SRINIVAS UNIVERSITY

INSTITUTE OF ENGINEERING AND TECHNOLOGY

MUKKA, MANGALURU

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

LAB MANUA L

FOR

DATABASE MANAGEMENT SYSTEMS LABORATORY

SUBJECT CODE: 23SCS042

COMPILED BY

Dr. J. VENKATA KRISHNA
Assoc. Prof.

2024-25

COURSE OBJECTIVES:

This course will enable students to

- Foundation knowledge into database concepts, technology and practice to groom stud into well-informed database application developers.
- Strong practice in SQL programming through a variety of database problems.
- Develop database applications using front-end tools and back-end DBMS.

COURSE OUTCOMES:

The students should be able to:

- Create, Update and query on the database.
- Demonstrate the working of different concepts of DBMS
- Implement, analyze and evaluate the project developed for an application.

GUIDELINES TO STUDENTS

1. Equipment in the lab for the use of student community. Students need to maintain proper decorum in the computer lab. Students must use the equipment with care Any damage is caused is punishable.
2. Students are instructed to come to lab in formal dresses only.
3. Students are supposed to occupy the systems allotted to them and are not supposed to talk or make noise in the lab.
4. Students are required to carry their observation book and lab records with completed exercises while entering the lab.
5. Lab records need to be submitted every week. 6. Students are not supposed to use pen drives in the lab.

Do's

1. Come with completed observation and record.
2. Wear ID card before entering into the lab.
3. Read and understand how to carry out an activity thoroughly before coming to laboratory.
4. Report any broken plugs or exposed electrical wires to your lecturer/laboratory technician immediately.
5. Write in time, out time and system details in the login register.

Don'ts

1. Do not eat or drink in the laboratory.
2. Do not operate mobile phones in the lab.
3. Do not change system settings.
4. Do not disturb your neighboring students. They may be busy in completing tasks.
5. Do not remove anything from the computer laboratory without permission.
6. Do not use pen drives.

Lab Experiments

EXPERIMENTS NO.1:LIBRARY Database

A. Consider the following schema for a LibraryDatabase:

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)

Write SQL queries to

1. Retrieve details of all books in the library – id, title, name of publisher, authors,number of copies in each branch,etc.
2. Get the particulars of borrowers who have borrowed more than 3 books, but from Jan 2017 to Jun2017
3. Delete a book in BOOK table. Update the contents of other tables to reflect this data manipulation operation.
4. Partition the BOOK table based on year of publication. Demonstrate its working with a simplequery.
5. Create a view of all books and its number of copies that are currently available in the Library.

Creating tables using CREATE TABLE command.

```
CREATE TABLE PUBLISHER(NAME VARCHAR(20) PRIMARY KEY, ADDRESS VARCHAR(20),PHONE
VARCHAR(10));

CREATE TABLE LIBRARY_BRANCH(BRANCH_ID INT PRIMARY KEY,BRANCH_NAME
 VARCHAR(20),ADDRESS VARCHAR(20));

CREATE TABLE BOOK(BOOK_ID INT PRIMARY KEY,TITLE VARCHAR(20), PUBLISHER_NAME
 VARCHAR(20) REFERENCES PUBLISHER(NAME) ON DELETE CASCADE, PUB_YEAR VARCHAR(20));

CREATE TABLE BOOK_AUTHORS(BOOK_ID INT REFERENCES BOOK(BOOK_ID) ON      DELETE
CASCADE,AUTHOR_NAME VARCHAR(20),PRIMARY KEY (BOOK_ID,AUTHOR_NAME));

CREATE TABLE BOOK_COPIES(BOOK_ID INT REFERENCES BOOK(BOOK_ID) ON DELETE
CASCADE,BRANCH_ID INT REFERENCES LIBRARY_BRANCH(BRANCH_ID)ON DELETE CASCADE,
NO_OF_COPIES INT, PRIMARY KEY(BOOK_ID,BRANCH_ID));

CREATE TABLE BOOK_LENDING(BOOK_ID INT REFERENCES BOOK(BOOK_ID) ON DELETE
CASCADE,BRANCH_ID INT REFERENCES LIBRARY_BRANCH(BRANCH_ID)ON DELETE
CASCADE,CARD_NO INT, DATE_OUT DATE, DUE_DATE DATE,PRIMARY
KEY(BOOK_ID,BRANCH_ID,CARD_NO));
```

Inserting Values to the tables using INSERT INTO command.

```
INSERT INTO PUBLISHER VALUES ('Pearson','BANGALORE', 9989076587);
INSERT INTO PUBLISHER VALUES ('SBC','MANGALORE', 9971264715);
INSERT INTO PUBLISHER VALUES ('CELLO','BANGALORE', 9949076558);
INSERT INTO PUBLISHER VALUES ('AB+','MANGALORE', 8989076555);
INSERT INTO PUBLISHER VALUES ('DC','BANGALORE', 8855779621);


INSERT INTO BOOK VALUES (101,'DBMS', 'Pearson',2016);
INSERT INTO BOOK VALUES (102,'VISUAL STUDIO', 'Pearson',2016);
INSERT INTO BOOK VALUES (103,'NET', 'SBC',2016);
INSERT INTO BOOK VALUES (104,'ADE', 'SBC',2016);
INSERT INTO BOOK VALUES (105,'JAVA', 'SBC',2013);

INSERT INTO BOOK_AUTHORS VALUES(101 ,'SHAMAKANTH');
INSERT INTO BOOK_AUTHORS VALUES(102 ,'RAMEZ');
INSERT INTO BOOK_AUTHORS VALUES(103 ,'JOHN');
INSERT INTO BOOK_AUTHORS VALUES(104 ,'GENRKE');
INSERT INTO BOOK_AUTHORS VALUES(105 ,'JOHN');

INSERT INTO LIBRARY_BRANCH VALUES (101,'PEARSON','BANGALORE');
INSERT INTO LIBRARY_BRANCH VALUES (102,'PEARSON','BANGALORE');
INSERT INTO LIBRARY_BRANCH VALUES (103,'SBC','MANGALORE');
INSERT INTO LIBRARY_BRANCH VALUES (104,'SBC','MANGALORE');
INSERT INTO LIBRARY_BRANCH VALUES (105,'SBC','MANGALORE');

INSERT INTO BOOK_COPIES VALUES(101, 101,200);
INSERT INTO BOOK_COPIES VALUES(102, 102,400);
INSERT INTO BOOK_COPIES VALUES(103, 103,500);
INSERT INTO BOOK_COPIES VALUES(104, 104,600);
INSERT INTO BOOK_COPIES VALUES(105, 105,700);
INSERT INTO BOOK_COPIES VALUES(101, 105,400);
INSERT INTO BOOK_COPIES VALUES(101, 104,500);


INSERT INTO BOOK_LENDING VALUES(101,101,02,'2017-08-17','2017-08-25');
INSERT INTO BOOK_LENDING VALUES(102,102,09,'2017-08-14','2017-08-22');
INSERT INTO BOOK_LENDING VALUES(103,103,05,'2017-08-18','2017-08-27');
INSERT INTO BOOK_LENDING VALUES(104,104,06,'2017-08-01','2017-08-09');
INSERT INTO BOOK_LENDING VALUES(105,105,01,'2017-08-15','2017-08-24');
INSERT INTO BOOK_LENDING VALUES(101,105,02,'2017-01-16','2017-02-15');
INSERT INTO BOOK_LENDING VALUES(101,103,02,'2017-02-16','2017-02-25');
INSERT INTO BOOK_LENDING VALUES(103,105,02,'2017-02-16','2017-02-25');
INSERT INTO BOOK_LENDING VALUES(104,105,01,'2017-04-16','2017-06-25');
INSERT INTO BOOK_LENDING VALUES(104,101,02,'2017-01-16','2017-02-25');
INSERT INTO BOOK_LENDING VALUES(102,105,02,'2017-01-01','2017-02-25');
```

