AMERICAN INTERNATIONAL UNIVERSITY – BANGLADESH

PROJECT: HOSPITAL MANAGEMENT SYSTEM

Course Name: Advance Database Management System

Section: B2

Submitted to: Rezwan Ahmed

Group Name: #Semicolon

Group members:

Name	ID
Zaman, Isra	15-28938-1
Showmik, MD. Jannatul Baki	15-28584-1

TABLE OF CONTENTS

SYSTEM SUMMERY

Hospital are the essential part of our lives, providing best medical facilities to people suffering from various ailments. It is necessary for the hospitals to keep track of its day-to-day activities and records of its patients, doctors, nurses and other staff personals that keep the hospital running smoothly and successfully.

But keeping track of all the activities and their records on paper is very cumbersome and error prone. It also is very inefficient and a time-consuming process observing the continuous increase in population and number of people visiting the hospital. Recording and maintaining all these records is highly unreliable, inefficient and paper-prone. It is also not economically and technically feasible to maintain these records on paper.

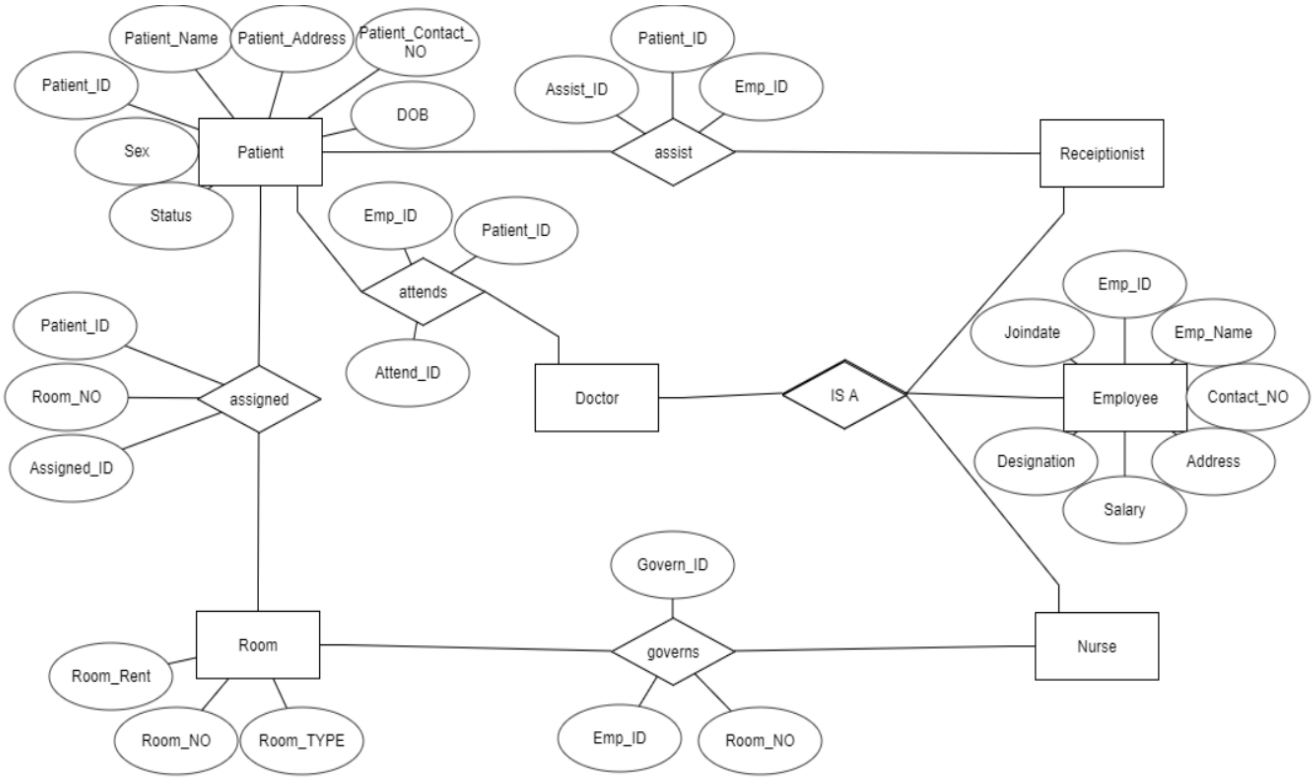
Thus keeping the working of the manual system as the basis of our project. We have developed an automated version of the manual system, names as “Hospital Management System”.

