



American International University- Bangladesh

**ADVANCE DATABASE MANAGEMENT SYSTEM
SECTION: B
GROUP NAME: ROYAL BLUE**

**Mid-Term Project
Fall 20-21**

**Project Topic:
Hospital Management System**

With respectfully submitted to Rezwan Ahmed

Student Name	Student ID
DIN MOHAMMAD DOHAN	17-34465-2
SHAILY SARKER	17-35951-3
SAMIA ISLAM OYSE	18-37543-1
SAYED MAHMUDUL HASSAN	18-38095-2

2. Table of Contents

1. System Summary
2. Entity Relationship Diagram
3. Class Diagram
4. Use case Diagram
5. Schema Diagram
6. Activity Diagram
7. Examples of Sample Data
8. Use in Certain Database Scenarios
9. Login page With Database Connection

3. System Summary:

The hospital management system is served simplify the work of healthcare professionals as well as their interactions with patients. Its vast application and in real time, it involves many users and maintaining database including information of all the patients, all the hospital employee's including the doctors, the receptionist and the nurses. The hospital Management System can be registration a username and password. It is accessible only admin. Admin can add, delete and update data into the database. The data can be retrieved easily. The interface is very user friendly. The data are well protected for personal use and makes the data processing very fast.

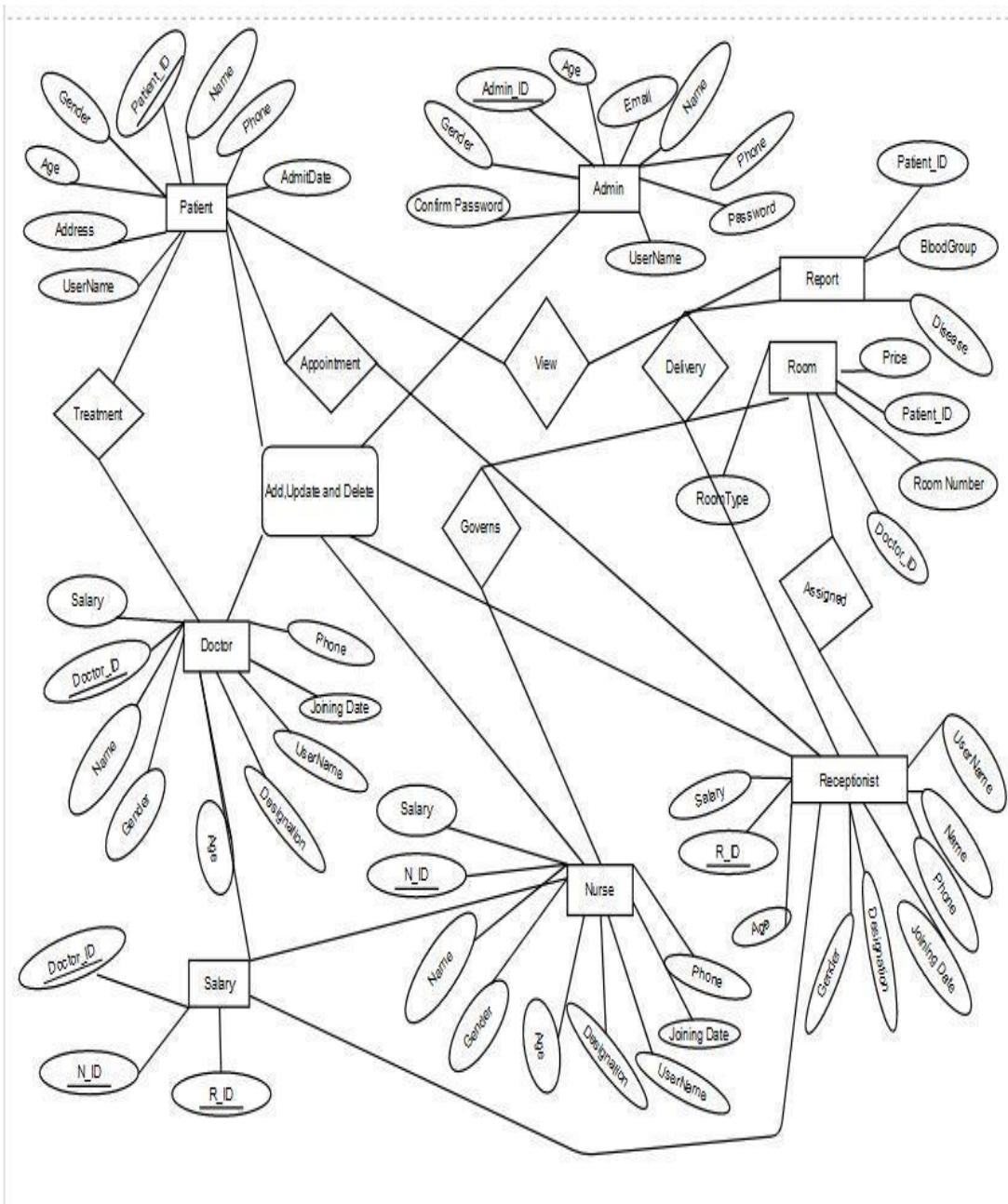
The software has the facility to each patient and employee who registration the form will be assigned a particular unique id available on the system and the receptionist can access the database to help the visitors by retrieving all the information about the patients and help them out in the best way. They can access the doctor information like the name of the doctor and their availability etc.

It includes a search facility to know the current status of each room. The receptionist will be responsible for managing the appointments with the doctor. Also, the receptionist needs to keep track about the information of all the patients report. User can search availability of a doctor and the details of a patient using the id.

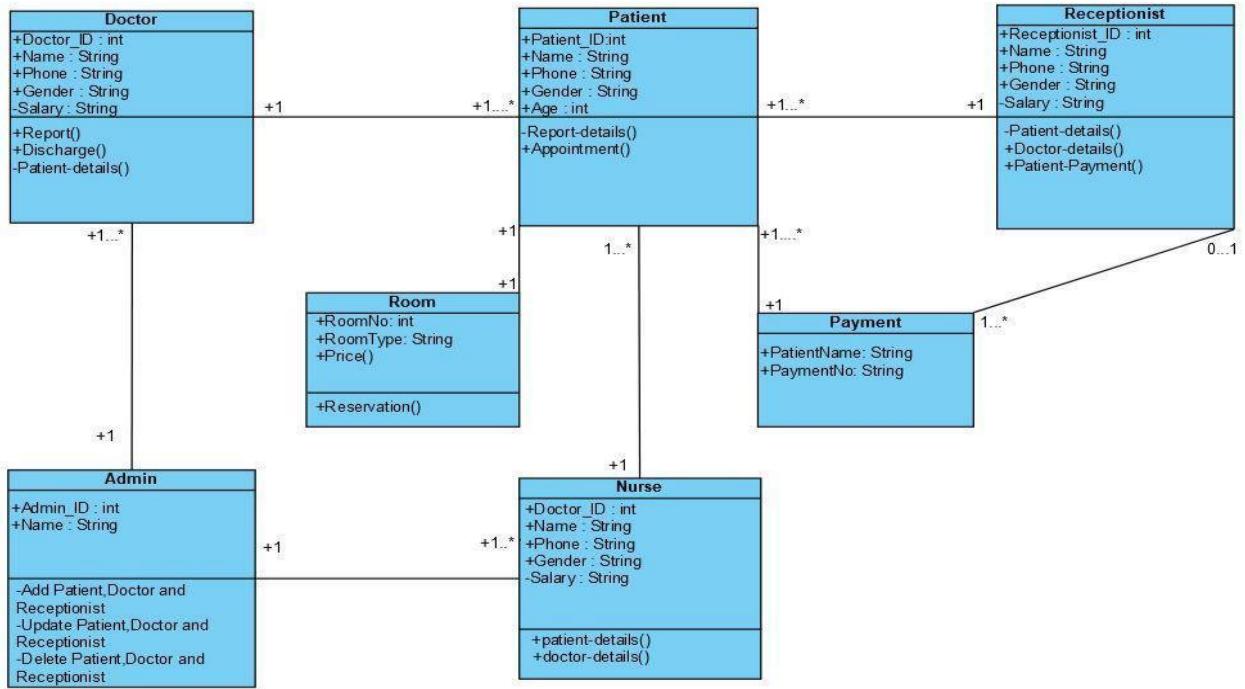
There are uses in the system. The user groups involved with respect to hospital database are:

- Admin
- Doctor
- Nurse
- Receptionist
- Patient

4. ER Diagram:

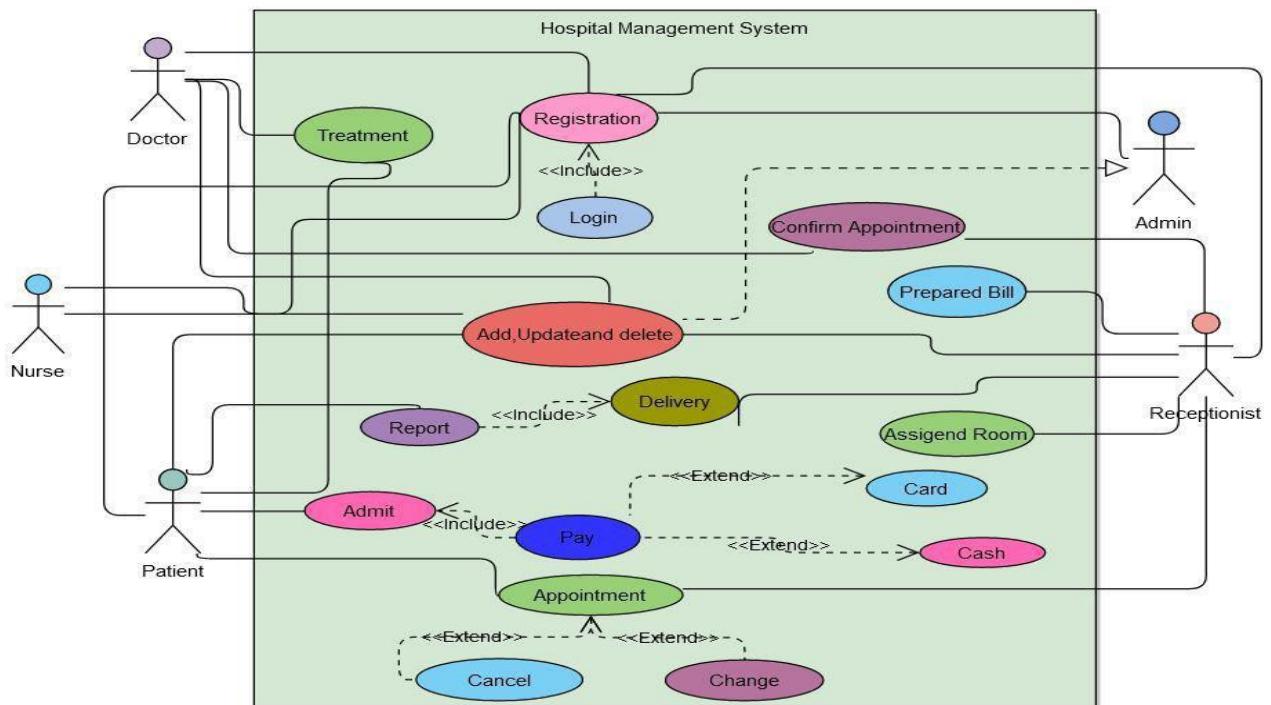


5. Class Diagram:



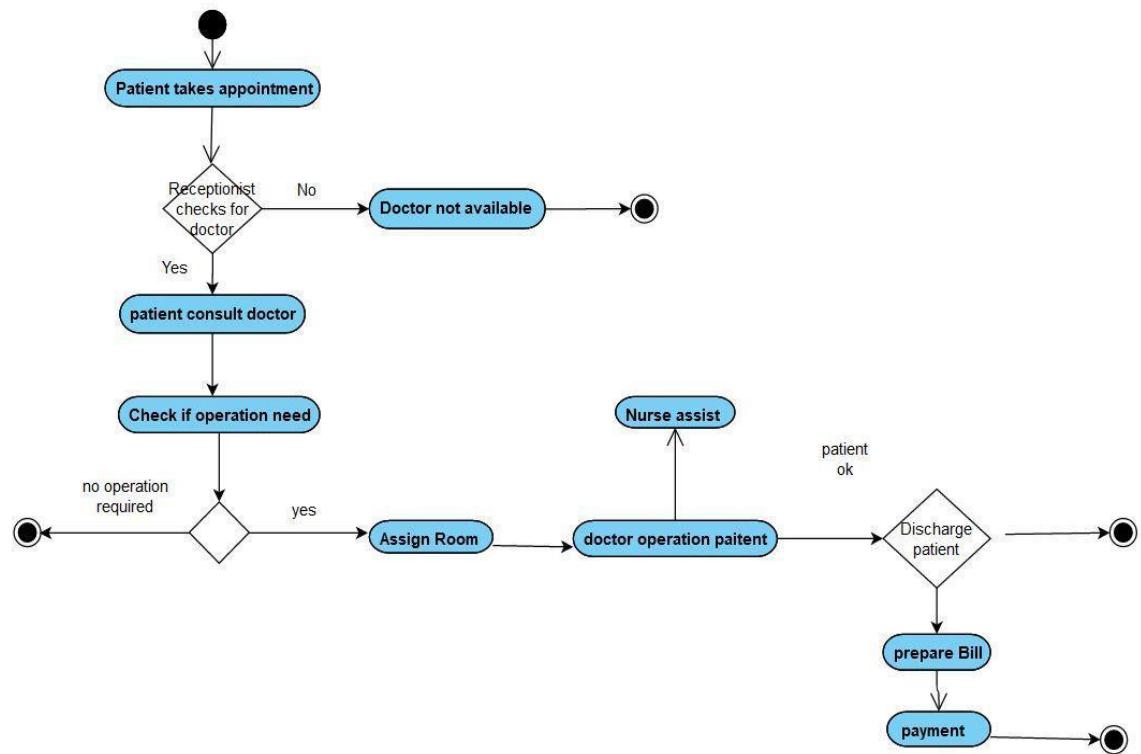
6

I. Use case Diagram:

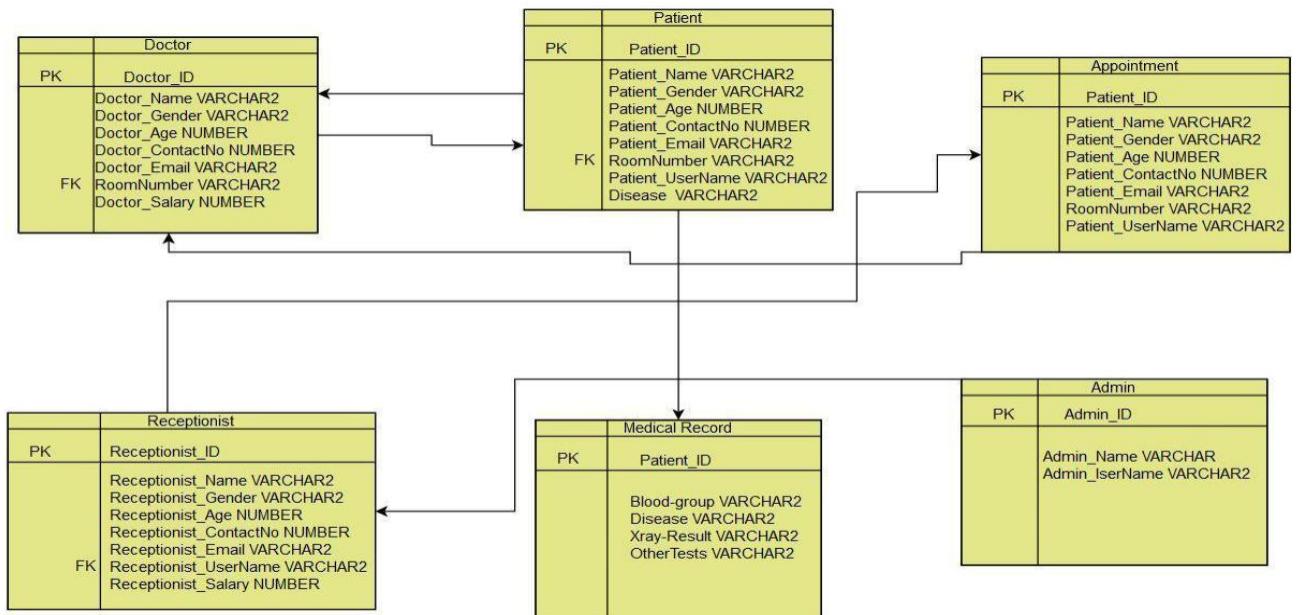


6.

II. Activity Diagram



7. Schema Diagram:



8. Table Creation & Value Insertion:

CREATE TABLE ADMIN

```
(Admin_ID NUMBER (10) PRIMARY KEY,  
Admin_Name VARCHAR2 (50) NOT NULL,  
Admin_Gender VARCHAR2 (10) NOT NULL,  
Admin_Age NUMBER (10) NOT NULL,  
Admin_ContactNo NUMBER (20) NOT NULL,  
Admin_UserName VARCHAR2 (50) NOT NULL);
```

ADMIN Value Insertion :

```
insert into ADMIN values(101,'Sakibul Khan','Male',28,'01745665555','xyz3'); insert  
into ADMIN values(102,'Hasan Habib','Male',31,'01345665789','hasm'); insert into  
ADMIN values(201,'Hayat Islam','Female',28,'01845665111','m4kl'); insert into  
ADMIN values(211,'Ayain Rahman','Male',30,'01915664548','qop1'); insert into  
ADMIN values(235,'Sakib Iskam','Male',25,'01715664545','wzq7'); insert into ADMIN  
values(303,'Puja Hawladar','Female',30,'01745667878','7hyx');
```

CREATE TABLE DOCTOR

```
(Doctor_ID NUMBER (10) PRIMARY KEY  
Doctor_Name VARCHAR2(50) NOT NULL,  
Doctor_Gender VARCHAR2 (10) NOT NULL,  
Doctor_Age NUMBER (10) NOT NULL,  
Doctor_ContactNo NUMBER(11) NOT NULL,  
Doctor_JoiningDate DATE NOT NULL,  
Department_Name VARCHAR(100) NOT NULL,  
Doctor_Designation VARCHAR2(100) NOT NULL,  
Doctor_Salary NUMBER (10) NOT NULL);
```

DOCTOR Value Insertion :

INSERT INTO DOCTOR

VALUES (1,'Dr.Saifulla Khan','Male',50,01821212,to_date('05-FEB-95','DD-MM-YY'),'Cardiology', 'Assistant ',50000);

INSERT INTO DOCTOR

VALUES (2,'Dr.Rashed Islam','Male',26,01721212,to_date('05-MAR-92','DD-MM-YY'),'Diagnostic','professior',60000);

INSERT INTO DOCTOR

VALUES (3,'Dr.Ripa','Female',27,01321212,to_date('05-Nov-95','DD-MM-YY'),'Orthopedics','Assistant professor',45000);

INSERT INTO DOCTOR

VALUES (4,'Dr.Seemaa','Female',27,011721212, to_date('13-jan-99','DD-MM-YY'),'Neurology','professior',65000);

INSERT INTO DOCTOR

VALUES (5,'Dr. Uzzal Kumar','Male',29,0136255212, to_date('13-Nov-99','DD-MM-YY'),'Neurology', 'assistant professor',55000);

CREATE TABLE PATIENT

(Patient_ID NUMBER (10) PRIMARY KEY,
Patient_Name VARCHAR2 (50) NOT NULL,
Patient_Gender VARCHAR2 (10) NOT NULL,
Patient_Age NUMBER (10) NOT NULL,
Patient_ContactNo NUMBER (20) NOT NULL,
Patient_UserName VARCHAR2 (50) NOT NULL,
Patient_Address VARCHAR2(100) NOT NULL,
AdmitDate VARCHAR2 (20) NOT NULL,
Department_Name VARCHAR2 (20) NOT NULL);

PATIENT Value Insertion:

INSERT INTO PATIENT VALUES(1,'Samia','Female',22,01521322246,'Samia12','Badda','21-july-2020','Cardiology')

INSERT INTO PATIENT VALUES(2,'Shoily','Female',21,0152223246,'Soily12','Mirpur-1','22-june-2020','Diagnostic')

INSERT INTO PATIENT VALUES (3,'Srity','Female',22,0152223246,'Srity44','Nilkhet ','21-Nov-2020','Orthopedics')

```
INSERT INTO PATIENT VALUES(4,'dohan','male',25,0162223246,'dohan44','Noakhali','22-june-2020','Neurology')

INSERT INTO PATIENT VALUES(5,'sifat','male',28,01655223246,'sify4','khilkhet','23-april-2020','Orthopedic')

INSERT INTO PATIENT VALUES (6,'saiful','male',25,01655223246,'sif000','Old Dhaka','22-april-2020','Cardiology')
```

CREATE TABLE NURSE

```
(Nurse_ID NUMBER (10) PRIMARY KEY  
Nurse_Name VARCHAR2(50) NOT NULL,  
Nurse_Gender VARCHAR2 (10) NOT NULL,  
Nurse_Age NUMBER (10) NOT NULL,  
Nurse_ContactNo NUMBER (11) NOT NULL,  
Nurse_JoiningDate DATE NOT NULL,  
Department_Name VARCHAR (100) NOT NULL,  
Nurse_Designation VARCHAR2(100) NOT NULL,  
Nurse_Salary NUMBER (10) NOT NULL);
```

NURSE Value Insertion:

```
insert into NURSE values (1,'Anika','Female',22,'01612842314', to_date('05-FEB-1995','DD-MM-YY'),'Cardiology', 'Junior Nurse');  
insert into NURSE values (2,'Keya','Female',21,'01712842777', to_date('08-NOV-2003','DD-MM-YY'), 'Diagnostic', 'Senior Nurse');  
insert into NURSE values (3,'Nur','Male',28,'01612843535', to_date('11-MAR-2008','DD-MM-YY'), 'Orthopedics', 'Junior Nurse');  
insert into NURSE values (4,'Liya','Female',22,'01912942121', to_date('01-FEB-1999','DD-MM-YY'), 'Cardiology', 'Senior Nurse');  
insert into NURSE values (5,'Joy','Male',23,'01812842314', to_date('05-JAN-1999','DD-MM-YY'), 'Diagnostic', 'Junior Nurse');  
insert into NURSE values (6,'Nil','Male',22,'01922842888', to_date('05-FEB-2002','DD-MM-YY'), 'Neurology', 'Nurse');  
insert into NURSE values (7,'Anu','Female',22,'01612842555', to_date('05-DEC-1999','DD-MM-YY'), 'Cardiology', 'Nurse');
```

CREATE TABLE RECEPTIONIST

```
(Receptionist_ID NUMBER (10) PRIMARY KEY,
```

```
Receptionist_Name VARCHAR2 (50) NOT NULL,  
Receptionist_Gender VARCHAR2 (10) NOT NULL,  
Receptionist_Age NUMBER (10) NOT NULL,  
Receptionist_UserName VARCHAR2 (50) NOT NULL,  
Receptionist_JoiningDate DATE NOT NULL,  
Receptionist_Designation VARCHAR2(50) NOT NULL,  
Receptionist_Salary NUMBER (20) NOT NULL;
```

RECEPTIONIST Value Insertion:

```
INSERT INTO RECEPTIONIST  
VALUES (1,'jasmin','female',50,'jasmin12', to_date('08-NOV-2003','DD-MM-YY'),'Senior ',18000);  
INSERT INTO RECEPTIONIST  
VALUES (2,'ira','female',23,'ira123', to_date('07-NOV-2004','DD-MM-YY'),'Junior',12000);  
INSERT INTO RECEPTIONIST  
VALUES (3,'ikraam','male',25,'ikram123', to_date('04-AUG-2004','DD-MM-YY'),'junior',12000);  
INSERT INTO RECEPTIONIST  
VALUES (4,'dohan','male',30,'dohan234', to_date('04-AUG-2005','DD-MM-YY'),'Senior',19000);  
INSERT INTO RECEPTIONIST  
VALUES (5,'sajib','male',22,'sajib234', to_date('05-DEC-2007','DD-MM-YY'),'Junior',11000);
```