Query 1:

SELECT B.Book_id,Title,P.Name,Author_Name,Branch_id,No_of_copies FROM Book B, Book_Authors BA,Publisher P,Book_Copies BC WHERE B.Book_id=BA.Book_id AND B.Publisher_Name=P.Name AND B.Book_id=BC.Book_id;

Output:

Book	Title	Name	Author Name	Branch Id	No Of Copies
101	Fundamentals of DBMS	Pearson	SHAMAKANTH	101	200
102	Microsoft Visual Studio	Pearson	RAMEZ	102	400
103	.NET	SBC	JOHN	103	500
104	ADE	SBC	GENRKE	104	600
105	JAVA	SB(	JOHN	105	700
101	Fundamentals of DBMS	Pea	SHAMAKANTH	105	400
101	Fundamentals of DBMS	Pearson	SHAMAKANTH	104	500

Query 2:

SELECT Card_No,COUNT(*) FROM BOOK_LENDING WHERE Date_out BETWEEN '2017-01-01' AND '2017-06-30' GROUP BY Card_No HAVING COUNT(*)>3;

Output:

Card_no	No_of_Copies
2	5

Query 3:

DELETE FROM BOOK WHERE Book_id='101';

Output:

1 row deleted

Query 4:
CREATE VIEW V_PUBLICATION AS SELECT PUB_YEAR FROM BOOK;

select * from V_PUBLICATION;

Output:
pub_year
2016
2016
2016
2013

Query 5:
CREATE VIEW TOTAL_BOOK_COPIES(Book_id,Total_Books) AS Select Book_id, SUM(No_of_copies) FROM BOOK_COPIES GROUP BY Book_id;

Select * FROM TOTAL_BOOK_COPIES;

Output:

Book_id	**Total_Books**
102	400
101	1100
107	700
104	600
103	500

EXPERIMENTS NO.2:ORDER Database

B. Consider the following schema for OrderDatabase:
SALESMAN (*Salesman_id, Name, City, Commission*)
CUSTOMER (*Customer_id, Cust_Name, City, Grade,Salesman_id*)
ORDERS (*Ord_No, Purchase_Amt, Ord_Date, Customer_id, Salesman_id*)

Write SQL queries to

1. Count the customers with grades above Bangalore's average.
2. Find the name and numbers of all salesmen who had more than one customer.
3. List all salesmen and indicate those who have and don't have customers in their cities (Use UNIONoperation.)
4. Create a view that finds the salesman who has the customer with the highest order of a day.
5. Demonstrate the DELETE operation by removing salesman with id 1000. All his orders must also be deleted.

Creating the tables

```
CREATE TABLE SALESMAN (SALESMAN_ID INTEGER PRIMARY KEY,
NAME VARCHAR (20),CITY VARCHAR (30),COMMISSION VARCHAR (20));


CREATE TABLE CUSTOMER (CUSTOMER_ID INTEGER PRIMARY KEY,
CUST_NAME VARCHAR (20),CITY VARCHAR (30), GRADE INT, SALESMAN_ID
 INTEGER,FOREIGN KEY(SALESMAN_ID) REFERENCES SALESMAN (SALESMAN_ID) ON
 DELETE SET NULL);


CREATE TABLE ORDERS (ORD_NO INTEGER PRIMARY KEY, PURCHASE_AMT
INTEGER,ORD_DATE DATE, CUSTOMER_ID INTEGER, SALESMAN_ID INTEGER,
FOREIGN KEY(SALESMAN_ID)REFERENCES SALESMAN (SALESMAN_ID)
ON DELETE CASCADE,FOREIGN KEY(CUSTOMER_ID) REFERENCES
CUSTOMER(CUSTOMER_ID));
```

Inserting the values

```
INSERT INTO SALESMAN VALUES (1000, 'JOHN','BANGALORE',2000);
INSERT INTO SALESMAN VALUES (1001, 'RAMEZ','BANGALORE',1000);
INSERT INTO SALESMAN VALUES (1002, 'GENRKE','MANGALORE',800);
INSERT INTO SALESMAN VALUES (1003, 'GORGE','MANGALORE',2800);
INSERT INTO SALESMAN VALUES (1004, 'RAMESH','BANGALORE',2890);
INSERT INTO SALESMAN VALUES (1005, 'SUNIL','MANGALORE',3890);
```

```
INSERT INTO CUSTOMER VALUES (2000, 'SUNIL','MANGALORE', 1, 1000);
INSERT INTO CUSTOMER VALUES (2001, 'ANIL','BANGALORE', 1, 1000);
INSERT INTO CUSTOMER VALUES (2002, 'JOHN','BANGALORE', 2, 1005);
INSERT INTO CUSTOMER VALUES (2003, 'JOHN','BANGALORE', 2, 1000);
INSERT INTO CUSTOMER VALUES (2004, 'SUNIL','MANGALORE', 3, 1005);
INSERT INTO CUSTOMER VALUES (2005, 'SUNIL','BANGALORE', 1, 1002);
INSERT INTO CUSTOMER VALUES (2006, 'SUNIL','BANGALORE', 1, 1003);
INSERT INTO CUSTOMER VALUES (2007, 'RAHUL','CHENAI', 2, 1003);

INSERT INTO ORDERS VALUES(101, 7000, '2017-08-21, 2002, 1005);
INSERT INTO ORDERS VALUES(102, 3000, '2017-08-21, 2001, 1000);
INSERT INTO ORDERS VALUES(105, 8000, '2017-08-21, 2000, 1000);
INSERT INTO ORDERS VALUES(108, 9000, '2017-08-21, 2000, 1000);
INSERT INTO ORDERS VALUES(111, 5000, '2017-08-29, 2000, 1000);
INSERT INTO ORDERS VALUES(121, 5900, '2017-08-29, 2003, 1000);
INSERT INTO ORDERS VALUES(221, 5970, '2017-08-29, 2004, 1005);
INSERT INTO ORDERS VALUES(291, 5970, '2017-08-29, 2005, 1002);
INSERT INTO ORDERS VALUES(991, 5970, '2017-08-29, 2006, 1003);
```