As the total project is a huge project to deal with, in this short amount of time a reasonable portion of important features will be implemented in this project.

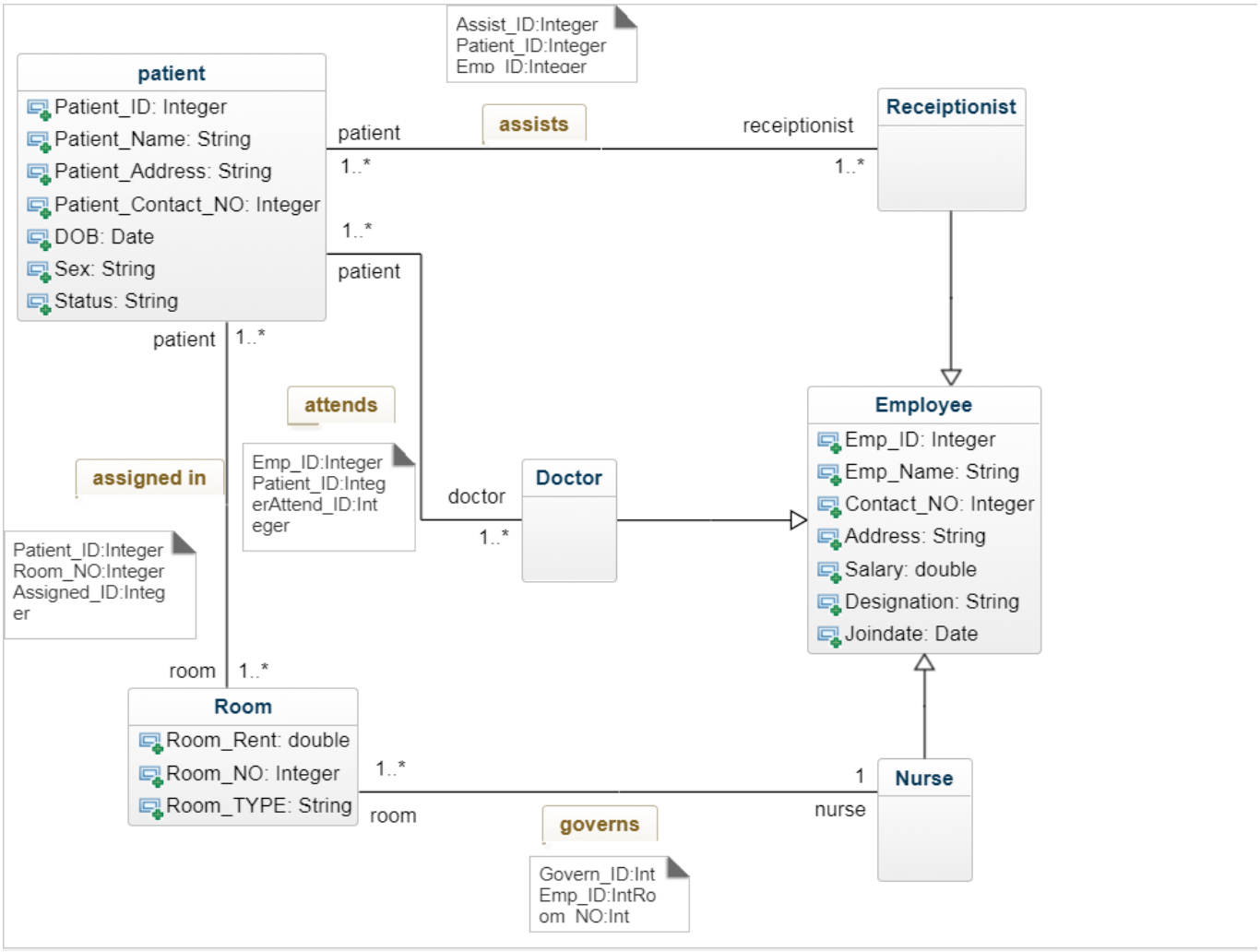
This project is aimed to develop to maintain the day-to-day state of admission/discharge of patients, list of doctors, reports generation and etc. It is designed to achieve the following objectives:

- To computerize all details regarding patient details and hospital details.
- Scheduling the appointment of patient with doctors to make it convenient for both.
- Scheduling the services of specialized doctors and emergency properly so that the facilities provided by the hospital are fully utilized in effective and efficient manner.
- The information of the patients should be kept up to date and there should be kept in the system for hospital purposes.

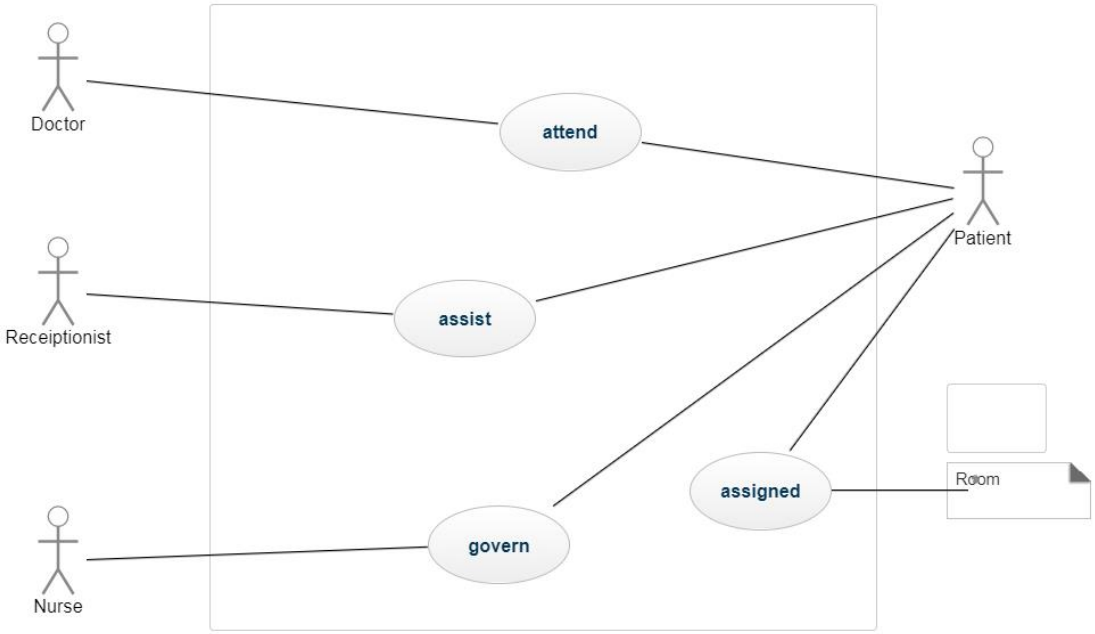
ENTITY RELATIONSHIP DIAGRAM



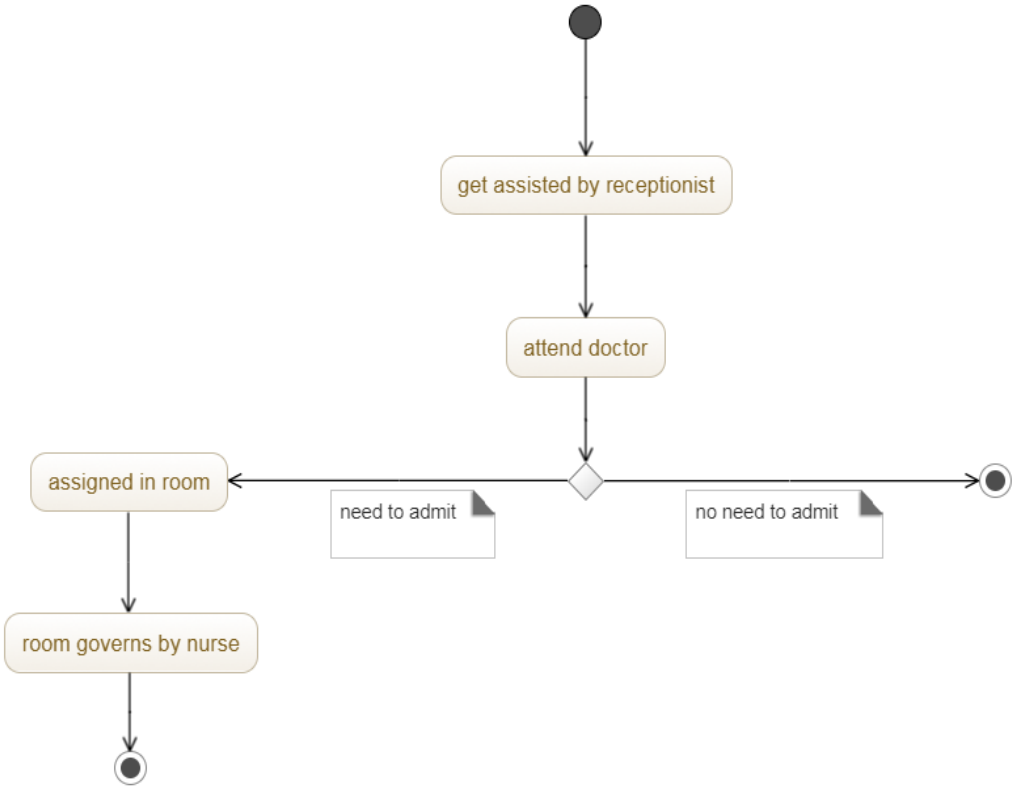
CLASS DIAGRAM



USE CASE DIAGRAM



ACTIVITY DIAGRAM



RELATIONSHIP DIAGRAMS

1. PATIENT TABLE

Column Name	Data Type	Nullable	Default	Primary Key
N_ID	NUMBER(7,0)	No	-	1