CREATE TABLE REPORT

```
(Report_NO NUMBER (10),  
Patient_ID NUMBER (10) NOT NULL,  
Doctor_ID NUMBER (10) NOT NULL,  
Patient_BloodGroup VARCHAR2 (20) NOT NULL,  
Patient_Disease VARCHAR2 (30) NOT NULL,  
PRIMARY KEY (Patient_ID ,Doctor_ID,Report_NO));
```

REPORT Value Insertion:

```
Insert into REPORT values (101,1,1,'A+', 'Heart Probleam');
```

```
Insert into REPORT values (102,2,2,'B+', 'Diabetics'); Insert  
into REPORT values (103,3,3,'A-', 'knee surgery'); Insert into  
REPORT values (104,4,4,'o+', 'Neuro diseas'); Insert into  
REPORT values (105,5,5,'A+', 'Brain Tumour'); Insert into  
REPORT values (106,6,1,'AB+', 'Heart Probleam');
```

CREATE TABLE ROOM

```
(Room_NO NUMBER (10) ,  
Patient_ID NUMBER (10),  
Nurse_ID NUMBER (10) ,  
Doctor_ID NUMBER (10) ,  
PRIMARY KEY(Patient_ID,Nurse_ID,Doctor_ID ));
```

ROOM Value Insertion:

```
Insert into ROOM values (300,1,2,3);  
Insert into ROOM values (300,2,1,3);  
Insert into ROOM values (301,3,4,1);  
Insert into ROOM values (302,4,5,1);  
Insert into ROOM values (302,5,6,5);  
Insert into ROOM values (303,6,7,2);
```

CREATE TABLE DEPARTMENT

```
(Department_ID NUMBER (10) PRIMARY KEY,  
Department_Name VARCHAR2 (14) FOREIGN KEY);
```

DEPERTMENT Value Insertion:

```
Insert into DEPARTMENT values (7823, 'Cardiology');  
Insert into DEPARTMENT values (9123, 'Diagnostic');  
Insert into DEPARTMENT values (4125, 'Orthopedics');  
Insert into DEPARTMENT values (9783, 'Neurology');
```

9. Screen shots of sample data:

```
CREATE TABLE ADMIN
(Admin_ID NUMBER (10) PRIMARY KEY,
Admin_Name VARCHAR2 (50) NOT NULL,
Admin_Gender VARCHAR2 (10) NOT NULL,
Admin_Age NUMBER (10) NOT NULL,
Admin_ContactNo NUMBER (20) NOT NULL,
Admin_UserName VARCHAR2 (50) NOT NULL);
```

Results Explain Describe Saved SQL History

Table created.

0.02 seconds

```

insert into ADMIN values(101,'Sakibul Khan','Male',28,'01745665555','xyz3');
insert into ADMIN values(102,'Hasan Habib','Male',31,'01345665789','hasm');
insert into ADMIN values(201,'Hayat Islam','Female',28,'01845665111','m4kl');
insert into ADMIN values(211,'Ayain Rahman','Male',30,'01915664548','qop1');
insert into ADMIN values(235,'Sakib Iskam','Male',25,'01715664545','wzq7');
insert into ADMIN values(303,'Puja Hawladar','Female',30,'01745667878','7hyx');

select * from admin

```

Results Explain Describe Saved SQL History

ADMIN_ID	ADMIN_NAME	ADMIN_GENDER	ADMIN AGE	ADMIN_CONTACTNO	ADMIN_USERNAME
101	Sakibul Khan	Male	28	1745665555	xyz3
102	Hasan Habib	Male	31	1345665789	hasm
201	Hayat Islam	Female	28	1845665111	m4kl
211	Ayain Rahman	Male	30	1915664548	qop1
235	Sakib Iskam	Male	25	1715664545	wzq7
303	Puja Hawladar	Female	30	1745667878	7hyx

6 rows returned in 0.00 seconds [Download](#)

```

CREATE TABLE NURSE
(Nurse_ID NUMBER (10) PRIMARY KEY,
Nurse_Name VARCHAR2(50) NOT NULL,
Nurse_Gender VARCHAR2 (10) NOT NULL,
Nurse_Age NUMBER (10) NOT NULL,
Nurse_ContactNo NUMBER(11) NOT NULL,
Nurse_JoiningDate DATE NOT NULL,
Department_Name VARCHAR(100) NOT NULL,
Nurse_Designation VARCHAR2(100) NOT NULL,
Nurse_Salary NUMBER (10) NOT NULL);

```

Results Explain Describe Saved SQL History

Table created.

```

insert into NURSE values(1,'Anika','Female',22,01612842314,to_date('05-FEB-1995','DD-MM-YY'),'Cardiology', 'Junior Nurse',13000);
insert into NURSE values(2,'Keya','Female',21,01712842777, to_date('08-NOV-2003','DD-MM-YY'),'Diagnostic', 'Senior Nurse',18000);
insert into NURSE values(3,'Nur','Male',28,01612843535, to_date('11-MAR-2008','DD-MM-YY'),'Orthopedics', 'Junior Nurse',16000);
insert into NURSE values(4,'Liya ','Female',22,01912942121, to_date('01-FEB-1999','DD-MM-YY'),'Cardiology', 'Senior Nurse',12000);
insert into NURSE values(5,'Joy ','Male',23,01812842314, to_date('05-JAN-1999','DD-MM-YY'),'Diagnostic', 'Junior Nurse',13000);
insert into NURSE values(6,'Nil ','Male',22,01922842888, to_date('05-FEB-2002','DD-MM-YY'),'Neurology', 'Nurse',10000);
insert into NURSE values(7,'Anu ','Female',22,01612842555, to_date('05-DEC-1999','DD-MM-YY'),'Cardiology', 'Nurse',10000);

select * from nurse

```

[Results](#) [Explain](#) [Describe](#) [Saved SQL](#) [History](#)

NURSE_ID	NURSE_NAME	NURSE_GENDER	NURSE_AGE	NURSE_CONTACTNO	NURSE_JOININGDATE	DEPARTMENT_NAME	NURSE_DESIGNATION	NURSE_SALARY
1	Anika	Female	22	1612842314	02/05/1995	Cardiology	Junior Nurse	13000
2	Keya	Female	21	1712842777	11/08/2003	Diagnostic	Senior Nurse	18000
3	Nur	Male	28	1612843535	03/11/2008	Orthopedics	Junior Nurse	16000
4	Liya	Female	22	01912942121	02/01/1999	Cardiology	Senior Nurse	12000
5	Joy	Male	23	1812842314	01/05/1999	Diagnostic	Junior Nurse	13000
6	Nil	Male	22	1922842888	02/05/2002	Neurology	Nurse	10000
7	Anu	Female	22	1612842555	12/05/1999	Cardiology	Nurse	10000

7 rows returned in 0.00 seconds [Download](#)

```

CREATE TABLE DOCTOR
(Doctor_ID NUMBER (10) PRIMARY KEY,
Doctor_Name VARCHAR2(50) NOT NULL,
Doctor_Gender VARCHAR2 (10) NOT NULL,
Doctor_Age NUMBER (10) NOT NULL,
Doctor_ContactNo NUMBER(11) NOT NULL,
Doctor_JoiningDate DATE NOT NULL,
Department_Name VARCHAR(100) NOT NULL,
Doctor_Designation VARCHAR2(100) NOT NULL,
Doctor_Salary NUMBER (10) NOT NULL);

```

[Results](#) [Explain](#) [Describe](#) [Saved SQL](#) [History](#)

Table created.

```

INSERT INTO DOCTOR
VALUES (1,'Dr.Saifulla Khan','Male',50,01821212,to_date('05-FEB-95','DD-MM-YY'),'Cardiology', 'Assistant ',50000);
INSERT INTO DOCTOR
VALUES (2,'Dr.Rashed Islam','Male',26,01721212,to_date('05-MAR-92','DD-MM-YY'),'Diagnostic','professor',60000);
INSERT INTO DOCTOR
VALUES (3,'Dr.Ripa','Female',27,01321212,to_date('05-Nov-95','DD-MM-YY'),'Orthopedics', 'Assistant professor',45000);
INSERT INTO DOCTOR
VALUES (4,'Dr.Seemaa','Female',27,011721212,to_date('13-jan-99','DD-MM-YY'),'Neurology','professor',65000);
INSERT INTO DOCTOR
VALUES (5,'Dr.Uzzal Kumar','Male',29,0136255212,to_date('13-Nov-99','DD-MM-YY'), 'Neurology', 'assistant professor',55000);

Select * From DOCTOR

```

[Results](#) [Explain](#) [Describe](#) [Saved SQL](#) [History](#)

DOCTOR_ID	DOCTOR_NAME	DOCTOR_GENDER	DOCTOR AGE	DOCTOR_CONTACTNO	DOCTOR_JOININGDATE	DEPARTMENT_NAME	DOCTOR_DESIGNATION	DOCTOR_SALARY
1	Dr.Saifulla Khan	Male	50	1821212	02/05/2095	Cardiology	Assistant	50000
2	Dr.Rashed Islam	Male	26	1721212	03/05/2092	Diagnostic	professor	60000
3	Dr.Ripa	Female	27	1321212	11/05/2095	Orthopedics	Assistant professor	45000
4	Dr.Seemaa	Female	27	11721212	01/13/2099	Neurology	professor	65000
5	Dr.Uzzal Kumar	Male	29	136255212	11/13/2099	Neurology	assistant professor	55000

5 rows returned in 0.01 seconds

[Download](#)

```

CREATE TABLE RECEPTIONIST
(Receptionist ID NUMBER (10) PRIMARY KEY,
Receptionist Name VARCHAR2 (50) NOT NULL,
Receptionist Gender VARCHAR2 (10) NOT NULL,
Receptionist Age NUMBER (10) NOT NULL,
Receptionist UserName VARCHAR2 (50) NOT NULL,
Receptionist JoiningDate DATE NOT NULL,
Receptionist Designation VARCHAR2(50) NOT NULL,
Receptionist Salary NUMBER(20) NOT NULL);

```

[Results](#) [Explain](#) [Describe](#) [Saved SQL](#) [History](#)

Table created.