Query 1:

```
SELECT COUNT(Customer_id) FROM CUSTOMER WHERE GRADE>(SELECT
AVG(GRADE) FROM CUSTOMER WHERE CITY='BANGALORE');
```

Output:

```
COUNT(Customer_id)
        4
```

Query 2:

```
SELECT S.Salesman_id,S.Name FROM SALESMAN S,CUSTOMER C WHERE
S.Salesman_id=C.Salesman_id GROUP BY S.Salesman_id,S.Name HAVING
COUNT(Customer_id)>1;
```

Output:

Salesman_Id	Name
1003	GORGE
1000	JOHN
1005	SUNIL

Query 3:
SELECT S.Salesman_id FROM SALESMAN S,CUSTOMER C WHERE S.Salesman_id=C.Salesman_id AND S.City!=C.City UNION SELECT S.Salesman_id FROM SALESMAN S,CUSTOMER C WHERE S.Salesman_id=C.Salesman_id AND S.City=C.City;

Output:
Salesman_Id 1000
1002
1003
1005

Query 4:
create view SALESMAN_WITH_MAX_ORDER as select B.ORD_DATE, A.SALESMAN_ID, A.NAME
from salesman a, orders b where a.salesman_id=b.salesman_id and b.purchase_amt=(select max(purchase_amt) from orders c where c.ord_date=b.ord_date);

select * from SALESMAN_WITH_MAX_ORDER;

Output:

Salesman_Id	Name	City	Commision
1000	JOHN	BANGALORE	2000

Query 5:
DELETE FROM SALESMAN WHERE Salesman_id=1000;

Output:
1 Row Deleted.

EXPERIMENTS NO.3: MOVIE Database

C. Consider the following schema for Movie Database:

ACTOR (Act_id, Act_Name, Act_Gender)
DIRECTOR (Dir_id, Dir_Name, Dir_Phone)
MOVIES (Mov_id, Mov_Title, Mov_Year, Mov_Lang, Dir_id)
MOVIE_CAST (Act_id, Mov_id, Role)
RATING (Mov_id, Rev_Stars)

Write SQL queries to

1. List the titles of all movies directed by 'Hitchcock'.
2. Find the movie names where one or more actors acted in two or more movies.
3. List all actors who acted in a movie before 2000 and also in a movie after 2015 (use JOIN operation).
4. Find the title of movies and number of stars for each movie that has at least one rating and find the highest number of stars that movie received. Sort the result by movie title.
5. Update rating of all movies directed by 'Steven Spielberg' to 5.

Creating the tables

```
CREATE TABLE ACTOR( ACT_ID INT PRIMARY KEY, ACT_NAME VARCHAR(20), ACT_GENDER CHAR);

CREATE TABLE DIRECTOR( DIR_ID INT PRIMARY KEY, DIR_NAME VARCHAR(20), DIR_PHONE VARCHAR(10));

CREATE TABLE MOVIES( MOV_ID INT PRIMARY KEY, MOV_TITLE VARCHAR(30), MOV_YEAR VARCHAR(4), MOV_LANG VARCHAR(20), DIR_ID INT, FOREIGN KEY(DIR_ID) REFERENCES DIRECTOR(DIR_ID));

CREATE TABLE MOVIE_CAST( ACT_ID INT, MOV_ID INT,ROLE VARCHAR(20), PRIMARY KEY(ACT_ID,MOV_ID),
FOREIGN KEY(ACT_ID) REFERENCES ACTOR(ACT_ID),FOREIGN KEY(MOV_ID) REFERENCES MOVIES(MOV_ID));

CREATE TABLE RATING( MOV_ID INT, REV_STARS INT, PRIMARY KEY(MOV_ID), FOREIGN KEY(MOV_ID) REFERENCES MOVIES(MOV_ID));
```