N_NAME	VARCHAR2(50)	No	-	-
N_ADDRESS	VARCHAR2(50)	No	-	-
N_CONTACT_NO	NUMBER(11,0)	No	-	-
N_JOINING_DATE	DATE	No	-	-
N_SALARY	NUMBER(10,0)	No	-	-
N_DESIGNATION	VARCHAR2(50)	No	-	-
				1 - 7

2. DOCTOR TABLE

Column Name	Data Type	Nullable	Default	Primary Key
D_ID	NUMBER(7,0)	No	-	1
D_NAME	VARCHAR2(50)	No	-	-
D_ADDRESS	VARCHAR2(50)	No	-	-
D_CONTACT_NO	NUMBER(11,0)	No	-	-
D_JOINING_DATE	DATE	No	-	-
D_SALARY	NUMBER(10,0)	No	-	-
D_DESIGNATION	VARCHAR2(50)	No	-	-
				1 - 7

3. NURSE TABLE

Column Name	Data Type	Nullable	Default	Primary Key
N_ID	NUMBER(7,0)	No	-	1
N_NAME	VARCHAR2(50)	No	-	-
N_ADDRESS	VARCHAR2(50)	No	-	-
N_CONTACT_NO	NUMBER(11,0)	No	-	-
N_JOINING_DATE	DATE	No	-	-
N_SALARY	NUMBER(10,0)	No	-	-
N_DESIGNATION	VARCHAR2(50)	No	-	-
				1 - 7

