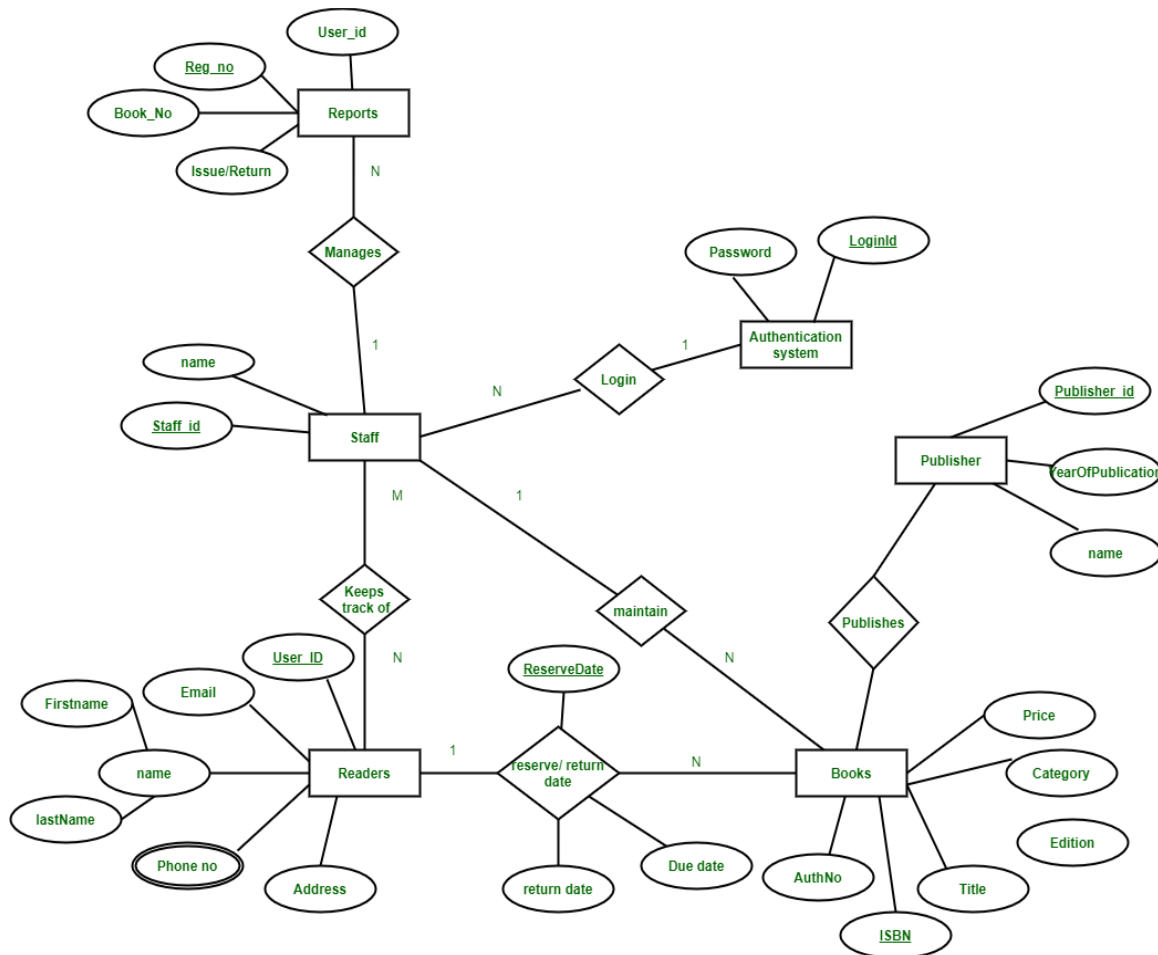


1) Convert the below ER diagram into row based logical model ed

Devakesh.V(6957)