Inserting the values

```
INSERT INTO ACTOR VALUES(1,'GANESH','M');
INSERT INTO ACTOR VALUES(2,'SUDEEP','M');
INSERT INTO ACTOR VALUES(3,'RADHIKA','F');
INSERT INTO ACTOR VALUES(4, 'RAKSHITHA','F');
INSERT INTO ACTOR VALUES(5,'RAVICHANDRAN','M');
INSERT INTO ACTOR VALUES(6,'LAKSHMI','F');

INSERT INTO DIRECTOR VALUES(101 ,'DORAI BHAGAVAN',9408757465);
INSERT INTO DIRECTOR VALUES(102,'YOGARAJ BHAT',9408432566);
INSERT INTO DIRECTOR VALUES(103,'SUDEEP',9987657465);
INSERT INTO DIRECTOR VALUES(104,'RAVICHANDRAN',9888757465);
INSERT INTO DIRECTOR VALUES(105,'DWARAKEESH',9484567465);
INSERT INTO DIRECTOR VALUES(106,'HITCHCOCK',9574635243);
INSERT INTO DIRECTOR VALUES(107,'STEVEN SPIELBERG',9384756452);

INSERT INTO MOVIES VALUES(11,'MUNGARU MALE',2008,'KANNADA',102);
INSERT INTO MOVIES VALUES(12,'KEMPEGOUDA ',2010,'KANNADA', 103);
INSERT INTO MOVIES VALUES(13,'PREMALOKA', 1990,'KANNADA', 104);
 INSERT INTO MOVIES VALUES(14,'APTA MITRA',2010,,'KANNADA', 105);
INSERT INTO MOVIES VALUES(15,'APTA RAKSHAKA',2012,'KANNADA', 105);
INSERT INTO MOVIES VALUES(16,'RANADHEERA', 1992 'KANNADA', 104);
INSERT INTO MOVIES VALUES(17,'GANESHANA MADUVE', 1990,'KANNADA',101);
INSERT INTO MOVIES VALUES(18,'THE LAST CHANCE', 2015,'ENGLISH',106);
INSERT INTO MOVIES VALUES(19,'THE WAR' ,2016,'ENGLISH', 106);
INSERT INTO MOVIES VALUES(20,'MURDER', 2016,'ENGLISH', 106);
INSERT INTO MOVIES VALUES(21, 'THE BEAUTIFUL BIRD' ,'ENGLISH', 107);

INSERT INTO MOVIE_CAST VALUES(1,11,'HERO');
INSERT INTO MOVIE_CAST VALUES(2 ,12,' HERO');
INSERT INTO MOVIE_CAST VALUES(3 ,14,' HERO');
INSERT INTO MOVIE_CAST VALUES(5 ,16,' HERO');
INSERT INTO MOVIE_CAST VALUES(6 ,17,' HERO');
INSERT INTO MOVIE_CAST VALUES(1 ,17,' HERO');
INSERT INTO MOVIE_CAST VALUES(1 ,18,' HERO');
INSERT INTO MOVIE_CAST VALUES(1 ,19,' SUBORDINATE');
INSERT INTO MOVIE_CAST VALUES(2 ,20,' HERO');
INSERT INTO MOVIE_CAST VALUES(2 ,21,' HERO');

NSERT INTO RATING VALUES(11,5);
INSERT INTO RATING VALUES(12,4);
INSERT INTO RATING VALUES(13,5);
INSERT INTO RATING VALUES(14,4);
INSERT INTO RATING VALUES(15,5);
INSERT INTO RATING VALUES(16,5);
```

```
INSERT INTO RATING VALUES(17,5);
INSERT INTO RATING VALUES(18,3);
INSERT INTO RATING VALUES(19,3);
INSERT INTO RATING VALUES(20,1);
INSERT INTO RATING VALUES(21,4);
```

Query 1:

```
select MOV_TITLE from MOVIES where DIR_ID=(select DIR_ID from director where
DIR_NAME = 'HITCHCOCK');
```

Output:

```
MOV_TITLE
THE LAST CHANCE
THE WAR MURDER
```

Query 2:

```
select M.MOV_ID,MOV_TITLE from MOVIES M,MOVIE_CAST MC where
M.MOV_ID=MC.MOV_ID and ACT_ID in ( select ACT_ID from MOVIE_CASt
group by ACT_ID having count(*)>=2) order by M.MOV_ID;
```

Output:

MOV_ID	MOV_TITLE
11	MUNGARU MALE
12	RENIPEGOUDA
17	GANESHANA MACLUVC
18	THE LAST CHANCE
19	THE WAR
20	MURDER
21	THE BEAUTIFUL BIRD

Query 3:

select A.ACT_ID,ACT_NAME from ACTOR A, MOVIES M, MOVIE_CAST MC where A.ACT_ID= MC.ACT_ID and M.MOV_ID =MC.MOV_ID and MOV_YEAR> 2015 INTERSECT select A.ACT_ID,ACT_NAMe from ACTOR A, MOVIES M, MOVIE_CAST MC where A.ACT_ID =MC.ACT_ID and M.MOV_ID =MC.MOV_ID and MOV_YEAR< 2000;

Output:

ACT_ID	ACT_NAME
1	ANANTHNAG

Query 4:

select MOV_TITLE,REV_STARS from MOVIES M, RATING R where M.MOV_ID= R.MOV_ID and REV_STARS in (select max(REV_STARS) from rating) order by MOV_TITLE;

Output:

MOV_TITLE	REV_STARS
APTA RAKSHAKA	5
GANESHANA MADUVE	5
MUNGARU MALE	5
PREMALOKA	5
RANADHEERA	5

Query 5:

update rating set REV_STARS= 5 where MOV_ID IN (select MOV_ID from MOVIES M, DIRECTOR D where M.DIR_ID =D.DIR_ID and DIR_NAME='STEVEN SPIELBERG');

Output:

1 row updated.

EXPERIMENTS NO.4: STUDENT Database

D. Consider the following schema for the College Database:

STUDENT(USN, SName, Address, Phone, Gender)
SEMSEC(SSID, Sem, Sec)
CLASS(USN, SSID)
SUBJECT(Subcode, Title, Sem, Credits)
IAMARKS(USN, Subcode, SSID, Test1, Test2, Test3, FinalIA)

Write SQL queries to

1. List all the student details studying in fourth semester 'C-section.
2. Compute the total number of male and female students in each semester and in each section.
3. Create a view of Test1 marks of student USN '1BI15CS101' in all subjects.
4. Calculate the Final IA (average of best two test marks) and update the corresponding table for all students.
5. Categorize students based on the following criterion:
   If Final IA = 17 to 20 then CAT ='Outstanding'

If Final IA = 12 to 16 then CAT = 'Average' If Final IA< 12 then CAT = 'Weak'

Give these details only for 8th semester A, B, and C section students.

Creating the tables

```
CREATE TABLE STUDENT(USN CHAR( 10) PRIMARY KEY, SNAME VARCHAR(20), ADDRESS VARCHAR(20), PHONE VARCHAR( 10), GENDER CHAR);

CREATE TABLE SEMSEC(SSID INT PRIMARY KEY, SEM INT, SEC CHAR);

CREATE TABLE CLASS(USN CHAR(10) PRIMARY KEY, SSID INT REFERENCES SEMSEC(SSID), FOREIGN KEY(USN) REFERENCES STUDENT(USN));

CREATE TABLE SUBJECT(SUBCODE VARCHAR(8) PRIMARY KEY, TITLE VARCHAR(20), SEM INT, CREDITS INT);

CREATE TABLE IAMARKS(USN CHAR( 10) REFERENCES STUDENT(USN), SUBCODE VARCHAR(8) REFERENCES SUBJECT(SUBCODE), SSID INT REFERENCES SEMSEC(SSID), TEST1 INT, TEST2 INT, TEST3 INT, FINALIA INT, PRIMARY KEY(USN,SUBCODE,SSID));
```