4. RECEPTIONIST TABLE

Column Name	Data Type	Nullable	Default	Primary Key
R_ID	NUMBER(7,0)	No	-	1
R_NAME	VARCHAR2(50)	No	-	-
R_ADDRESS	VARCHAR2(50)	No	-	-
R_CONTACT_NO	NUMBER(11,0)	No	-	-
R_JOINING_DATE	DATE	No	-	-
R_SALARY	NUMBER(10,0)	No	-	-
R_DESIGNATION	VARCHAR2(50)	No	-	-
				1 - 7

5. ROOM TABLE

Column Name	Data Type	Nullable	Default	Primary Key
ROOM_NO	NUMBER(7,0)	No	-	1
ROOM_TYPE	VARCHAR2(20)	No	-	-
ROOM_COST	NUMBER(5,0)	No	-	-
				1 - 3

6. ASSIGNED_ROOM TABLE

Column Name	Data Type	Nullable	Default	Primary Key
AR_ID	NUMBER(7,0)	No	-	1
P_ID	NUMBER(7,0)	No	-	-
ROOM_NO	NUMBER(7,0)	No	-	-
				1 - 3

7. ASSIST_PATIENT ROOM

Column Name	Data Type	Nullable	Default	Primary Key
AP_ID	NUMBER(7,0)	No	-	1
P_ID	NUMBER(7,0)	No	-	-
R_ID	NUMBER(7,0)	No	-	-
				1 - 3

8. ATTEND_PATIENT ROOM

Column Name	Data Type	Nullable	Default	Primary Key
ATP_ID	NUMBER(7,0)	No	-	1
P_ID	NUMBER(7,0)	No	-	-
D_ID	NUMBER(7,0)	No	-	-
				1 - 3

9. GOVERNS_ROOM TABLE

Column Name	Data Type	Nullable	Default	Primary Key
GR_ID	NUMBER(7,0)	No	-	1
N_ID	NUMBER(7,0)	No	-	-
ROOM_NO	NUMBER(7,0)	No	-	-
				1 - 3

TABLES AND SAMPLE DATA

1. PATIENT TABLE

EDIT	P_ID	P_NAME	P_ADDRESS	P_GENDER	P_DOB	P_CONTACT_NO	P_STATUS
	101	Tanjir	Basundhara	Male	11-APR-95	1811112233	In Door
	102	Arafat	Kochukhet	Male	10-JAN-90	1711804015	Out Door
	103	Wrivu	Banani	Male	02-DEC-98	1744338165	In Door
	104	Dipto	Khilkhet	Male	11-FEB-95	1933451987	In Door
	105	Bijoy	Basundhara	Male	07-FEB-97	1557998877	Out Door
	106	Kawsur	Niketon	Male	11-APR-95	1811112233	In Door

row(s) 1 - 6 of 6

2. DOCTOR TABLE

EDIT	D_ID	D_NAME	D_ADDRESS	D_CONTACT_NO	D_JOINING_DATE	D_SALARY	D_DESIGNATION
	304	ISRA	HATIRJHEEL	1833451951	07-SEP-07	50000	ASSISTANT PROFESSOR
	305	PLABON	KOCHUKHET	1933482967	12-JUL-95	25000	ASSISTANT SURGEN
	301	SOURAV	UTTARA	1811112153	21-SEP-80	30000	M.O
	302	AKIB	AIRPORT	1682408438	12-JAN-90	50000	ASSISTANT PROFESSOR
	303	SHOWMIK	BASUNDHARA	1833451951	01-FEB-04	50000	ASSISTANT PROFESSOR