Answer:-

1. AUTHENTICATION_SYSTEM (LOGIN_ID (PRIMARY KEY), PASSWORD);
2. STAFF (STAFF_ID (PRIMARY KEY), NAME, LOGIN_ID (FOREIGN KEY -> AUTHENTICATION_SYSTEM));
3. REDEARS(USER_ID(PRIMARY KEY), EMAIL, FIRST_NAME, LAST_NAME, ADDRESS);
4. REPORTS(USER_ID (FOREIGN KEY -> REDEARES ,REG_NO (PRIMARY KEY), BOOK_NO, ISSUE, STAFF_ID (FOREIGN KEY -> STAFF));
5. PHONE_NUMBER(USER_ID(FOREIGN KEY -> REDEARS, PHONE_NUMBER);
6. KEEPS_TRACK_OFF(USER_ID(FOREIGN KEY -> REDEARS, STAFF_ID(FOREIGN KEY -> STAFF);
7. RETURN_DATE(RESERVE_DATE (PRIMARY KEY), RETURN_DATE, DUE_DATE);
8. BOOKS(ISBN (PRIMARY KEY), AUTH_NO, TITTLE, EDITION, CATEGORY, PRICE, USER_ID(FOREIGN KEY -> REDEARS), STAFF_ID (FOREIGN KEY -> STAFF));
9. PUBLISHER(PUBLISHER_ID (PRIMARY KEY), YEAR_OF_PRODUCTION, NAME);
10. PUBLISH(PUBLISHER_ID (FOREIGN KEY -> PUBLISHER), ISBN (FOREIGN KEY -> BOOKS);

- =====
- 2) Create the tables and insert at least 3 records in each table for the above record based logical model

Answer:-

- a. CREATE TABLE AUTHENTICATION_SYSTEM(LOGIN_ID NUMBER CONSTRAINT LOGIN_PK PRIMARY KEY, PASSWORD VARCHAR2(20));

INSERT INTO AUTHENTICATION_SYSTEM VALUES(6700, 'ABD123@456');

INSERT INTO AUTHENTICATION_SYSTEM VALUES(6701, 'ABC123\$456');

INSERT INTO AUTHENTICATION_SYSTEM VALUES(6702, 'ABE123&456');

LOGIN_ID	PASSWORD
6701	ABC123\$456
6702	ABE123&456
6700	ABD123@456

[Download CSV](#)

- b. CREATE TABLE STAFF(STAFF_ID NUMBER CONSTRAINT STAFF_PK PRIMARY KEY, NAME VARCHAR2(20), LOGIN_ID NUMBER, FOREIGN KEY(LOGIN_ID) REFERENCES AUTHENTICATION_SYSTEM(LOGIN_ID));

INSERT INTO STAFF VALUES(546, 'DEVA', 6701);

INSERT INTO STAFF VALUES(547, 'SARA', 6700);

INSERT INTO STAFF VALUES(548, 'UDAY', 6701);

INSERT INTO STAFF VALUES(549, 'AJAY', 6702);

STAFF_ID	NAME	LOGIN_ID
546	DEVA	6701
547	SARA	6700
548	UDAY	6701
549	AJAY	6702

[Download CSV](#)

- c. CREATE TABLE REDEARS(USER_ID NUMBER CONSTRAINT USER_PK PRIMARY KEY, EMAIL VARCHAR2(30), FIRST_NAME VARCHAR2(20), LAST_NAME VARCHAR2(20), ADDRESS VARCHAR2(40));

INSERT INTO REDEARS VALUES(9100, 'xavier@yahoo.com', 'xavier', 'carls', 'USA');

INSERT INTO REDEARS VALUES(9101, 'mighty@yahoo.com', 'mighty', 'ran', 'austraila');

INSERT INTO REDEARS VALUES(9102, 'tony@gmail.com', 'tony', 'stark', 'USA');

INSERT INTO REDEARS VALUES(9103, 'chris@hotmail.com', 'chris', 'hemsworth', 'USA');

USER_ID	EMAIL	FIRST_NAME	LAST_NAME	ADDRESS
9100	xavier@yahoo.com	xavier	carls	USA
9101	mighty@yahoo.com	mighty	ran	austraila
9102	tony@gmail.com	tony	stark	USA
9103	chris@hotmail.com	chris	hemsworth	USA

[Download CSV](#)

- d. CREATE TABLE REPORTS(USER_ID NUMBER, REG_NO NUMBER CONSTRAINT REG_PK PRIMARY KEY, BOOK_NO NUMBER, ISSUE VARCHAR2(20), STAFF_ID NUMBER, FOREIGN KEY(STAFF_ID) REFERENCES STAFF(STAFF_ID), FOREIGN KEY(USER_ID) REFERENCES REDEARS(USER_ID));