0.01 seconds

```

INSERT INTO RECEPTIONIST
VALUES (1,'jasmin','female',50,'jasmin12',to_date('08-NOV-2003','DD-MM-YY'),'Senior ',18000);
INSERT INTO RECEPTIONIST
VALUES (2,'ira ','female',23,'ira123',to_date('07-NOV-2004','DD-MM-YY'),'Junior',12000);
INSERT INTO RECEPTIONIST
VALUES (3,'ikram ','male',25,'ikram123',to_date('04-AUG-2004','DD-MM-YY'),'junior',12000);
INSERT INTO RECEPTIONIST
VALUES (4,'dohan ', 'male',30,'dohan234',to_date('04-AUG-2005','DD-MM-YY'),'Senior',19000);
INSERT INTO RECEPTIONIST
VALUES (5,'sajib ', 'male',22,'sajib234',to_date('05-DEC-2007','DD-MM-YY'),'Junior',11000);

Select * from RECEPTIONIST

```

[Results](#) [Explain](#) [Describe](#) [Saved SQL](#) [History](#)

RECEPTIONIST_ID	RECEPTIONIST_NAME	RECEPTIONIST_GENDER	RECEPTIONIST AGE	RECEPTIONIST_USERNAME	RECEPTIONIST_JOININGDATE	RECEPTIONIST_DESIGNATION	RECEPTIONIST_SALARY
1	jasmin	female	50	jasmin12	11/08/2003	Senior	18000
2	ira	female	23	ira123	11/07/2004	Junior	12000
3	ikram	male	25	ikram123	08/04/2004	junior	12000
4	dohan	male	30	dohan234	08/04/2005	Senior	19000
5	sajib	male	22	sajib234	12/05/2007	Junior	11000

5 rows returned in 0.00 seconds [Download](#)

```

CREATE TABLE PATIENT
(Patient ID NUMBER (10) PRIMARY KEY,
Patient Name VARCHAR2 (50) NOT NULL,
Patient Gender VARCHAR2 (10) NOT NULL,
Patient Age NUMBER (10) NOT NULL,
Patient ContactNo NUMBER (20) NOT NULL,
Patient UserName VARCHAR2 (50) NOT NULL,
Patient Address VARCHAR2(100) NOT NULL,
AdmitDate VARCHAR2 (20) NOT NULL,
Department Name VARCHAR2 (20) NOT NULL);

```

[Results](#) [Explain](#) [Describe](#) [Saved SQL](#) [History](#)

Table dropped.

```

INSERT INTO PATIENT VALUES(1,'Samia','Female',22,01521322246,'Samia12','Badda','21-july-2020','Cardiology')
INSERT INTO PATIENT VALUES(2,'Shoily','Female',21,0152223246,'Soily12','Mirpur-1','22-june-2020','Diagnostic')
INSERT INTO PATIENT VALUES(3,'Srity','Female',22,0152223246,'Srity44','Nilkhet ','21-Nov-2020','Orthopedics')
INSERT INTO PATIENT VALUES(4,'dohan','male',25,0162223246,'dohan44','Noakhali','22-june-2020','Neurology')
INSERT INTO PATIENT VALUES(5,'sifat','male',28,01655223246,'sify4','khilkhet ',23-april-2020,'Orthopedic')
INSERT INTO PATIENT VALUES(6,'saiful','male',25,01655223246,'sif000','Old Dhaka',22-april-2020,'Cardiology')

select * from PATIENT

```

Results Explain Describe Saved SQL History									
PATIENT_ID	PATIENT_NAME	PATIENT_GENDER	PATIENT_AGE	PATIENT_CONTACTNO	PATIENT_USERNAME	PATIENT_ADDRESS	ADMITDATE	DEPARTMENT_NAME	
1	Samia	Female	22	1521322246	Samia12	Badda	21-july-2020	Cardiology	
2	Shoily	Female	21	152223246	Soily12	Mirpur-1	22-june-2020	Diagnostic	
3	Srity	Female	22	152223246	Srity44	Nilkhet	21-Nov-2020	Orthopedics	
4	dohan	male	25	162223246	dohan44	Noakhali	22-june-2020	Neurology	
5	sifat	male	28	1655223246	sify4	khilkhet	23-april-2020	Orthopedic	
6	saiful	male	25	1655223246	sif000	Old Dhaka	22-april-2020	Cardiology	

6 rows returned in 0.01 seconds [Download](#)

```

CREATE TABLE REPORT
(Report_NO NUMBER (10) ,
Patient_ID NUMBER (10) NOT NULL ,
Doctor_ID NUMBER (10) NOT NULL,
Patient_BloodGroup VARCHAR2 (20) NOT NULL,
Patient_Disease VARCHAR2 (30) NOT NULL,
PRIMARY KEY(Patient_ID ,Doctor_ID,Report_NO));

```

Results Explain Describe Saved SQL History

Table created.

```
Insert into REPORT values (101,1,1,'A+', 'Heart Problem');
Insert into REPORT values (102,2,2,'B+', 'Diabetics');
Insert into REPORT values (103,3,3,'A-', 'knee surgery');
Insert into REPORT values (104,4,4,'o+', 'Neuro diseas');
Insert into REPORT values (105,5,5,'A+', 'Brain Tumour');
Insert into REPORT values (106,6,1,'AB+', 'Heart Problem');

select * from report
```

[Results](#) [Explain](#) [Describe](#) [Saved SQL](#) [History](#)

REPORT_NO	PATIENT_ID	DOCTOR_ID	PATIENT_BLOODGROUP	PATIENT_DISEASE
101	1	1	A+	Heart Problem
102	2	2	B+	Diabetics
103	3	3	A-	knee surgery
104	4	4	o+	Neuro diseas
105	5	5	A+	Brain Tumour
106	6	1	AB+	Heart Problem

6 rows returned in 0.00 seconds [Download](#)

```
CREATE TABLE ROOM
(Room_NO NUMBER (10) ,
Patient_ID NUMBER (10),
Nurse_ID NUMBER (10) ,
Doctor_ID NUMBER (10) ,
PRIMARY KEY(Patient_ID,Nurse_ID,Doctor_ID ));
```

Results Explain Describe Saved SQL History

Table created.

```
Insert into ROOM values (300,1,2,3);
Insert into ROOM values (300,2,1,3);
Insert into ROOM values (301,3,4,1);
Insert into ROOM values (302,4,5,1);
Insert into ROOM values (302,5,6,5);
Insert into ROOM values (303,6,7,2);
```

```
select * from ROOM
```

Results Explain Describe Saved SQL History

ROOM_NO	PATIENT_ID	NURSE_ID	DOCTOR_ID
300	1	2	3
300	2	1	3
301	3	4	1
302	4	5	1
302	5	6	5
303	6	7	2

10. Ten Questions & Answers:

1. Display the Patient id, name, age if the age is above 22.

Answer: SELECT PATIENT_ID, PATIENT_NAME, PATIENT_GENDER FROM PATIENT WHERE PATIENT_AGE > 22;

2. Display Admin age, gender of Hasan Habib.

Answer: SELECT ADMIN_AGE, ADMIN_GENDER from ADMIN where ADMIN_NAME= 'Hasan Habib';

3. Concatenate doctor name and age and display room from doctor table.

Answer: SELECT CONCAT (DOCTOR_NAME, DOCTOR_AGE) from DOCTOR;

4. Display Patient ID, name and Doctor ID, name from patient, doctor table where room no is same.

Answer: select PATIENT.PATIENT_ID, PATIENT.PATIENT_NAME, DOCTOR.DOCTOR_ID, DOCTOR.DOCTOR_NAME, ROOM_NO from PATIENT, DOCTOR, ROOM Where ROOM.PATIENT_ID = ROOM.DOCTOR_ID;

5. Display patient Id and their doctor name, doctor salary and doctor Id.

Answer: Select DOCTOR_NAME, DOCTOR_SALARY, PATIENT_ID, DOCTOR_ID from DOCTOR, PATIENT Where PATIENT.DEPARTMENT_NAME= DOCTOR. DEPARTMENT_NAME;

6. List of the obedient Doctor who are the PROFESSION of the Cardiology Department.

Answer: SELECT DOCTOR_NAME from DOCTOR Where DEPARTMENT_NAME='Cardiology' and DOCTOR_DESIGNATION='PROFESSION';

7. Update nurse salary from nurse table which nurse age is 28, they will get extra 1000 with their salary.

Answer: UPDATE NURSE SET NURSE_SALARY = NURSE_SALARY + 1000 WHERE NURSE_AGE IN (SELECT NURSE_AGE FROM NURSE WHERE NURSE_AGE=28);

8. To display NAME, GENDER of the RECEPTIONISTS from RECEPTIONIST table whose Designation As 'Senior'.

Answer: Select RECEPTIONIST_ID, RECEPTIONIST_NAME, RECEPTIONIST_GENDER from RECEPTIONIST WHERE RECEPTIONIST_ID IN (SELECT RECEPTIONIST_ID from RECEPTIONIST where RECEPTIONIST_DESIGNATION ='Senior');

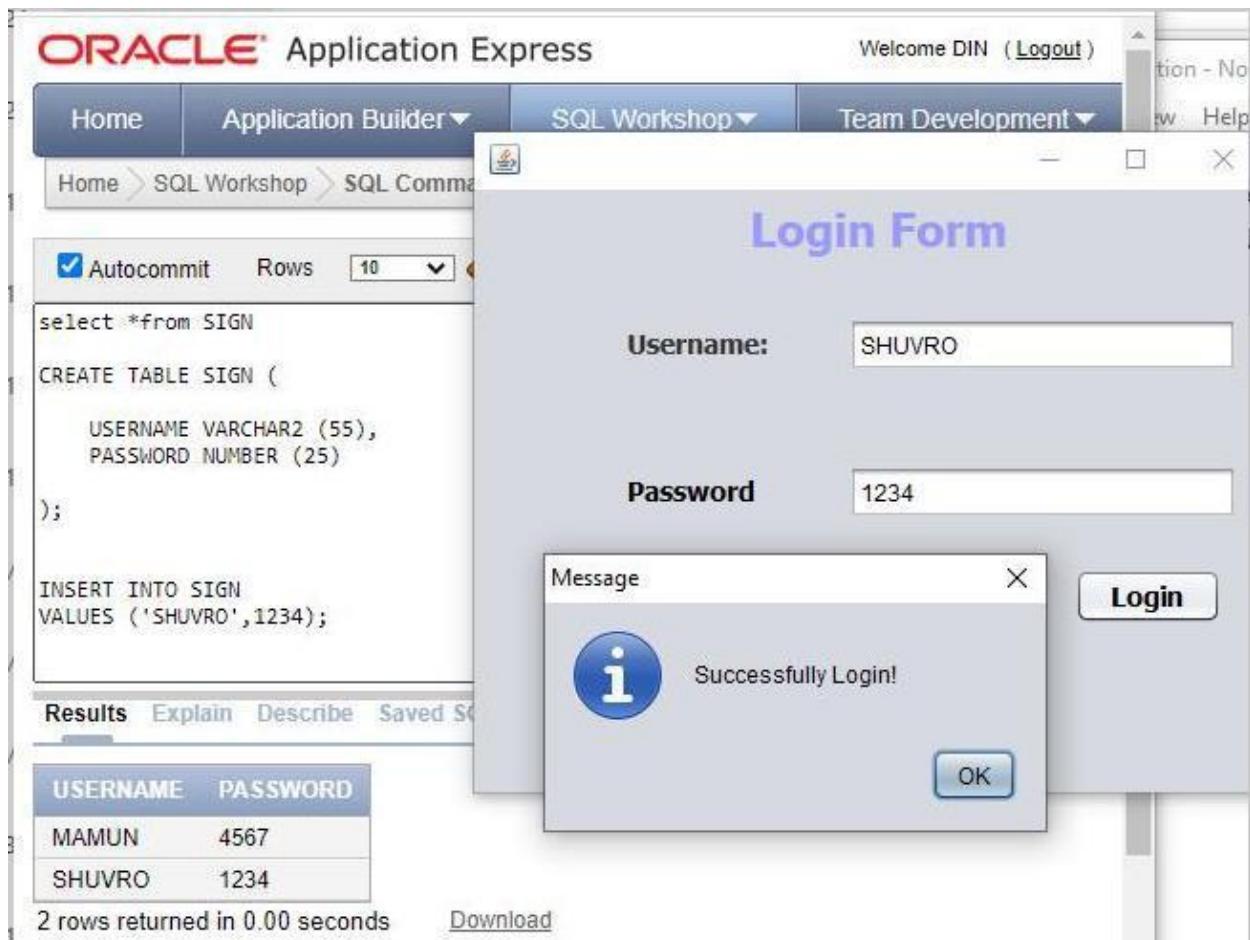
9. Display the list of Doctor which salary is greater than 50000 (In subquery).