row(s) 1 - 5 of 5

3. RECEPTIONIST TABLE

EDIT	R_ID	R_NAME	R_ADDRESS	R_CONTACT_NO	R_JOINING_DATE	R_SALARY	R_DESIGNATION
	501	AKASH	UTTARA	1811112153	21-SEP-80	5000	JUNIOR RECEPTIONIST
	502	ARMAN	MIRPUR	1822012153	21-JUN-80	8000	SENIOR RECEPTIONIST
	503	TARIQUR	BANANI	1701112153	21-JUL-90	5000	JUNIOR RECEPTIONIST
	504	TANVIR	GULSHAN	1515112153	01-SEP-95	5000	JUNIOR RECEPTIONIST
	505	ANIKA	BADDA	1811100053	21-FEB-98	8000	SENIOR RECEPTIONIST

row(s) 1 - 5 of 5

4. NURSE TABLE

EDIT	N_ID	N_NAME	N_ADDRESS	N_CONTACT_NO	N_JOINING_DATE	N_SALARY	N_DESIGNATION
	401	TAMANNA	UTTARA	1811112153	21-SEP-80	5000	JUNIOR NURSE
	402	DIA	MIRPUR	1822012153	21-JUN-80	8000	SENIOR NURSE
	403	PROJNA	BANANI	1701112153	21-JUL-90	5000	JUNIOR NURSE
	404	ANTARA	GULSHAN	1515112153	01-SEP-95	5000	JUNIOR NURSE
	405	DOLLON	BADDA	1811100053	21-FEB-98	8000	SENIOR NURSE
							row(s) 1 - 5 of 5

5. ROOM TABLE

EDIT	ROOM_NO	ROOM_TYPE	ROOM_COST
	1	NON AC	1000
	2	NON AC	1000
	3	NON AC	1000
	4	AC	1500
	5	AC	1500
	6	AC	1500
	7	DULEX	2500
	8	DULEX	2500
	9	DULEX	2500
			row(s) 1 - 9 of 9

6. ASSIGNED_ROOM TABLE

EDIT	AR_ID	P_ID	ROOM_NO
	701	101	5
	702	103	6
	703	104	7
	704	106	8
	705	107	2
			row(s) 1 - 5 of 5

7. ASSIST_PATIENT TABLE

EDIT	AP_ID	P_ID	R_ID
	801	106	501
	802	101	501
	803	107	502
	804	103	503
	805	105	504
	806	102	505
	807	104	504

row(s) 1 - 7 of 7

8. ATTEND_PATIENT TABLE

EDIT	ATP_ID	P_ID	D_ID
	201	106	301
	202	101	301
	203	107	302
	204	103	303
	205	105	304
	206	102	305

row(s) 1 - 6 of 6

9. GOVERNS_ROOM TABLE

EDIT	GR_ID	N_ID	ROOM_NO
	901	401	1
	902	402	2
	903	403	3
	904	402	4
	905	403	5
	906	404	6
	907	405	7
	908	405	8
	909	403	9
row(s) 1 - 9 of 9			

SQL

1. **Write a query to find which patient is assigned to which room.**

```
SELECT P.P_NAME, R.ROOM_NO
FROM PATIENT P, ROOM R, ASSIGNED_ROOM A
WHERE A.P_ID=P.P_ID AND A.ROOM_NO=R.ROOM_NO;
```

2. **Find the receptionist with the maximum salary.**

```
SELECT R_NAME, R_SALARY
FROM RECEPTIONIST
WHERE R_SALARY= (SELECT MAX (R_SALARY)
                 FROM RECEPTIONIST;);
```

3. **Write a query to display the room type and their rounded average cost for all rooms as room type wise.**

```
SELECT ROOM_TYPE, ROUND (AVG (ROOM_COST))
FROM ROOM
GROUP BY ROOM_TYPE;
```

4. **List the highest paid doctor joined before the most recently hired doctor whose designation is assistant professor.**

```
SELECT * FROM DOCTOR
WHERE D_SALARY= (SELECT MAX (D_SALARY)
          FROM DOCTOR;)
AND D_JOINING_DATE> (SELECT MIN (D_JOINING_DATE)
             FROM DOCTOR;)
AND D_DESIGNATION='ASSISTANT PROFESSOR';
```