INSERT INTO REPORTS VALUES(9103, 23478, 134, 'wrong printing', 547);

INSERT INTO REPORTS VALUES(9102, 23479, 133, 'wrong book', 548);

INSERT INTO REPORTS VALUES(9100, 23480, 134, 'missing pages', 548);

INSERT INTO REPORTS VALUES(9103, 23481, 130, null, 549);

USER_ID	REG_NO	BOOK_NO	ISSUE	STAFF_ID
9103	23478	134	wrong printing	547
9102	23479	133	wrong book	548
9100	23480	134	missing pages	548
9103	23481	130	-	549

[Download CSV](#)

- e. CREATE TABLE PHONE_NUMBER(USER_ID NUMBER, FOREIGN KEY(USER_ID) REFERENCES REDEARS(USER_ID), PHONE_NUMBER NUMBER);

```
INSERT INTO PHONE_NUMBER VALUES(9103, 8994578905);
```

```
INSERT INTO PHONE_NUMBER VALUES(9102, 8945267890);
```

```
INSERT INTO PHONE_NUMBER VALUES(9101, 9876440967);
```

```
INSERT INTO PHONE_NUMBER VALUES(9100, 7803456278);
```

USER_ID	PHONE_NUMBER
9102	8945267890
9101	9876440967
9100	7803456278
9103	8994578905

[Download CSV](#)

- f. CREATE TABLE KEEPS_TRACK_OFF(USER_ID NUMBER, FOREIGN KEY(USER_ID) REFERENCES REDEARS(USER_ID), STAFF_ID NUMBER, FOREIGN KEY(STAFF_ID) REFERENCES STAFF(STAFF_ID));

```
INSERT INTO KEEPS_TRACK_OFF VALUES(9103, 546);
```

```
INSERT INTO KEEPS_TRACK_OFF VALUES(9102, 547);
```

```
INSERT INTO KEEPS_TRACK_OFF VALUES(9100, 548);
```

```
INSERT INTO KEEPS_TRACK_OFF VALUES(9103, 547);
```

```
INSERT INTO KEEPS_TRACK_OFF VALUES(9101, 549);
```

```
INSERT INTO KEEPS_TRACK_OFF VALUES(9102, 546);
```

USER_ID	STAFF_ID
9103	546
9102	547
9100	548
9103	547
9101	549
9102	546

[Download CSV](#)

- g. CREATE TABLE RETURN_DATE(RESERVE_DATE DATE CONSTRAINT RESERVE_PK PRIMARY KEY, RETURN_DATE DATE, DUE_DATE DATE);

```
INSERT INTO RETURN_DATE VALUES(TO_DATE('22042000','DDMMYYYY'),  
TO_DATE('23072001','DDMMYYYY'), TO_DATE('23072001','DDMMYYYY'));
```

```
INSERT INTO RETURN_DATE VALUES(TO_DATE('01011999','DDMMYYYY'),  
TO_DATE('01012002','DDMMYYYY'), TO_DATE('10012002','DDMMYYYY'));
```

```
INSERT INTO RETURN_DATE VALUES(TO_DATE('05061998','DDMMYYYY'),  
TO_DATE('09092000','DDMMYYYY'), TO_DATE('19092000','DDMMYYYY'));
```

RESERVE_DATE	RETURN_DATE	DUE_DATE
22-APR-00	23-JUL-01	23-JUL-01
01-JAN-99	01-JAN-02	10-JAN-02
05-JUN-98	09-SEP-00	19-SEP-00

[Download CSV](#)

- h. CREATE TABLE BOOKS(ISBN NUMBER CONSTRAINT ISBN_PK PRIMARY KEY, AUTH_NO NUMBER, TITTLE VARCHAR2(20), EDITION VARCHAR2(20), CATEGORY VARCHAR2(20), PRICE NUMBER, USER_ID NUMBER, FOREIGN KEY(USER_ID) REFERENCES REDEARS(USER_ID), STAFF_ID NUMBER, FOREIGN KEY(STAFF_ID) REFERENCES STAFF(STAFF_ID));

```
INSERT INTO BOOKS VALUES(9004567, 90012, 'I AM LEGEND', 'VOL-1', 'MOTIVATION', 2099,  
9103, 546);
```

```
INSERT INTO BOOKS VALUES(9104567, 90013, 'I AM KING', 'VOL-4', 'BIOGRAPHY', 2199, 9100,  
548);
```

```
INSERT INTO BOOKS VALUES(9034567, 90014, 'NIGHT HUNTER', 'VOL-2', 'CRIME', 3099, 9103,  
549);
```

```
INSERT INTO BOOKS VALUES(9134567, 90015, 'SERIAL KILLER', 'VOL-3', 'CRIME', 5099, 9102,  
547);
```