Answer: Select * FROM DOCTOR WHERE DOCTOR_ID IN (SELECT DOCTOR_ID FROM DOCTOR WHERE DOCTOR_SALARY > 50000);

10. Display the list of nurse ID for each department by left joining.

Answer: Select D. DEPARTMENT_NAME, COUNT (N. NURSE_ID) AS numberofNurse FROM N. NURSE LEFT JOIN DEPARTMENT ON N. DEPARTMENT_NAME = D. DEPARTMENT_NAME GROUP BY DEPARTMENT_NAME;

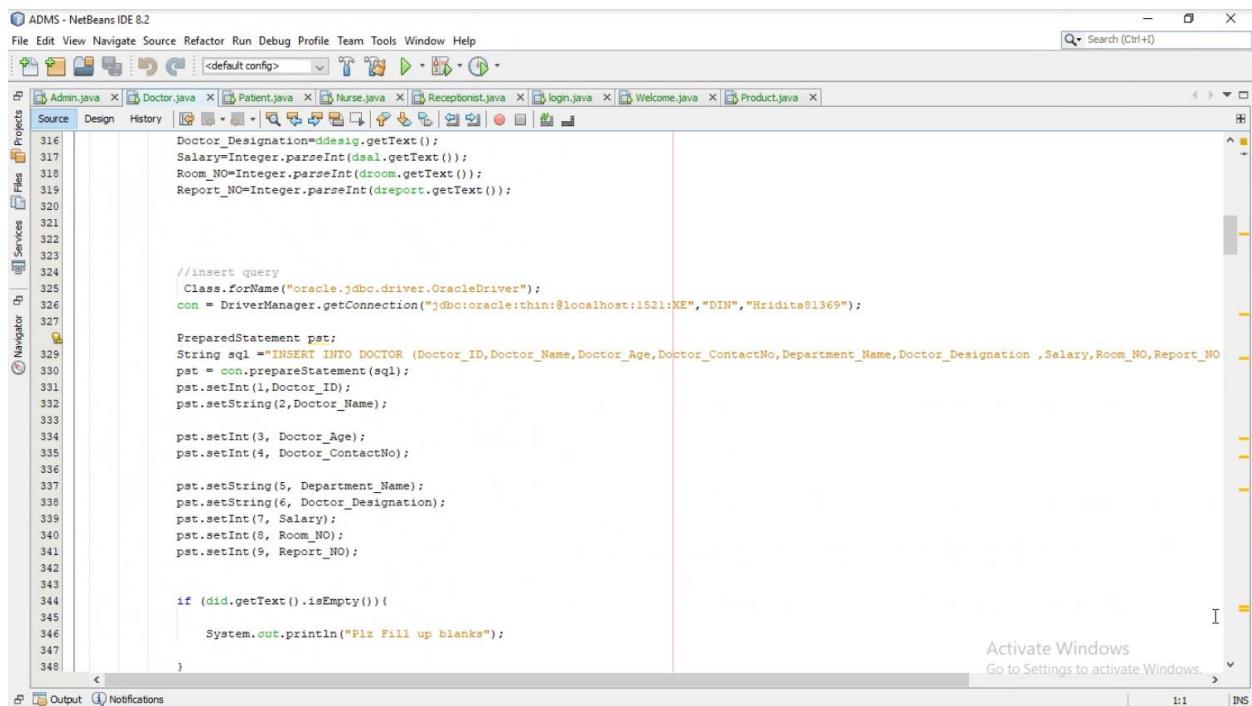
11. A simple User Interface for user authentication for successful login operation to the system by connecting oracle database using JAVA.



Final Project Deliverable

Question No 01:

Yes, Oracle database has been used successfully.

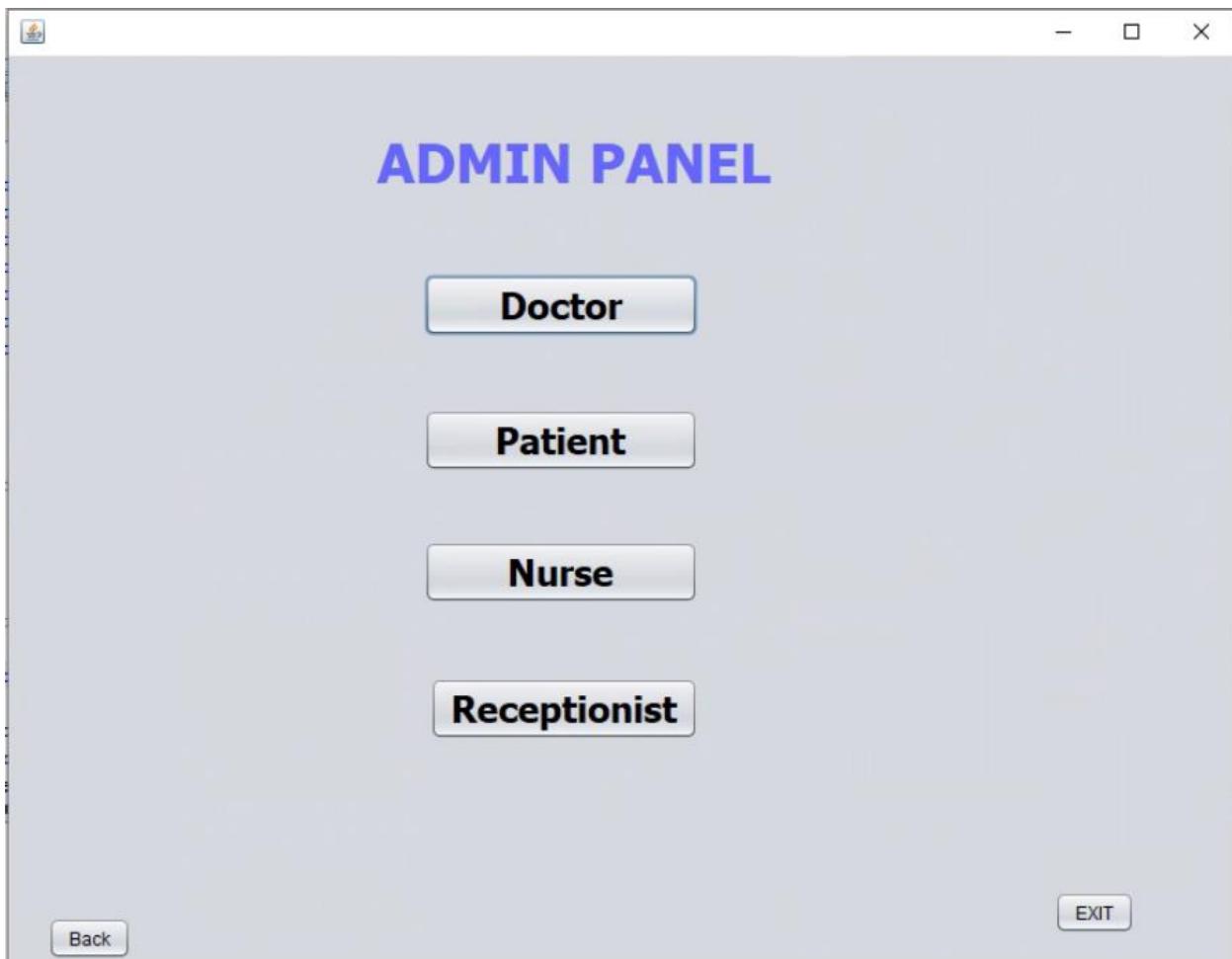


The screenshot shows the NetBeans IDE interface with the title "ADMS - NetBeans IDE 8.2". The main window displays a Java file named "Doctor.java" containing code for inserting data into an Oracle database. The code includes imports for `java.sql.*` and `oracle.jdbc.driver.*`, and uses `PreparedStatement` to execute an `INSERT INTO` query. The code also handles user input for doctor details and prints a message if any fields are empty. The IDE's toolbar, menu bar, and various tool windows like Projects, Files, Services, and Navigator are visible around the code editor.

```
316 Doctor_Designation=ddesig.getText();
317 Salary=Integer.parseInt(dsal.getText());
318 Room_NO=Integer.parseInt(droom.getText());
319 Report_NO=Integer.parseInt(dreport.getText());
320
321
322
323
324 //insert query
325 Class.forName("oracle.jdbc.driver.OracleDriver");
326 con = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:XE","DINH","Hridita81369");
327
328 PreparedStatement pst;
329 String sql ="INSERT INTO DOCTOR (Doctor_ID,Doctor_Name,Doctor_Age,Doctor_ContactNo,Department_Name,Doctor_Designation ,Salary,Room_NO,Report_NO
330 pst = con.prepareStatement(sql);
331 pst.setInt(1,Doctor_ID);
332 pst.setString(2,Doctor_Name);
333
334 pst.setInt(3, Doctor_Age);
335 pst.setInt(4, Doctor_ContactNo);
336
337 pst.setString(5, Department_Name);
338 pst.setString(6, Doctor_Designation);
339 pst.setInt(7, Salary);
340 pst.setInt(8, Room_NO);
341 pst.setInt(9, Report_NO);
342
343
344 if (did.getText().isEmpty()){
345     System.out.println("Plz Fill up blanks");
346 }
347 }
```

Question No 02:

Yes, we have created Admin Panel.



Question No 03:

Yes, we have maintained proper interface to communicate with database.



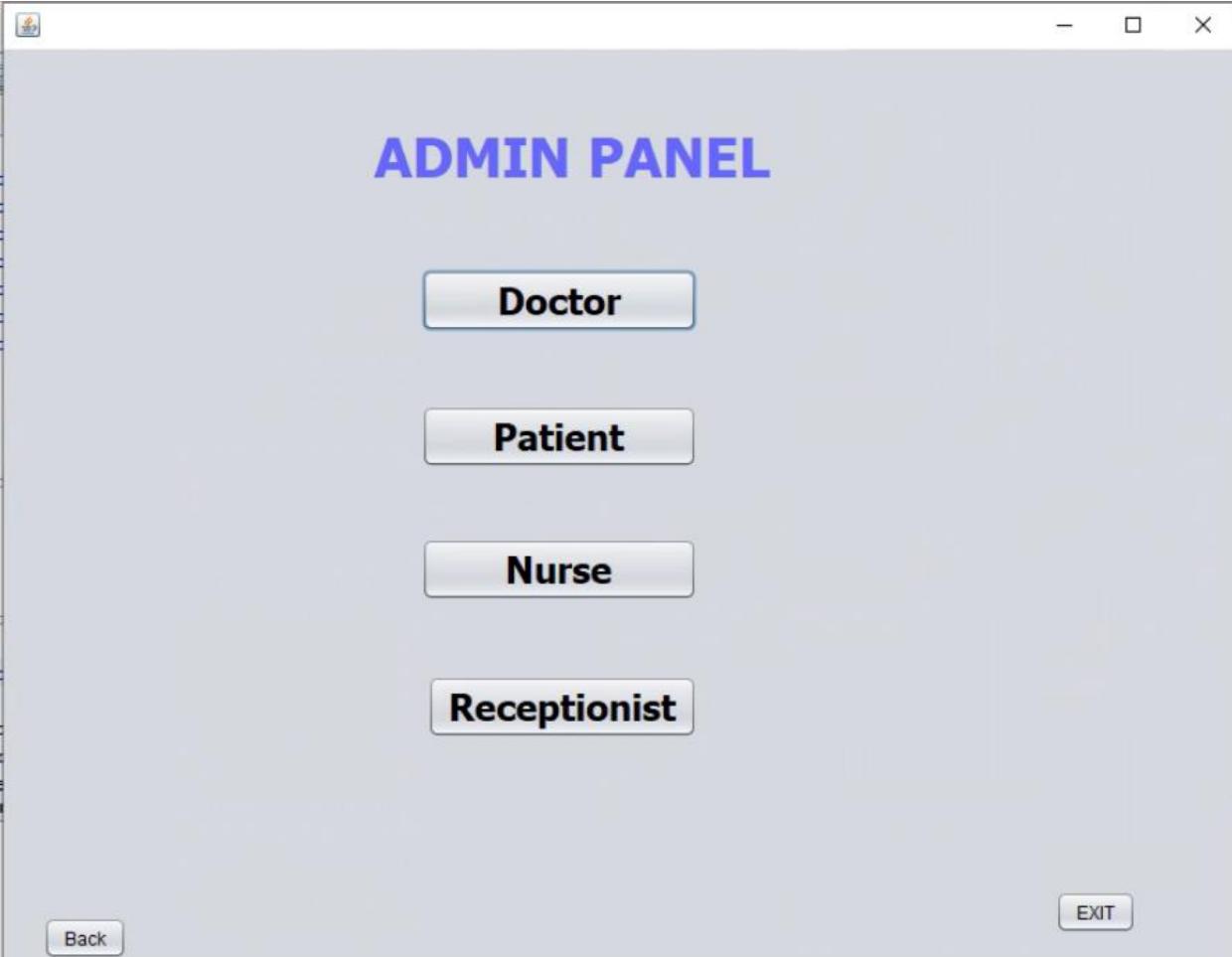
A screenshot of a Windows application window titled "Login Form". The window has a standard title bar with minimize, maximize, and close buttons. The main content area contains the following elements:

- Username:** A label followed by a text input field.
- Password:** A label followed by a password input field.
- Login:** A button located below the password field.
- Back:** A button located at the bottom left.
- Exit:** A button located at the bottom right.