Inserting the values

```
INSERT INTO STUDENT VALUES('4MT15CS001','ABHIRAM','KERALA',7658475647,'M');
INSERT INTO STUDENT VALUES('4MT15CS003','ADITHYA','KERALA',8658495647,'M');
INSERT INTO STUDENT VALUES('4MT15CS005','AKHIL','KERALA',8858475947,'M');
INSERT INTO STUDENT VALUES('4MT15CS006','AKSHATHA','MANGALORE',9958475697,'F');
INSERT INTO STUDENT VALUES('4MT15CS007','AKSHAY','MOODBIDRI',7758475647,'M');
INSERT INTO STUDENT VALUES('4MT15CS008','ALOMA','MANGALORE',9658475697,'M);
INSERT INTO STUDENT VALUES('4MT15CS010','ANANTHA','KARKALA',8656975647,'M');
INSERT INTO STUDENT VALUES('4MT15CS011','ANIL','KERALA',9658471647,'M');
INSERT INTO STUDENT VALUES('4MT15CS012','ANKITHA','MANGALORE',9988275647,'F');
INSERT INTO STUDENT VALUES('4MT15CS014','ANUSHA','MANGALORE',7658987447,'F');
INSERT INTO STUDENT VALUES('4MT15CS015','ANUSHREE','MOODBIDRI',9988445522,'F');
INSERT INTO STUDENT VALUES('4MT15CS017','ASHIKA','MANGALORE',8866335577,'F');


INSERT INTO SEMSEC VALUES( 1 ,5,'A');
INSERT INTO SEMSEC VALUES(2,5,'B');
INSERT INTO SEMSEC VALUES(3,7,'A');
INSERT INTO SEMSEC VALUES(4,7,'B');
INSERT INTO SEMSEC VALUES(5,3,'A');
INSERT INTO SEMSEC VALUES(6,3,'B');

INSERT INTO CLASS VALUES('4MT15CS001',L);
INSERT INTO CLASS VALUES('4MT15CS003',L);
INSERT INTO CLASS VALUES('4MT15CS005',L);
INSERT INTO CLASS VALUES('4MT15CS006',L);
INSERT INTO CLASS VALUES('4MT15CS007',L);
INSERT INTO CLASS VALUES('4MT15CS008',L);
INSERT INTO CLASS VALUES('4MT15CS010',L);
INSERT INTO CLASS VALUES('4MT15CS011',2);
INSERT INTO CLASS VALUES('4MT15CS012',2);
INSERT INTO CLASS VALUES('4MT15CS014',2);
INSERT INTO CLASS VALUES('4MT15CS015',2);
INSERT INTO CLASS VALUES('4MT15CS017',2);

INSERT INTO SUBJECT VALUES('15CS51','ME',5,4);
INSERT INTO SUBJECT VALUES('15CS52', 'CN',5,4);
INSERT INTO SUBJECT VALUES('15CS53','DBMS',5,4);
INSERT INTO SUBJECT VALUES('15CS54','ATC',5,4);
INSERT INTO SUBJECT VALUES('15CS553','ADVANCED JAVA',5,4);
INSERT INTO SUBJECT VALUES('15CS561','DOT NET',5,4);
```

```
INSERT INTO IAMARKS VALUES('4MT15CS001','15CS51',1,17,18,15,NULL);
INSERT INTO IAMARKS VALUES('4MT15CS001','15CS52',1,12,18,13,NULL);
INSERT INTO IAMARKS VALUES('4MT15CS001','15CS53',1,10,14,17,NULL);
INSERT INTO IAMARKS VALUES('4MT15CS001','15CS54',1,11,18,14,NULL);
INSERT INTO IAMARKS VALUES('4MT15CS001','15CSS53',1,17,18,19,NULL);
INSERT INTO IAMARKS VALUES('4MT15CS001','15CS561',1,8,8,18,NULL);
INSERT INTO IAMARKS VALUES('4MT15CS003','15CS51',1,17,18,15,NULL);
INSERT INTO IAMARKS VALUES('4MT15CS003','15CS52',1,12,18,13,NULL);
INSERT INTO IAMARKS VALUES('4MT15CS003','15CS53',1,10,14,17,NULL);
INSERT INTO IAMARKS VALUES('4MT15CS003','15CS54',1,11,18,14,NULL);
INSERT INTO IAMARKS VALUES('4MT15CS003','15CSS53',1,17,18,19,NULL);
INSERT INTO IAMARKS VALUES('4MT15CS003','15CS561',1,8,8,18,NULL);
INSERT INTO IAMARKS VALUES('4MT15CS005','15CS51',1,17,18,15,NULL);
INSERT INTO IAMARKS VALUES('4MT15CS005','15CS52',1,12,18,13,NULL);
INSERT INTO IAMARKS VALUES('4MT15CS005','15CS53',1,10,14,17,NULL);
INSERT INTO IAMARKS VALUES('4MT15CS005','15CS54',1,11,18,14,NULL);
INSERT INTO IAMARKS VALUES('4MT15CS005','15CSS53',1,17,18,19,NULL);
INSERT INTO IAMARKS VALUES('4MT15CS005','15CS561',1,8,8,18,NULL);
INSERT INTO IAMARKS VALUES('4MT15CS006','15CS51',1,17,18,15,NULL);
INSERT INTO IAMARKS VALUES('4MT15CS006','15CS52',1,12,18,13,NULL);
INSERT INTO IAMARKS VALUES('4MT15CS006','15CS53',1,10,14,17,NULL);
INSERT INTO IAMARKS VALUES('4MT15CS006','15CS54',1,11,18,14,NULL);
INSERT INTO IAMARKS VALUES('4MT15CS006','15CSS53',1,17,18,19,NULL);
INSERT INTO IAMARKS VALUES('4MT15CS006','15CS561',1,8,8,18,NULL);
INSERT INTO IAMARKS VALUES('4MT15CS007','15CS51',1,17,18,15,NULL);
INSERT INTO IAMARKS VALUES('4MT15CS007','15CS52',1,12,18,13,NULL);
INSERT INTO IAMARKS VALUES('4MT15CS007','15CS53',1,10,14,17,NULL);
INSERT INTO IAMARKS VALUES('4MT15CS007','15CS54',1,11,18,14,NULL);
INSERT INTO IAMARKS VALUES('4MT15CS007','15CSS53',1,17,18,19,NULL);
INSERT INTO IAMARKS VALUES('4MT15CS007','15CS561',1,8,8,18,NULL);
```

Query 1:

select s.usn,s.sname,address,phone,gender from student s, semsec ss, class c where s.usn = c.usn and c.ssid = ss.ssid and sem =5 and sec ='B';

Output:

USN	SNAME	ADDRESS	PHONE	G
4MT15CS011	ANIL KUMAR	KERALA	7657561234	M
4MT15CS012	ANKITHA	MANGALORE	9876555647	F
4MT15CS014	ANUSHA	MANGALORE	798123457	F
4MTl 5CS015	ANUSHREE M	MOODBIDRI	7653456743	M
4MT15CS017	ASHIKA SHETTY	MANGALORE	7877564547	F

Query 2:

select ss.sem, ss.sec,s.gender,count(s.gender) as count from student s, semsec ss,class c where s.usn=c.usn and ss.ssid=c.ssid group by ss.sem,ss.sec,s.gender order by sem;

Output:

SEM	S	No of Male Students	No of Female Students
5	A	5	2
5	B	2	3

Query 3:

create view stu_test1_marks_view as select test1, subcode from iamarks where usn='4MT15CS005';

select * from stu_test1_marks_view;

Output:

Usn	Student Name	SubjectCode	Subject Name	Test1 Marks
4MT15CS005	AKHIL	15CS51	ME	17
4MT15CS005	AKHIL	15CS52	CN	12
4MT15CS005	AKHIL	15CS53	DBMS	10
4MT15CS005	AKHIL	15CS54	ATC	11
4MT15CS005	AKHIL	15CS553	JAVA	17
4MT15CS005	AKHIL	15CS561	DOT NET	8

Query 4:

```
CREATE OR REPLACE PROCEDURE AVGMARKS IS CURSOR C_IAMARKS IS
SELECT GREATEST(TEST1,TEST2) AS A, GREATEST(TEST1,TEST3) AS B,
GREATEST(TEST3,TEST2) AS C FROM IAMARKS WHERE FINALIA IS NULL FOR UPDATE
C_A NUMBER;
C_B NUMBER;
C_C NUMBER;
C_SM NUMBER;
C_AV NUMBER;


BEGIN
OPEN C_IAMARKS; LOOP
FETCH C_IAMARKS INTO C_A, C_B, C_C; EXIT WHEN C_IAMARKS%NOTFOUND;
--DBMS_OUTPUT.PUT_LINE(C_A || ' ' || C_B || ' ' ||C_C); IF (C_A != C_B)
THEN C_SM:=C_A+C_B; ELSE C_SM:=C_A+C_C; END IF;
C_AV:=C_SM/2;
--DBMS_OUTPUT.PUT_LINE('SUM = '||C_SM);
--DBMS_OUTPUT.PUT_LINE('AVERAGE = '||C_AV);
UPDATE IAMARKS SET FINALIA=C_AV WHERE CURRENT OF C_IAMARKS; END LOOP;
CLOSE C_IAMARKS;
END;
/
```

Output:

1 ROW UPDATED

Query 5:

select s.usn,sname,finalia, (CASE WHEN finalia>=17 and finalia<=20 then 'Outstanding' WHEN finalia>=12 and finalia<17 then 'Average' WHEN finalia<12 then 'Weak' END) CAT from student s, iamarks ia where s.usn=ia.usn;

Output:

USN	SNAME	FINALIA	CAT
4MT15CS001	ABHIRAM	17.5	OUTSTANDING
4MT15CS001	ABHIRAM	15.5	AVERAGE
4MT15CS001	ABHIRAM	16	AVERAGE
4MT15CS001	ABHIRAM	18.5	OUTSTANDING
4MT15CS001	ABHIRAM	13	AVERAGE
4MT15CS001	ADHITYA	15.5	AVERAGE
4MT15CS001	ADHITYA	15.5	AVERAGE
4MT15CS001	ADHITYA	16	AVERAGE
4MT15CS001	ADHITYA	18.5	OUTSTANDING
4MT15CS001	ADHITYA	13	AVERAGE
4MT15CS001	AKHIL	17.5	OUTSTANDING
4MT15CS001	AKHIL	15.5	AVERAGE
4MT15CS001	AKHIL	15.5	AVERAGE
4MT15CS001	AKHIL	16	OUTSTANDING
4MT15CS001	AKHIL	18.5	OUTSTANDING
4MT15CS001	AKHIL	13	AVERAGE

EXPERIMENTS NO.5: EMPLOYEE Database

E.Consider the following schema for Company Database:

EMPLOYEE(SSN, Name, Address, Sex, Salary, SuperSSN, DNo)
DEPARTMENT(DNo, DName, MgrSSN, MgrStartDate)
DLOCATION(DNo,DLoc)
PROJECT(PNo, PName, PLocation, DNo)
WORKS_ON(SSN, PNo, Hours)

Write SQL queries to

1. Make a list of all project numbers for projects that involve an employee whose last name is 'Scott', either as a worker or as a manager of the department that controls the project.
2. Show the resulting salaries if every employee working on the 'IoT' project is given a 10 percent raise.
3. Find the sum of the salaries of all employees of the 'Accounts' department, as well as the maximum salary, the minimum salary, and the average salary in this department.
4. Retrieve the name of each employee who works on all the projects controlled by department number 5 (use NOT EXISTS operator).
5. For each department that has more than five employees, retrieve the department number and the number of its employees who are making more than Rs.6,00,000.