ISBN	AUTH_NO	TITTLE	EDITION	CATEGORY	PRICE	USER_ID	STAFF_ID
9004567	90012	I AM LEGEND	VOL-1	MOTIVATION	2099	9103	546
9104567	90013	I AM KING	VOL-4	BIOGRAPHY	2199	9100	548
9034567	90014	NIGHT HUNTER	VOL-2	CRIME	3099	9103	549
9134567	90015	SERIAL KILLER	VOL-3	CRIME	5099	9102	547

[Download CSV](#)

4 rows selected.

- i. CREATE TABLE PUBLISHER(PUBLISHER_ID NUMBER CONSTRAINT PUBLI_PK PRIMARY KEY, YEAR_OF_PRODUCTION DATE, NAME VARCHAR2(20));

```
INSERT INTO PUBLISHER VALUES(1643, TO_DATE('1998','YYYY'), 'chris');
```

```
INSERT INTO PUBLISHER VALUES(1646, TO_DATE('1979','YYYY'), 'TONY');
```

```
INSERT INTO PUBLISHER VALUES(1648, TO_DATE('1968','YYYY'), 'ELISA');
```

PUBLISHER_ID	YEAR_OF_PRODUCTION	NAME
1646	01-NOV-79	TONY
1648	01-NOV-68	ELISA
1643	01-NOV-98	chris

[Download CSV](#)

3 rows selected.

- j. CREATE TABLE PUBLISH(PUBLISHER_ID NUMBER, FOREIGN KEY(PUBLISHER_ID) REFERENCES PUBLISHER(PUBLISHER_ID), ISBN NUMBER, FOREIGN KEY(ISBN) REFERENCES BOOKS(ISBN));

```
INSERT INTO PUBLISH VALUES(1646, 9004567);
```

```
INSERT INTO PUBLISH VALUES(1646, 9104567);
```

```
INSERT INTO PUBLISH VALUES(1648, 9034567);
```

```
INSERT INTO PUBLISH VALUES(1643, 9104567);
```

```
INSERT INTO PUBLISH VALUES(1648, 9034567);
```

```
INSERT INTO PUBLISH VALUES(1646, 9104567);
```

```
INSERT INTO PUBLISH VALUES(1643, 9004567);
```

PUBLISHER_ID	ISBN
1646	9104567
1648	9034567
1643	9104567
1648	9034567
1646	9104567
1643	9004567
1646	9004567

[Download CSV](#)

7 rows selected.

=====

3) Create one procedure[PLSQL PROGRAM] to insert the records in EMP table with (empno, ename, sal) columns

ANSWER)-----

```
CREATE TABLE EMP(EMPNO NUMBER, ENAME VARCHAR2(20), SAL NUMBER);
```

```
DESC EMP;
```

```
CREATE OR REPLACE PROCEDURE HIRE_EMP(P_EMPNO NUMBER, P_ENAME VARCHAR2, P_SAL  
NUMBER)
```

```
AS
```

```
BEGIN
```

```
INSERT INTO EMP (EMPNO, ENAME, SAL) VALUES (P_EMPNO, P_ENAME, P_SAL);
```

```
END HIRE_EMP;
```

```
EXECUTE HIRE_EMP(6957,'DEVAKESH',24500);
```

```
EXECUTE HIRE_EMP(7019,'SUDEEP',24400);
```

```
EXECUTE HIRE_EMP(6958,'AKASH',24450);
```

```
SELECT * FROM EMP;
```

=====

4) Create one cursor to display the records from the above emp table

ANSWER)-----

```
ALTER TABLE EMP ADD DEPTNO NUMBER;
```

```
UPDATE EMP SET DEPTNO=10;
```

```
DECLARE
```

```
CURSOR EMP_CUR IS SELECT ENAME FROM EMP WHERE DEPTNO=10;
```

```
EMP_NAME VARCHAR2(20);
```

```
BEGIN
```

```
OPEN EMP_CUR;
```

```
LOOP
```

```
FETCH EMP_CUR INTO EMP_NAME;
```

```
EXIT WHEN EMP_CUR%NOTFOUND;
```

```
DBMS_OUTPUT.PUT_LINE('EMPLOYEE NAME IS ' || EMP_NAME);
```

```
END LOOP;
```

```
END;
```

=====

5) Identify any repeating groups and functional dependences in the PATIENT relation.