The "Login Form" title is displayed in blue text, while the other labels and buttons are in black text. The input fields are represented by rectangular boxes with a light blue border.







Doctor Details

Doctor_ID:	<input type="text"/>	Department_Name:	<input type="text"/>	<input type="button" value="Search"/>
Doctor_Name:	<input type="text"/>	Doctor_Designation:	<input type="text"/>	<input type="button" value="Add Doctor"/>
Doctor_Age:	<input type="text"/>	Salary:	<input type="text"/>	<input type="button" value="Remove Doctor"/>
Doctor_ContactNo:	<input type="text"/>	Room_NO :	<input type="text"/>	<input type="button" value="Update"/>
		Report_NO :	<input type="text"/>	<input type="button" value="Display"/>

DOCTOR_ID	Doctor_Name	Doctor_Age	Doctor_ContactNo	Department_Name	Doctor_Designation	Salary	Room_No	Report_No

Doctor Details

```
← → ⌂ ⓘ
_apps YouTube
select *from DOCT
```

Doctor_ID:	<input type="text" value="101"/>	Department_Name:	<input type="text" value="Cardiology"/>	<input type="button" value="Search"/>
Doctor_Name:	<input type="text" value="Dr.Hridita Srabon"/>	Doctor_Designation:	<input type="text" value="Assistant"/>	<input type="button" value="Add Doctor"/>
Doctor_Age:	<input type="text" value="27"/>	Salary:	<input type="text" value="50000"/>	<input type="button" value="Remove Doctor"/>
Doctor_ContactNo:	<input type="text" value="13546436"/>	Room_NO :	<input type="text" value="300"/>	<input type="button" value="Update"/>
		Report_NO :	<input type="text" value="901"/>	<input type="button" value="Display"/>

DOCTOR_ID	Doctor_Name	Doctor_Age	Doctor_ContactNo	Department_Name	Doctor_Designation	Salary	Room_No	Report_No

Results
Explain

DOCTOR_ID	Doctor_Name	Doctor_Age	Doctor_ContactNo	Department_Name	Doctor_Designation	Salary	Room_No	Report_No
101	Dr Hridita Srabon	27	13546436	Cardiology	Assistant	50000	300	901
108	Dr Rashed Islam	45	1721212	Diagnostic	Assistant Professor	60000	305	902
115	Dr Ripa	41	1321212	Orthopedics	Assistant professor	45000	303	903
111	Dr Seemaa	38	11721212	Neurology	Professor	65000	301	904
198	Dr Uzzal Kumar	55	136255212	Neurology	Professor	55000	302	908
128	Dr. Hasan Mahabub	51	166255212	Cardiology	Assistant Professor	55000	304	905
200	Dr Samia	27	187683454	Cardiology	Lecturer	40000	300	901
121	Dr Khan	50	1821212	Cardiology	Assistant	50000	300	901
131	Dr Mahmud	50	1821212	Cardiology	Assistant	50000	300	901

9 rows returned in 0.02 seconds [Download](#)

Patient Details

Patient_ID:	Patient_Address:	Search
Patient_Name:	Department_Name	Add Patient
Patient_Age:	Room_NO:	Remove Patient
Patient_ContactNo:	Report_NO:	Update
Display		

Patient_ID	Patient_Name	Patient_Age	Patient_ContactNo	Patient_Address	Department_Name	Room_No	Report_No

Back Exit

Activate Windows
Go to Settings to activate Windows.

Patient Details

Patient_ID:	210	Patient_Address:	Comilla	Search
Patient_Name:	Mahmudul Hasan	Department_Name	Cardiology	Add Patient
Patient_Age:	30	Room_NO:	200	Remove Patient
Patient_ContactNo:	1324565545	Report_NO:	800	Update
Display				

Patient_ID	Patient_Name	Patient_Age	Patient_ContactNo	Patient_Address	Department_Name	Room_No	Report_No

Results Explain Desc

PATIENT_ID	PATI						
201	Samia Is						
202	Sholly S						
203	Sriti Kha						
204	Din Mohammad Dohan	25	162223246	Bogura	Neurology	303	904
206	Abcde	28	1655223246	khilkhet	Orthopedic	304	907
210	Mahmudul Hasan	30	1324565545	Comilla	Cardiology	200	800

6 rows returned in 0.02 seconds Download

Exit

Activate Windows
Go to Settings to activate Windows.

Nurse Details

Nurse_ID:	<input type="text"/>	Department_Name:	<input type="text"/>	Search
Nurse_Name:	<input type="text"/>	Nurse_Designation:	<input type="text"/>	Add Doctor
Nurse_Age:	<input type="text"/>	Salary:	<input type="text"/>	Remove Doctor
Nurse_ContactNo:	<input type="text"/>	Room_NO :	<input type="text"/>	Update
Display				

Nurse_ID	Nurse_Name	Nurse_Age	Nurse_ContactNo	Department_Name	Nurse_Designation	Salary	Room_No

Back	Exit
------	------

Activate Windows

Nurse Details

Nurse_ID:	<input type="text"/>	Department_Name:	<input type="text"/>	Search
Nurse_Name:	<input type="text"/>	Nurse_Designation:	<input type="text"/>	Add Doctor
Nurse_Age:	<input type="text"/>	Salary:	<input type="text"/>	Remove Doctor
Nurse_ContactNo:	<input type="text"/>	Room_NO :	<input type="text"/>	Update
Display				

Nurse_ID	Nurse_Name	Nurse_Age	Nurse_ContactNo	Department_Name	Nurse_Designation	Salary	Room_No
514	Liya	22	1912942121	Cardiology	Senior Nurse	14000	303

Receptionist Details

Receptionist_ID:	<input type="text"/>	Receptionist_Username:	<input type="text"/>	<input type="button" value="Search"/>
Receptionist_Name:	<input type="text"/>	Receptionist_Designation:	<input type="text"/>	<input type="button" value="Add"/>
Receptionist_Age:	<input type="text"/>	Salary:	<input type="text"/>	<input type="button" value="Remove"/> <input type="button" value="Update"/> <input type="button" value="Display"/>

Receptionist_ID	Receptionist_Name	Receptionist_Age	Receptionist_Username	Receptionist_Designation	Salary

Activsoft WinDev

Receptionist Details

Receptionist_ID:	<input type="text" value="1"/>	Receptionist_Username:	<input type="text" value="anik11"/>	<input type="button" value="Search"/>
Receptionist_Name:	<input type="text" value="Anik"/>	Receptionist_Designation:	<input type="text" value="Junior Receptionist"/>	<input type="button" value="Add"/>
Receptionist_Age:	<input type="text" value="27"/>	Salary:	<input type="text" value="30000"/>	<input type="button" value="Remove"/> <input type="button" value="Update"/> <input type="button" value="Display"/>

Receptionist_ID	Receptionist_Name	Receptionist_Age	Receptionist_Username	Receptionist_Designation	Salary

Activsoft WinDev

Question No 04:

Different type of searching and advanced searching (SQL Queries)

- 1. Display NAME, GENDER of the RECEPTIONISTS from RECEPTIONIST table whose Designation As 'Senior'.**

Answer: Select RECEPTIONIST_ID, RECEPTIONIST_NAME, RECEPTIONIST_GENDER from RECEPTIONIST WHERE RECEPTIONIST_ID IN (SELECT RECEPTIONIST_ID from RECEPTIONIST where RECEPTIONIST_DESIGNATION ='Senior');

- 2. Update NURSE SALARY from NURSE table which NURSE age is 28, they will get extra 1000 with their SALARY.**

Answer: UPDATE NURSE SET SALARY = SALARY + 1000 WHERE NURSE_AGE IN (SELECT NURSE_AGE FROM NURSE WHERE NURSE_AGE=28);

- 3. Find out the list of whose working at same DEPARTMENT as 'Cardiology from DOCTOR & NURSE table.**

Answer: SELECT 'DOCTOR' AS TYPE, Doctor_Name, Department_Name FROM DOCTOR WHERE Department_Name = 'Cardiology' UNION SELECT 'NURSE', NURSE_Name, Department_Name FROM NURSE WHERE Department_Name = 'Cardiology' order by Department_Name;

- 4. Find when Patient admit into hospital**

Answer:

```
BEGIN
FOR PATIENT IN (SELECT PATIENT_NAME, AdmitDate
FROM PATIENT)
LOOP
dbms_output.put_line (PATIENT.PATIENT_NAME || ':' ||
PATIENT.AdmitDate);
END LOOP;
END;
```

5. Find the Admin id where the id is 0 to 4

Answer:

```
DECLARE
n_ADMIN_ID ADMIN.ADMIN_ID%TYPE := 0;
BEGIN
LOOP
n_ADMIN_ID := n_ADMIN_ID + 1;
IF
n_ADMIN_ID>= 4 THEN
EXIT;
END IF;
dbms_output.put_line('ADMIN' ||':'|| n_ADMIN_ID );
END LOOP;
END;
```

6. Find the doctor name where the doctor id is 108

Answer:

```
DECLARE
d_DOCTOR DOCTOR%ROWTYPE;
BEGIN
SELECT * INTO d_DOCTOR
FROM DOCTOR
WHERE DOCTOR_ID = 108;
dbms_output.put_line(d_DOCTOR.DOCTOR_NAME ||':'||
d_DOCTOR.DOCTOR_ID );
END;
```

Question No 05:

Use sequence to maintain primary key in at least 2 tables.

1.

```
CREATE SEQUENCE doctor_seq
INCREMENT BY 10
START WITH 101
MINVALUE 10
MAXVALUE 200
CYCLE
CACHE 2;
```

```
INSERT INTO DOCTOR
VALUES (doctor_seq.nextval,'Dr. Khan', 50, 01821212,'Cardiology', 'Assistant',
50000, 300, 901);
```

```
INSERT INTO DOCTOR
VALUES (doctor_seq.nextval,'Dr. Mahmud', 50, 01821212,'Cardiology', 'Assistant',
50000, 300, 901);
```

```
Select *from DOCTOR;
```

```

CREATE SEQUENCE doctor_seq
INCREMENT BY 10
START WITH 101
MINVALUE 10
MAXVALUE 200
CYCLE
CACHE 2;]

INSERT INTO DOCTOR
VALUES
(doctor_seq.nextval,'Dr. Mahmud',50,01821212,'Cardiology', 'Assistant',50000,300,901)

select *from DOCTOR

```

[Results](#) [Explain](#) [Describe](#) [Saved SQL](#) [History](#)

DOCTOR_ID	DOCTOR_NAME	DOCTOR AGE	DOCTOR_CONTACTNO	DEPARTMENT_NAME	DOCTOR_DESIGNATION	SALARY	ROOM_NO	REPORT_NO
101	Dr.Hridita Srabon	27	13546436	Cardiology	Assistant	50000	300	901
108	Dr.Rashed Islam	45	1721212	Diagnostic	Assistant Professor	60000	305	902
115	Dr.Ripa	41	1321212	Orthopedics	Assistant professor	45000	303	903
111	Dr.Seemaa	38	11721212	Neurology	Professor	65000	301	904
198	Dr. Uzzal Kumar	55	136255212	Neurology	Professor	55000	302	908
128	Dr. Hasan Mahabub	51	166255212	Cardiology	Assistant Professor	55000	304	905
200	Dr. Samia	27	187683454	Cardiology	Lecturer	40000	300	901
121	Dr. Khan	50	1821212	Cardiology	Assistant	50000	300	901
131	Dr. Mahmud	50	1821212	Cardiology	Assistant	50000	300	901

```

INSERT INTO DOCTOR
VALUES
(doctor_seq.nextval,'Dr. Khan',50,01821212,'Cardiology', 'Assistant',50000,300,901)

select *from DOCTOR