Creating the tables

```
CREATE TABLE EMPLOYEE5( SSN INT PRIMARY KEY, NAME VARCHAR(20),
ADDRESS VARCHAR(30), SEX VARCHAR(3), SALARY INT, SUPERSSN INT, FOREIGN
KEY(SUPERSSN) REFERENCES EMPLOYEE5(SSN), DNO INT,FOREIGN KEY(DNO) REFERENCES
DEPARTMENT5(DNO));


CREATE TABLE DEPARTMENT5( DNO INT PRIMARY KEY, DNAME VARCHAR(20));

CREATE TABLE DLOCATION5( DNO INT, DLOC VARCHAR(20), PRIMARY KEY
(DNO,DLOC),FOREIGN KEY (DNO) REFERENCES DEPARTMENT5(DNO));

CREATE TABLE PROJECT5( PNO INT PRIMARY KEY, PNAME VARCHAR(20),
PLOCATION VARCHAR(20), DNO INT, FOREIGN KEY (DNO)REFERENCES DEPARTMENT5(DNO));

CREATE TABLE WORKS_ON5( SSN INT , PNO INT, FOREIGN KEY (SSN) REFERENCES
EMPLOYEE5(SSN), FOREIGN KEY (PNO) REFERENCES PROJECT5(PNO), HOURS INT,
PRIMARY KEY( SSN,PNO));


INSERT INTO EMPLOYEE5 VALUES(123456789, 'JOHN SMITH', 'HOUSTON','M', 25000, NULL,1);
INSERT INTO EMPLOYEE5 VALUES(123123123, 'SCOTT', 'HOUSTON', 'M', 25000, 123456789,1);
INSERT INTO EMPLOYEE5 VALUES(134134134, 'ALICIA', 'HOUSTON', 'F' ,23000, 156156156,5);
INSERT INTO EMPLOYEE5 VALUES(124124124, 'RAMESH', 'INDIA', 'M', 612000, 156156156,5);
INSERT INTO EMPLOYEE5 VALUES(145145145, 'NARAYANT', 'INDIA', 'M', 25000, 156156156, 5);
```

```
INSERT INTO EMPLOYEE5 VALUES(156156156, 'FRANKLIN', 'TEXAS', 'M', 775000, NULL,5);
INSERT INTO EMPLOYEE5 VALUES(167167167, 'GANESH', 'TEXAS', 'M', 79000, 178178178, 2);
INSERT INTO EMPLOYEE5 VALUES(178178178, 'MAHESH', 'HOUSTON', 'M', 610000, NULL, 2);
INSERT INTO EMPLOYEE5 VALUES(189189189, 'PRAKASH', 'HOUSTON', 'M', 85000, 156156156, 5);
INSERT INTO EMPLOYEE5 VALUES(190190190, 'SMITH', 'TEXAS', 'M', 75000, 178178178, 2);


INSERT INTO DEPARTMENT5 VALUES( 1 ,'ACCOUNTS');
INSERT INTO DEPARTMENT5 VALUES(2,'HEADQUARTERS');
INSERT INTO DEPARTMENT5 VALUES(5,'ADMINISTRATION');


INSERT INTO DLOCATION5 VALUES( 1,'HOUSTON');
INSERT INTO DLOCATION5 VALUES( 1 ,'TEXAS');
INSERT INTO DLOCATION5 VALUES(2,'INDIA');
INSERT INTO DLOCATION5 VALUES(5,'HOUSTON');
INSERT INTO DLOCATION5 VALUES(5,'SUGARLAND');
 INSERT INTO DLOCATION5 VALUES(5,'INDIA');


INSERT INTO PROJECT5 VALUES(101,'IOT','HOUSTON', 1);
INSERT INTO PROJECT5 VALUES( 102,'PRODUCTX','HOUSTON',5);
INSERT INTO PROJECT5 VALUES( 105,'PRODUCTY','SUGARLAND',2);
INSERT INTO PROJECT5 VALUES( 106,'PRODUCTZ','HOUSTON',5);


INSERT INTO WORKS_ON5 VALUES(123456789,101,10);
INSERT INTO WORKS_ON5 VALUES(123456789,106,20);
INSERT INTO WORKS_ON5 VALUES(123123123,102,30);
INSERT INTO WORKS_ON5 VALUES(123456789,105,10);
INSERT INTO WORKS_ON5 VALUES(124124124,101,5);
INSERT INTO WORKS_ON5 VALUES(124124124,106,40);
INSERT INTO WORKS_ON5 VALUES(145145145,102,20);
INSERT INTO WORKS_ON5 VALUES(145145145,106,20);
INSERT INTO WORKS_ON5 VALUES(156156156, 105,30);
INSERT INTO WORKS_ON5 VALUES(167167167,106,10);
INSERT INTO WORKS_ON5 VALUES( 178178178, 105,5);
INSERT INTO WORKS_ON5 VALUES( 189189189,101,10);
```

Query 1:

select pno from employee5 e, works_on5 w where e.ssn=w.ssn and name='Scott' UNION select pno from employee5 e, department5 d, project5 p where e.ssn=d.mgrssn and d.dno=p.pno and name='Scott' ;

Output:

PNO 102

105

Query 2:

select Name, Salary "Old Salary", Salary* 1.1 "New Salary" from employee5 e,works_on5 w, project5 p where e.ssn =w.ssn and w.pno= p.pno and pname = 'IoT';

Output:

NAME	OLD SALARY	NEW SALARY
JOHN	25000	27500
ALICIA	23000	25300
PRAKASH	85000	93500

Query 3:

select sum(salary) "Total Salary", Max(Salary) "Maximum Salary" , Min(Salary) "Minimum Salary", Avg(Salary) "Average Salary" from employee5 e, department5 d where e.dno= d.dno and dname='Accounts';

Output:

TOTAL SALARY	MAX SALARY	MIN SALARY	AVERAGE SALARY
50000	25000	25000	25000

Query 4:

select e.name from employee5 e where not exists ((select pno from project5 where dno=5) except (select pno from works_on5 where e.ssn= ssn));

Output:
SSN 145145145

Query 5:

select dno, (select count(*) from employee5 where salary>600000 and dno in (select dno from employee5 group by dno having count(*)>=5)) as count_salary from employee5 group by dno having count(*)>=5;

Output:

DNO	NO_OF_EMP_MORE_THAN_600000
5	2

Viva Questions

1. What is SQL?

It's a Query Language and Structured Query Language

2. What is database?

A database is a logically coherent collection of data with some inherent meaning, representing some aspect of real world and which is designed, built and populated with data for a specific purpose.

3. What is DBMS?

It is a collection of programs that enables user to create and maintain a database. In other words it is general-purpose software that provides the users with the processes of defining, constructing and manipulating the database for various applications.

4. What is a Database system?

The database and DBMS software together is called as Database system.

5. Advantages of DBMS?

- Redundancy is controlled.
- Unauthorized access is restricted.
- Providing multiple user interfaces.
- Enforcing integrity constraints.
- Providing backup and recovery.

6. Disadvantage in File Processing System?

- Data redundancy &inconsistency.
- Difficult in accessing data.
- Data isolation.
- Data integrity.
- Concurrent access is not possible.
- Security Problems.

7. Describe the three levels of data abstraction? There are three levels of abstraction:

   - Physical level: The lowest level of abstraction describes how data are stored.
   - Logical level*:* The next higher level of abstraction, describes what data are stored in database and what relationship among those data.
   - View level: The highest level of abstraction describes only part of entire atabase.

- Define the "integrity rules" There are two Integrity rules.
- Entity Integrity: States that—Primary key cannot have NULL value□
- Referential Integrity:States that —Foreign Key can be either a NULL value or should be Primary Key value of other relation.

8. What is extension andintension?

Extension - It is the number of tuples present in a table at any instance. This is time dependent.