Show all the intermediate steps to derive this data to third normal form for PATIENT table.

PATIENT(patno, patname, gpno, gpname, appdate, consultant, conaddr, sample)

patno	patname	gpno	gpname	appdate	consultant	conaddr	sample
01027	Grist	919	Robinson	3/9/2004	Farnes	Acadia Rd	blood
				20/12/2004	Farnes	Acadia Rd	none
				10/10/2004	Edwards	Beech Ave	urine
08023	Daniels	818	Seymour	3/9/2004	Farnes	Acadia Rd	none
				3/9/2004	Russ	Fir St	CREAT Ecreapug
191146	Falken	717	Ibbotson	4/10/2004	Russ	Fir St	blood
001239	Burges	818	Seymour	5/6/2004	Russ	Fir St	sputum
007249	Lynch	717	Ibbotson	9/11/2004	Edwards	Beach Ave	none

ANSWER)-----

1NF: -

PAT NO (PRIMARY)	PAT NAME	GP NO	GP NAME
01027	Grist	919	Robins
08023	Daniels	818	Seymour
91146	Falken	717	Ibbotson
0012239	Burges	818	Seymour
007249	Lynch	717	Ibbotson

PRIMARY KEY (PAT NO + APP DATE)

PAT NO	APP DATE	CONSULTANT	CONADDR	SAMPLE
01027	03-09-2004	Farnes	Acadia RD	Blood
01027	20-12-2004	Farnes	Acadia RD	None
01027	10-10-2004	Edwards	Beach AVE	Urine
08023	03-09-2004	Farnes	Acadia RD	None
08023	03-09-2004	Russ	Fir St	creeping
91146	04-10-2004	Russ	Fir St	Blood
0012239	05-06-2004	Russ	Fir St	Sputum
007249	09-11-2004	Edwards	Beach AVE	none

2NF: -

PAT NO (PRIMARY)	PAT NAME	GP NO	GP NAME
01027	Grist	919	Robins
08023	Daniels	818	Seymour
91146	Falken	717	Ibbotson
0012239	Burges	818	Seymour
007249	Lynch	717	Ibbotson

PRIMARY KEY (PAT NO + APP DATE)

PAT NO	APP DATE	CONSULTANT	SAMPLE
01027	03-09-2004	Farnes	Blood
01027	20-12-2004	Farnes	None
01027	10-10-2004	Edwards	Urine
08023	03-09-2004	Farnes	None
08023	03-09-2004	Russ	creeping
91146	04-10-2004	Russ	Blood
0012239	05-06-2004	Russ	Sputum
007249	09-11-2004	Edwards	none

CONSULTANT(PRIMARY)	CONADDR
Farnes	Acadia RD
Edwards	Beach AVE
Russ	Fir St

3NF: -

PAT NO (PRIMARY)	PAT NAME	GP NO
01027	Grist	919
08023	Daniels	818
91146	Falken	717
0012239	Burges	818
007249	Lynch	717

GP NO (PRIMARY)	GP NAME
919	Robins
818	Seymour
717	Ibbotson

PRIMARY KEY (PAT NO + APP DATE)

PAT NO	APP DATE	CONSULTANT	SAMPLE
01027	03-09-2004	Farnes	Blood
01027	20-12-2004	Farnes	None
01027	10-10-2004	Edwards	Urine
08023	03-09-2004	Farnes	None
08023	03-09-2004	Russ	creeping
91146	04-10-2004	Russ	Blood
0012239	05-06-2004	Russ	Sputum
007249	09-11-2004	Edwards	none

CONSULTANT(PRIMARY)	CONADDR
Farnes	Acadia RD
Edwards	Beach AVE
Russ	Fir St