5. **Write a query to display the nurse, which patient they are governing and in which room.**

```
SELECT N.N_NAME, G.ROOM_NO, P.P_NAME
FROM NURSE N, GOVERNS_ROOM G, PATIENT P, ASSIGNED_ROOM A
WHERE G.N_ID=N.N_ID AND G.ROOM_NO=A.ROOM_NO AND A.P_ID=P.P_ID;
```

6. **Write a query to display doctor and nurse and which patient they are attending.**

```
SELECT D.D_NAME, N.N_NAME, P.P_NAME
FROM DOCTOR D, PATIENT P, NURSE N, ATTEND_PATIENT AP, ASSIGNED_ROOM AR,
GOVERNS_ROOM G
WHERE D.D_ID=AP.D_ID AND P.P_ID=AP.P_ID AND AP.P_ID=AR.P_ID AND
AR.ROOM_NO=G.ROOM_NO AND G.N_ID=N.N_ID;
```

7. **Find which receptionist assisted the highest number of patients.**

```
SELECT R.R_ID
FROM RECEPTIONIST R, ASSIST_PATIENT AP
WHERE R.R_ID=AP.R_ID HAVING COUNT (*) = (SELECT MAX (COUNT (*))
                                         FROM ASSIST_PATIENT GROUP BY R_ID ;)
GROUP BY R.R_ID;
```

8. **Find the number of doctor in the hospital.**

```
SELECT COUNT (*) FROM DOCTOR;
```

9. **Find the most experienced doctor in the hospital.**

```
SELECT D_NAME
FROM DOCTOR
WHERE D_JOINING_DATE= (SELECT MAX (D_JOINING_DATE) FROM DOCTOR;);
```

10. **Find the nurse whose salary is same as PROJNA.**

```
SELECT N_NAME
FROM NURSE
WHERE N_SALARY= (SELECT N_SALARY FROM NURSE WHERE N_NAME='PROJNA';);
```

VIEWS

1. **Create a view of patient name and contact number those are admitted.**

```
CREATE VIEW PATIENT_VIEW
AS SELECT P_NAME, P_CONTACT_NO
FROM PATIENT
WHERE P_STATUS='In Door';
```

2. **Create view of doctors to see their name, salary and designation.**

```
CREATE VIEW DOCTOR_VIEW
AS SELECT D_NAME, D_SALARY, D_DESIGNATION
FROM DOCTOR;
```

3. **Create view of nurse to see her name, salary and address who lives in BADDA**

```
CREATE VIEW NURSE_VIEW
AS SELECT N_NAME, N_SALARY, N_ADDRESS
FROM NURSE
WHERE N_ADDRESS='BADDA';
```

4. **Create View to display the nurse, which the patient they are governing and in which room.**

```
CREATE VIEW NURSE_ROOM_PATIENT_VIEW
AS SELECT N.N_NAME, G.ROOM_NO, P.P_NAME
FROM NURSE N, GOVERNS_ROOM G, PATIENT P, ASSIGNED_ROOM A
WHERE G.N_ID=N.N_ID AND G.ROOM_NO=A.ROOM_NO AND A.P_ID=P.P_ID;
```

5. **Create view to see which receptionist assisted the highest number of patient.**

```
CREATE VIEW RECEPTIONIST_PATIENT_VIEW
AS SELECT R.R_ID
FROM RECEPTIONIST R, ASSIST_PATIENT AP
WHERE R.R_ID=AP.R_ID HAVING COUNT (*) = (SELECT MAX (COUNT (*))
                                         FROM ASSIST_PATIENT GROUP BY R_ID)
GROUP BY R.R_ID;
```

6. **Create a view to see the most experienced doctor.**

```
CREATE VIEW DOCTOR_EXPERIENCE_VIEW
AS SELECT D_NAME
FROM DOCTOR
WHERE D_JOINING_DATE = (SELECT MAX (D_JOINING_DATE) FROM DOCTOR);
```