```

[Results](#) [Explain](#) [Describe](#) [Saved SQL](#) [History](#)

DOCTOR_ID	DOCTOR_NAME	DOCTOR AGE	DOCTOR_CONTACTNO	DEPARTMENT_NAME	DOCTOR_DESIGNATION	SALARY	ROOM_NO	REPORT_NO
101	Dr.Hridita Srabon	27	13546436	Cardiology	Assistant	50000	300	901
108	Dr.Rashed Islam	45	1721212	Diagnostic	Assistant Professor	60000	305	902
115	Dr.Ripa	41	1321212	Orthopedics	Assistant professor	45000	303	903
111	Dr.Seemaa	38	11721212	Neurology	Professor	65000	301	904
198	Dr. Uzzal Kumar	55	136255212	Neurology	Professor	55000	302	908
128	Dr. Hasan Mahabub	51	166255212	Cardiology	Assistant Professor	55000	304	905
200	Dr. Samia	27	187683454	Cardiology	Lecturer	40000	300	901
121	Dr. Khan	50	1821212	Cardiology	Assistant	50000	300	901
131	Dr. Mahmud	50	1821212	Cardiology	Assistant	50000	300	901

9 rows returned in 0.00 seconds [Download](#)

2. CREATE SEQUENCE patient_seq

INCREMENT BY 5

START WITH 201

MINVALUE 10

MAXVALUE 300

NOCYCLE

CACHE 2;

INSERT INTO PATIENT

VALUES (patient_seq.nextval, 'Abcde', 28, 01655223246, 'khilkhet', 'Orthopedic', 304, 907);

Select * from PATIENT;

```
CREATE SEQUENCE patient_seq
INCREMENT BY 5
START WITH 201
MINVALUE 10
MAXVALUE 300
NOCYCLE
CACHE 2;
```

Results Explain Describe Saved SQL History

Sequence created.

```
INSERT INTO PATIENT
VALUES
(patient_seq.nextval,'Abcde',28,01655223246,'khilkhet','Orthopedic',304,907)
```

Results Explain Describe Saved SQL History

1 row(s) inserted.

Question No 06:

Minimum 5/6 view (min 2 complex among them)

- **View**

1.

```
CREATE OR REPLACE VIEW PatientInfo_View
AS
SELECT Patient_Name ,Patient_ID, AdmitDate,Report_NO
FROM PATIENT
WHERE Room_NO = 300;

SELECT * FROM PatientInfo_View;
```

```
CREATE OR REPLACE VIEW PatientInfo_View  
AS  
SELECT Patient_Name , Patient_ID, AdmitDate,Report_NO  
FROM PATIENT  
WHERE Room_NO = 300;
```

```
SELECT * FROM PatientInfo_View
```

Results Explain Describe Saved SQL History

View created.

```
CREATE OR REPLACE VIEW PatientInfo_View  
AS  
SELECT Patient_Name , Patient_ID, AdmitDate,Report_NO  
FROM PATIENT  
WHERE Room_NO = 300;
```

```
SELECT * FROM PatientInfo_View
```

Results Explain Describe Saved SQL History

PATIENT_NAME	PATIENT_ID	ADMITDATE	REPORT_NO
Samia Islam	201	21-july2020	901
Saiful Khan	206	22-april2020	905

2 rows returned in 0.00 seconds [Download](#)

2.

```
CREATE OR REPLACE VIEW Doctor_detp
AS
SELECT DOCTOR.DOCTOR_NAME,DOCTOR.DOCTOR_ID,DOCTOR.Salary,
CAST(DOCTOR.Doctor_JoiningDate AS Date) AS Hireing_DATE,
DEPARTMENT.Department_Name AS NameofDepartment
FROM DOCTOR
JOIN Department ON
DOCTOR.Department_Name = DEPARTMENT.Department_Name

SELECT * FROM Doctor_detp;
```

```
CREATE OR REPLACE VIEW Doctor_detp
AS
SELECT DOCTOR.DOCTOR_NAME,DOCTOR.DOCTOR_ID,DOCTOR.Salary,
CAST(DOCTOR.Doctor_JoiningDate AS Date) AS Hireing_DATE,
DEPARTMENT.Department_Name AS NameofDepartment
FROM DOCTOR
JOIN Department ON
DOCTOR.Department_Name = DEPARTMENT.Department_Name

SELECT * FROM Doctor_detp
```

Results Explain Describe Saved SQL History

DOCTOR_NAME	DOCTOR_ID	SALARY	HIREING_DATE	NAMEOFDEPARTMENT
Dr.Saifulla Khan	101	50000	02/05/2095	Cardiology
Dr.Rashed Islam	108	60000	03/09/2092	Diagnostic
Dr.Ripa	115	45000	11/05/2095	Orthopedics
Dr.Seemaa	111	65000	01/13/2099	Neurology
Dr. Uzzal Kumar	198	55000	11/13/2092	Neurology
Dr. Hasan Mahabub	128	55000	07/28/2099	Cardiology

6 rows returned in 0.12 seconds [Download](#)

3.

```
CREATE OR REPLACE VIEW nurse_salRange
AS
SELECT NURSE_NAME, AVG(salary) average_salary
FROM NURSE
GROUP BY NURSE_NAME
HAVING AVG(salary) > 12000;
```

```
SELECT * FROM nurse_salRange;
```

```
CREATE OR REPLACE VIEW nurse_salRange
AS
SELECT NURSE_NAME, AVG(salary) average_salary
FROM NURSE
GROUP BY NURSE_NAME
HAVING AVG(salary) > 12000;
```

```
SELECT * FROM nurse_salRange
```

Results Explain Describe Saved SQL History

NURSE_NAME	AVERAGE_SALARY
Liya	13000
Keya	13000

2 rows returned in 0.00 seconds

[Download](#)

4.

```
CREATE OR REPLACE VIEW RoomList(Doctor_ID,Patient_ID,Nurse_ID ,Room_NO)
AS
SELECT Doctor.Doctor_ID,PATIENT.Patient_ID,NURSE.Nurse_ID,ROOM.Room_NO
FROM DOCTOR, PATIENT,NURSE,ROOM
WHERE ROOM.Room_NO = 302 ;
SELECT * FROM RoomList
```

```
CREATE OR REPLACE VIEW RoomList
(Doctor_ID,Patient_ID,Nurse_ID ,Room_NO)

AS

SELECT Doctor.Doctor_ID,PATIENT.Patient_ID,NURSE.Nurse_ID, ROOM.Room_NO
FROM DOCTOR, PATIENT, NURSE, ROOM
WHERE ROOM.Room_NO = 302 ;

SELECT * FROM RoomList
```

Results Explain Describe Saved SQL History

DOCTOR_ID	PATIENT_ID	NURSE_ID	ROOM_NO
101	201	501	302
101	201	502	302
101	201	506	302
101	201	514	302
101	201	527	302
101	201	533	302
101	201	555	302
101	202	501	302
101	202	502	302
101	202	506	302

More than 10 rows available. Increase rows selector to view more rows.

5.

```
CREATE OR REPLACE VIEW NURSE_View
```

```
AS
```

```
SELECT Nurse_Name, Salary, Nurse_AGE, Department_Name, Nurse_Designation
```

```
FROM NURSE
```

```
WHERE Nurse_AGE >= 23;
```

```
SELECT * FROM NURSE_View;
```

```
CREATE OR REPLACE VIEW NURSE_View
```

```
AS
```

```
SELECT Nurse_Name, Salary, Nurse_AGE, Department_Name, Nurse_Designation
```

```
FROM NURSE
```

```
WHERE Nurse_AGE >= 23;
```

```
SELECT * FROM NURSE_View
```

Results Explain Describe Saved SQL History

NURSE_NAME	SALARY	NURSE_AGE	DEPARTMENT_NAME	NURSE_DESIGNATION
Nur	12000	28	Orthopedics™	Junior Nurse
Joy	11000	23	Diagnostic	Junior Nurse

2 rows returned in 0.00 seconds [Download](#)

6.

```
CREATE OR REPLACE FORCE VIEW joining_Date
AS
SELECT Receptionist_ID, Receptionist_Name, Receptionist_Age, Salary
FROM RECEPTIONIST
where Receptionist_JoiningDate between to_date('01-JAN-2007', 'DD-MM-YY') and
to_date('15-DEC-2007', 'DD-MM-YY');

SELECT * FROM joining_Date
```

```
CREATE OR REPLACE FORCE VIEW joining_Date
AS
SELECT Receptionist_ID, Receptionist_Name, Receptionist_Age, Salary
FROM RECEPTIONIST
where Receptionist_JoiningDate between to_date('01-JAN-2007', 'DD-MM-YY') and to_date('15-DEC-2007', 'DD-MM-YY');

SELECT * FROM joining_Date
```

Results Explain Describe Saved SQL History

RECEPTIONIST_ID	RECEPTIONIST_NAME	RECEPTIONIST_AGE	SALARY
704	Alif	30	19000
705	Sajib	26	11000
706	Rahim	28	11000

3 rows returned in 0.09 seconds [Download](#)

Question No 07:

Minimum 6 procedures and functions which must be embedded in the system

Procedures and functions

1. Checking the Patient Id is that valid id or not?

Answer:

```
Create or replace function CheckValidPatient_ID (pid Patient.Patient_ID %type)
Return boolean
Is
a number;
i number;
Cursor C
is
Select * from Patient;
Begin
a:=0;
For i in C
Loop
if(pid = i.Patient_ID) then
a:=1;
End if;
End loop;
if(a=1) then
Return true;
Else
Return false;
End if;
End;
```

```
create or replace function CheckValidPatient_ID (pid Patient.Patient_ID %type)
return boolean
is
a number;
i number;
cursor C
is
select * from Patient;
begin
a:=0;
for i in C
loop
if(pid = i.Patient_ID) then
a:=1;
end if;
end loop;
if(a=1) then
return true;
else
return false;
end if;
end;
```

Results Explain Describe Saved SQL History

Function created.

2. Search patient name

Answer:

```
CREATE OR REPLACE PROCEDURE SEARCH_PATIENT_NAME(Id IN
PATIENT.PATIENT_ID%TYPE)
```

AS

```
PatientName PATIENT.PATIENT_NAME%TYPE;
```

```
BEGIN

SELECT PatientName INTO PatientName FROM PATIENT WHERE PATIENT_ID = Id;
DBMS_OUTPUT.put_line('PATIENT_NAME' || PatientName);

END SEARCH_PATIENT_NAME;
```

```
CREATE OR REPLACE PROCEDURE SEARCH_PATIENT_NAME(Id IN PATIENT.PATIENT_ID%TYPE)
AS
PatientName  PATIENT.PATIENT_NAME%TYPE;
BEGIN
|
SELECT PatientName INTO PatientName FROM PATIENT WHERE PATIENT_ID      = Id;
DBMS_OUTPUT.put_line('PATIENT_NAME' || PatientName);
|
END SEARCH_PATIENT_NAME;
```

```
Select *from PATIENT
```

Results Explain Describe Saved SQL History

Procedure created.