Intension -It is a constant value that gives the name, structure of table and the constraints laid on it.

9. What is Data Independence?

   Dataindependencemeansthat—theapplicationisindependentofthestoragestructureand access strategy of data□. In other words, The ability to modify the schema definition in one level should not affect the schema definition in the next higher level.

Two types of Data Independence:

- ➢ Physical Data Independence: Modification in physical level should not affect the logical level.
- ➢ Logical Data Independence: Modification in logical level should affect theview level.

10. What is a view? How it is related to data independence?

A view may be thought of as a virtual table, that is, a table that does not really exist in its own right but is instead derived from one or more underlying base table. In other words, there is no stored file that direct represents the view instead a definition of view is stored in data dictionary.

Growth and restructuring of base tables is not reflected in views. Thus the view can insulate users from the effects of restructuring and growth in the database. Hence accounts for logical data independence.

11. What is Data Model?

A collection of conceptual tools for describing data, data relationships data semantics and constraints.

12. What is E-R model?

This data model is based on real world that consists of basic objects called entities and of relationship among these objects. Entities are described in a database by a set of attributes.

13. What is Object Oriented model?

This model is based on collection of objects. An object contains values stored in instance variables within the object. An object also contains bodies of code that operate on the object. These bodies of code are called methods. Objects that contain same types of values and the same methods are grouped together into classes.

14. What is an Entity?

It is an 'object' in the real world with an independent existence.

15. What is an Entity type?

It is a collection (set) of entities that have same attributes.

16. What is an Entity set?

It is a collection of all entities of particular entity type in the database.

17. What is an Extension of entity type?

The collections of entities of a particular entity type are grouped together into an entity set.

18. What is an attribute?

It is a particular property, which describes the entity.

19. What is a Relation Schema and a Relation?

A relation Schema denoted by R(A1, A2, …, An) is made up of the relation name R and the list of attributes A_i that it contains. A relation is defined as a set of tuples. Let r be the relation which contains set tuples (t1,t2, t3,...,tn). Each tuple is an ordered list of n- values t=(v1,v2, ...,vn).

20. What is degree ofa Relation?

It is the number of attribute of its relation schema.

21. What is Relationship?

It is an association among two or more entities.

22. What is Relationship set?

The collection (or set) of similar relationships.

23. What is Relationship type?

Relationship type defines a set of associations or a relationship set among a given set of entity types.

24. What is degree ofRelationship type?

It is the number of entity type participating.

25. What is DDL (DataDefinition Language)?

A data base schema is specified by a set of definitions. expressed by a special language called DDL.

26. What is VDL (ViewDefinition Language)?

It specifies user views and their mappings to the conceptual schema.

27. What is SDL (StorageDefinition Language)?

This language is to specify the internal schema. This language may specify the mapping between two schemas.

28. What is Data Storage – Definition Language?

The storage structures and access methods used by database system are specified by a set of definition in a special type of DDL called data storage-definition language.

29. What is DML (DataManipulation Language)?

This language that enable user to access or manipulate data as organized by appropriate data model.

- Procedural DML or Low level: DML requires a user to specify what data are needed and how to get those data.
- Non-Procedural DML or High level: DML requires a user to specify what data are needed without specifying how to get those data.

30. What is DML Compiler?

It translates DML statements in a query language into low-level instruction that the query evaluation engine can understand.

31. What is Relational Algebra?

It is a procedural query language. It consists of a set of operations that take one or two relations as input and produce a new relation.

32. What is Relational Calculus?

It is an applied predicate calculus specifically tailored for relational databases proposed by E.F. Codd. E.g. of languages based on it are DSL, ALPHA,QUEL.

33. What is normalization?

- ➢ It is a process of analyzing the given relation schemas based on their Dependencies (FDs) and primary key to achieve the properties
- ➢ Minimizing redundancy
- ➢ Minimizing insertion, deletion and update anomalies.

34. What is Functional Dependency?

A Functional dependency is denoted by X Y between two sets ~~of~~ attributes X and Y thataresubsetsofRspecifiesaconstraintonthepossibletuplethatcanformarelationstaterof R. The constraint is for any two tuples t1 and t2 in r if t1[X] = t2[X] then they have t1[Y] = t2[Y]. This means the value of X component of a tuple uniquely determines the value of component Y.

35. When is a functional dependency F said to be minimal?

- ➢ Every dependency in F has a single attribute for its right hand side.
- ➢ We cannot replace any dependency X A in F with a dependencyY A where Y is a proper subset of X and still have a set of dependency that is equivalent to F.
- ➢ We cannot remove any dependency from F and still have set of dependency that is equivalent to F.

36. What is Multivalued dependency?

Multivalued dependency denoted by X Y pecified on relation schema R, where X and Y are both subsets of R, specifies the following constraint on any relation r of R: if two tuples t1 and t2 exist in r such that t1[X] = t2[X] then t3 and t4 should also exist in r with the following properties

- ➢ t3[x] = t4[X] = t1[X] =t2[X]
- ➢ t3[Y] = t1[Y] and t4[Y] =t2[Y]
- ➢ t3[Z] = t2[Z] and t4[Z] = t1[Z]

where [Z = (R-(X U Y))]

37. What is Lossless joinproperty?

It guarantees that the spurious tuple generation does not occur with respect to relation schemas after decomposition.

38. What is 1 NF(Normal Form)?

The domain of attribute must include only atomic (simple, indivisible) values.

39. What is Fully Functional dependency?

It is based on concept of full functional dependency. A functional dependency X Y is fully functional dependency if removal of any attribute A from X means that the dependency does not hold anymore.

40. What is2NF?

A relation schema R is in 2NF if it is in 1NF and every non-prime attribute A in R is fully functionally dependent on primary key.

41. What is BCNF (Boyce-CoddNormal Form)?

A relation schema R is in BCNF if it is in 3NF and satisfies additional constraints that for every FD X A, X must be a candidate key.

42. What is4NF?

A relation schema R is said to be in 4NF if for every Multivalued dependency X

Y that holds over R, one of following is true

- X is subset or equal to (or) XY =R.
- X is a super key.

43. What is5NF?

A Relation schema R is said to be 5NF if for every join dependency {R1, R2, ...,Rn} that holds R, one the following is true

- Ri = R for somei.
- The join dependency is implied by the set of FD, over R in which the left side is keyofR.

44. What is Domain-Key NormalForm?

A relation is said to be in DKNF if all constraints and dependencies that should hold on the constraint can be enforced by simply enforcing the domain constraint and key constraint on the relation.