHAR2(15)
 ASSETS                                 NUMBER(63)
```

DESC accnt;

```
 Name                           Null?          Type
 ------------------------------- -------------- ---------------
 ACCOUNTNO                NOT NULL NUMBER(38)
 BRANCHNAME                            VARCHAR2(15)
 BALANCE                                NUMBER(63)
```

DESC custmer;

```
 Name                           Null?                Type
 ------------------------------- ---------------- -----------------
 CUSTOMERNAME                NOT NULL VARCHAR2(15)
 CUSTOMERSTREET                       VARCHAR2(15)
 CITY                                  VARCHAR2(15)
```

DESC depositor;

```
 Name                           Null?            Type
 ------------------------------- ---------------- ----------------------------
 CUSTOMERNAME        NOT NULL         VARCHAR2(15)
 ACCOUNTNO            NOT NULL         NUMBER(38)
```

desc loan;

```
 Name                          Null?            Type
 ------------------------------ ---------------- ----------------------------
 LOANNO                NOT NULL         NUMBER(38)
```

```
BRANCHNAME                              VARCHAR2(15)
AMOUNT                                  NUMBER(63)
```

DESC borrower;

```
Name                      Null?          Type
------------------------- ---------- --------------------------------
CUSTOMERNAME              NOT NULL       VARCHAR2(15)
LOANNO                    NOT NULL       NUMBER(38)
```

```
Insert into branch5 values('Jayanagar','Bangalore','15000000');
Insert into branch5 values('Basavanagudi','Bangalore','25000000');
Insert into branch5 values('Noida','NewDelhi','50000000');
Insert into branch5 values('Marinedrive','Mumbai','40000000');
Insert into branch5 values('GreenPark','Newdelhi','30000000');
```

Select * from branch5;

```
BRANCHNAME      BRANCHCITY        ASSETS
--------------- --------------- ----------
Jayanagar               Bangalore        15000000
Basavanagudi    Bangalore        25000000
Noida                   NewDelhi         50000000
Marinedrive     Mumbai           40000000
GreenPark       Newdelhi         30000000
```

BRANCHNAME	BRANCHCITY	ASSETS
ayanagar	Bangalore	15000000
Basavanagudi	Bangalore	25000000
Noida	NewDelhi	50000000
Marinedrive	Mumbai	40000000
GreenPark	NewDelhi	30000000

```
Insert into account5 values('123','Jayanagar','25000');
Insert into account5 values('156','Jayanagar','30000');
Insert into account5 values('456','Basavanagudi','15000');
Insert into account5 values('789','Noida','25000');
Insert into account5 values('478','Marinedrive','48000');
Insert into account5 values('778','GreenPark','60000');
```

Insert into account5 values('189','Basavanagudi','48888');

Select * from account5;

```
ACCOUNTNO      BRANCHNAME          BALANCE
---------          ---------------          ----------
      123                  Jayanagar          25000
      156                   Jayanagar          30000
      456                  Basavanagudi              15000
      789                  Noida              25000
      478                  Marinedrive        48000
      778                  GreenPark          60000
      189                  Basavanagudi              48888
```

ACCOUNTNO	BRANCHNAME	BALANCE
123	Jayanagar	25000
156	Jayanagar	30000
456	Basavanagudi	15000
789	Noida	25000
478	Marinedrive	48000
778	GreenPark	60000
189	Basavanagudi	48888

Insert into customer5 values('Ramu','Jayanagar','Bangalore');
Insert into customer5 values('Kumar','Basavanagudi','Bangalore');
Insert into customer5 values('John','Noida','Newdelhi');
Insert into customer5 values('Mike','Marinedrive','Mumbai');
Insert into customer5 values('Sachin','GreenPark','NewDelhi');

Select * from customer5;

```
CUSTOMERNAME    CUSTOMERSTREET       CITY
--------------- --------------- -------------
Ramu                    Jayanagar               Bangalore
Kumar                           Basavanagudi    Bangalore
John                    Noida                   Newdelhi
Mike                    Marinedrive         Mumbai
Sachin                          GreenPark       NewDelhi
```

CUSTOMERNAME	CUSTOMERSTREET	CITY
Ramu	Jayanagar	Bangalore
Kumar	Basavanagudi	Bangalore
John	Noida	Newdelhi
Mike	Marinedrive	Mumbai
Sachin	GreenPark	Newdelhi

```
Insert into depositor5 values('Ramu',123);
Insert into depositor5  values('Ramu',156);
Insert into depositor5  values('Ramu',189);
Insert into depositor5 values('Kumar',456);
Insert into depositor5  values('John',789);
Insert into depositor5 values('Mike',478);
Insert into depositor5  values('Sachin',778);
```

```
Select * from depositor5;
```

```
CUSTOMERNAME    ACCOUNTNO
--------------- ----------
Ramu                 123
Ramu                 156
Ramu                 189
Kumar                456
John                 789
Mike                 478
Sachin               778
```

CUSTOMERNAME	ACCOUNTNO
Ramu	123
Ramu	156
Ramu	189
Kumar	456
John	789
Mike	478
Sachin	778

```
Insert into loan5 values('1111','Jayanagar','250000');
```

```
Insert into loan5 values('2222','Basavanagudi','350000');
Insert into loan5 values('3333','Noida','150000');
Insert into loan5 values('4444','Marinedrive','1500000');
Insert into loan5 values('5555','GreenPark','7500000');

Select * from loan5;

LOANNO     BRANCHNAME     AMOUNT
-------          ---------------          ----------
  1111        Jayanagar          250000
  2222              Basavanagudi              350000
  3333        Noida              150000
  4444        Marinedrive        1500000
  5555        GreenPark          7500000
```

LOANNO	BRANCHNAME	AMOUNT
1111	Jayanagar	250000
2222	Basavanagudi	350000
3333	Noida	150000
4444	Marinedrive	1500000
5555	GreenPark	7500000

BORROWER

```
Insert into borrower5 values('Ramu',1111);
Insert into borrower5 values('Kumar',2222);
Insert into borrower5 values('John',3333);
Insert into borrower5 values('Mike',4444);
Insert into borrower5 values('Sachin',5555);

Select * from borrower5;

CUSTOMERNAME       LOANNO
---------------- ----------
Ramu                    1111
Kumar                    2222
John                    3333
Mike                    4444
Sachin                  5555
```

CUSTOMERNAME	LOANNO
Ramu	1111
Kumar	2222
John	3333
Mike	4444
Sachin	5555

(iii) Find all the customers who have at least two accounts at the main branch.

```
Select  d.customername
from depositor5  d , branch5 b, account5 a
where b.branchname='Jayanagar'
and b.branchname = a.branchname
and  a.accno = d.accno
 group by d.customername
 having count(*)>1;
```

```
CUSTOMERNAME
---------------
Ramu
```

(iv)Find all the customers who have an account at all the branches located in a specific city.

```
Select d.customername
 from account5 a,branch5 b,depositor5 d
 where a.accno=d.accno
and b.branchname = a.branchname
 and  b.branchcity='Bangalore'
  group by d.customername
  having count(unique(b.branchname)) = (select count(b.branchname)
                                         from branch5 b
                                        where b.branchcity='Bangalore');
```

```
CUSTOMERNAME
---------------
Ramu
```

```
Delete from account5
where branchname IN (select b.branchname
                     from branch5 b
                     where b.branchcity = 'Bangalore');
```

4 rows deleted.