3. Sum of Receptionist salary

Answer:

```
CREATE OR REPLACE PROCEDURE Receptionist_sum_sal(in_Receptionist_ID in Receptionist.Receptionist_ID %type)
```

```
IS
```

```
sum_sal number(10);
```

```
BEGIN
```

```
SELECT SUM(Salary) INTO sum_sal FROM Receptionist GROUP BY Receptionist_ID  
HAVING Receptionist_ID = in_Receptionist_ID;  
dbms_output.put_line('Receptionist_ID : ' || in_Receptionist_ID ||  
CHR(10)||'SUM of Salary: ' || sum_sal);  
END;
```

```
Declare
```

```
cursor em is
```

```
select Receptionist_ID from Receptionist;  
e_Receptionist_ID Receptionist.Receptionist_ID%type;
```

```
Begin
```

```
for i in em loop
```

```
e_Receptionist_ID :=i.Receptionist_ID;  
Receptionist_sum_sal(e_Receptionist_ID);  
End loop;  
End;
```

```

CREATE OR REPLACE PROCEDURE Receptionist_sum_sal(in_Receptionist_ID in Receptionist.Receptionist_ID %type)
|
IS
sum_sal number(10);

BEGIN
SELECT SUM(Salary) INTO sum_sal FROM Receptionist GROUP BY Receptionist_ID HAVING Receptionist_ID = in_Receptionist_ID;
dbms_output.put_line('Receptionist_ID : ' || in_Receptionist_ID || CHR(10)||'SUM of Salary: ' || sum_sal);
END;

```

[Results](#) [Explain](#) [Describe](#) [Saved SQL](#) [History](#)

Procedure created.

4. Is the nurse id is valid or not??

Answer:

```

create or replace function CheckValidNurse_ID (nid Nurse.Nurse_ID %type)
return boolean
is
a number;
i number;
cursor C
is
select * from Nurse;
begin
a:=0;
for i in C
loop
if(nid = i.Nurse_ID) then
a:=1;
end if;
end loop;
if(a=1) then
return true;

```

```
else
return false;
end if;
end;
```

```
create or replace function CheckValidNurse_ID (nid Nurse.Nurse_ID %type)
return boolean
is
a number;
i number;
cursor C
is
select * from Nurse;
begin
a:=0;
for i in C
loop
if(nid = i.Nurse_ID) then
a:=1;
end if;
end loop;
if(a=1) then
return true;
else
return false;
end if;
end;
```

[Results](#) [Explain](#) [Describe](#) [Saved SQL](#) [History](#)

Function created.

5. Delete Admin

Answer:

```
CREATE OR REPLACE PROCEDURE DELETE_ADMIN_ACCOUNT (Id in
ADMIN.ADMIN_ID%type)
```

```
AS
```

```
BEGIN
```

```
DELETE FROM ADMIN WHERE ADMIN_ID = Id;
```

```
END DELETE_ADMIN_ACCOUNT;
```

```
CREATE OR REPLACE PROCEDURE DELETE_ADMIN_ACCOUNT (Id in ADMIN.ADMIN_ID%type)
AS
BEGIN
DELETE FROM ADMIN WHERE ADMIN_ID = Id;
END DELETE_ADMIN_ACCOUNT;
```

Results Explain Describe Saved SQL History

Procedure created.

6. Check admin valid or not

Answer:

```
create or replace function CheckValidAdmin_ID (aid Admin.ADMIN_ID
%type)
return boolean
is
a number;
i number;
cursor C
is
select * from ADMIN;
begin
a:=0;
for i in C
loop
if(aid = i.ADMIN_ID) then
a:=1;
end if;
end loop;
if(a=1) then
return true;
else
return false;
end if;
end;
```

```
create or replace function CheckValidAdmin_ID (aid Admin.ADMIN_ID %type)
return boolean
is
a number;
i number;
cursor C
is
select * from ADMIN;
begin
a:=0;
for i in C
loop
if(aid = i.ADMIN_ID) then
a:=1;
end if;
end loop;
if(a=1) then
return true;
else
return false;
end if;
end;
```

[Results](#) [Explain](#) [Describe](#) [Saved SQL](#) [History](#)

Function created.

Question No 08:

Minimum 4 triggers which must be implemented in the system.

Triggers

1. Every doctor, nurse& RECEPTIONIST can be only register 8am to 5 pm
For Doctor

Answer:

For doctor: create or replace trigger Restricted_Insert
Before Insert on DOCTOR
Begin
if(TO_CHAR(sysdate,'HH24:MI') Not between '08:00' AND '17:00') then
raise_application_error(-20123,'You can add DOCTOR between 8:00 AM TO 5:00 PM ONLY');
End if;
End;

```
create or replace trigger Restricted_Insert
before Insert on DOCTOR
Begin|
if(TO_CHAR(sysdate,'HH24:MI') Not between '08:00' AND '17:00') then
raise application error(-20123,'You can add DOCTOR between 8:00 AM TO 5:00 PM ONLY');
end if;
end;
```

Results Explain Describe Saved SQL History

Trigger created.

0.03 seconds

For RECEPTIONIST:

Create or replace trigger Restricted_Insert_RECEPTIONIST

Before Insert on RECEPTIONIST

Begin

if(TO_CHAR(sysdate,'HH24:MI') Not between '08:00' AND '17:00') then
raise_application_error(-20123,'You can add RECEPTIONIST between 8:00 AM TO
5:00 PM ONLY');

End if;

End;

```
create or replace trigger Restricted_Insert_RECEPTIONIST
before Insert on RECEPTIONIST
Begin
if(TO_CHAR(sysdate,'HH24:MI') Not between '08:00' AND '17:00') then
raise_application_error(-20123,'You can add RECEPTIONIST between 8:00 AM TO 5:00 PM ONLY');
end if;
end;
```

Results Explain Describe Saved SQL History

Trigger created.

0.00 seconds

For NURSE:

Create or replace trigger Restricted_Insert_NURSE
Before Insert on NURSE
Begin
if(TO_CHAR(sysdate,'HH24:MI') Not between '08:00' AND '17:00') then
raise_application_error(-20123,'You can add NURSE between 8:00 AM TO 5:00
PM ONLY');
End if;
End;

```
create or replace trigger Restricted_Insert_NURSE
before Insert on NURSE
Begin
if(TO_CHAR(sysdate,'HH24:MI') Not between '08:00' AND '17:00') then
raise application error(-20123,'You can add NURSE between 8:00 AM TO 5:00 PM ONLY');
end if;
end;
```

Results Explain Describe Saved SQL History

Trigger created.

2. Doctor salary cannot be decrease than previous salary

Answer:

```
Create or replace trigger Salary_For_DOCTOR
Before Update of SALARY on DOCTOR
FOR EACH ROW
Begin
IF(:NEW.Salary<:OLD.Salary)
Then
    Raise_application_error(-20126, 'Doctors New salary cannot be less then
previous Salary');
End if;
End;
```

```
create or replace trigger Salary For DOCTOR
Before Update Of SALARY on DOCTOR
FOR EACH ROW
Begin
IF(:NEW.Salary<:OLD.Salary)
Then
    Raise application error(-20126, 'Doctors| New salary cannot be less then previous Salary');
end if;
end;
```

Results Explain Describe Saved SQL History

Trigger created.

3. Admin cannot be empty

Answer:

```
Create or replace TRIGGER ADMIN_Delete_Rule  
Before delete on ADMIN  
Declare  
Admin_count number(3);  
  
Begin  
Select count(Admin_id) into Admin_count from ADMIN;  
if(Admin_count<1) then  
    raise_application_error(-20126,'Minimum one ADMIN MUST be there');  
End if;  
End;
```

```
create or replace TRIGGER ADMIN Delete Rule  
before delete on ADMIN  
declare  
Admin_count number(3);  
  
Begin  
select count(Admin_id) into Admin|count from ADMIN;  
if(Admin count<1) then  
    raise application error(-20126,'Minimum one ADMIN MUST be there');  
end if;  
end;
```

Results Explain Describe Saved SQL History

Trigger created.

4. Nurse Salary cannot increase in 15% without his/her experience is less than 3 years

Answer:

Create or replace trigger Salary_Increment_For_NURSE
Before update of Salary on NURSE for each row

Declare

JoiningDate date;

Begin

select Nurse_JoiningDate into JoiningDate from NURSE where
NURSE_ID=:new.NURSE_ID;

if to_date(sysdate) - to_date(JoiningDate) >3*365 then
update NURSE set Salary = :new.Salary where NURSE_ID=:new.NURSE_ID;

elsif to_date(sysdate) - to_date(JoiningDate) < 3*365 and
:new.Salary<=(:old.Salary+:old.Salary*0.15) then
update NURSE set Salary = :new.Salary where NURSE_ID=:new.NURSE_ID;

else

Raise_application_error(-202026, 'Nurse salary cannot increase in 15% without
his/her experience is less than 3 years');

end if;

End;

```
Create or replace trigger Salary_Increment_For_NURSE
before update of Salary on NURSE for each row

Declare
    JoiningDate date;

Begin
    select Nurse_JoiningDate into JoiningDate from NURSE where NURSE_ID=:new.NURSE_ID;

    if to_date(sysdate) - to_date(JoiningDate ) >3*365 then
        update NURSE set Salary = :new.Salary where NURSE_ID=:new.NURSE_ID;

    elsif to_date(sysdate) - to_date(JoiningDate ) < 3*365 and :new.Salary<=(:old.Salary+:old.Salary*0.15) then
        update NURSE set Salary = :new.Salary where NURSE_ID=:new.NURSE_ID;

    else
        Raise_application_error(-202026, 'Nurse salary cannot increase in 15% without his/her experience is less than 3 years');
    end if;
End;
```

Results Explain Describe Saved SQL History

Trigger created.

Question No 09:

You may apply package, locking and exception handling which is optional and will carry bonus marks.

PACKAGE

1.

CREATE OR REPLACE PACKAGE ADD_PACKAGE

AS

```
PROCEDURE ADD_DOCTOR(DOCTOR_NAME IN DOCTOR.DOCTOR_NAME%TYPE,  
DOCTOR_AGE IN DOCTOR.DOCTOR_AGE%TYPE,DOCTOR_CONTACTNO IN  
DOCTOR.DOCTOR_CONTACTNO%TYPE, DOCTOR_DESIGNATION IN  
DOCTOR.DOCTOR_DESIGNATION%TYPE, SALARYIN DOCTOR.SALARY%TYPE,  
ROOM_NO IN DOCTOR.ROOM_NO%TYPE,REPORT_NO IN  
DOCTOR.REPORT_NO%TYPE);  
END ADD_PACKAGE;
```

```
CREATE OR REPLACE PACKAGE ADD_PACKAGE  
AS  
PROCEDURE ADD_DOCTOR(DOCTOR_NAME IN DOCTOR.DOCTOR_NAME%TYPE, DOCTOR_AGE IN DOCTOR.DOCTOR_AGE%TYPE,DOCTOR_CONTACTNO IN DOCTOR.DOCTOR_CONTACTNO%TYPE, DOCTOR_DESIGNATION IN DOCTOR.DOCTOR_DESIGNATION%TYPE, SALARYIN DOCTOR.SALARY%TYPE, ROOM_NO IN DOCTOR.ROOM_NO%TYPE,REPORT_NO IN DOCTOR.REPORT_NO%TYPE);  
END ADD_PACKAGE;
```

Results Explain Describe Saved SQL History



Package created.

2.

```
CREATE OR REPLACE PACKAGE BODY ADD_PACKAGE
```

```
AS
```

```
PROCEDURE ADD_DOCTOR(DOCTOR_NAME IN DOCTOR.DOCTOR_NAME%TYPE,  
DOCTOR_AGE IN DOCTOR.DOCTOR_AGE%TYPE,DOCTOR_CONTACTNO IN  
DOCTOR.DOCTOR_CONTACTNO%TYPE, DOCTOR_DESIGNATION IN  
DOCTOR.DOCTOR_DESIGNATION%TYPE, SALARY IN DOCTOR.SALARY%TYPE,  
ROOM_NO IN DOCTOR.ROOM_NO%TYPE,REPORT_NO IN  
DOCTOR.REPORT_NO%TYPE)
```

```
AS
```

```
BEGIN
```

```
INSERT INTO DOCTOR VALUES (DOCTOR_ID.NEXTVAL,DOCTOR_NAME,  
DOCTOR_AGE, DOCTOR_CONTACTNO,DOCTOR_DESIGNATION, SALARY,  
DocDesignation);
```

```
END ADD_DOCTOR;
```

LOCKING:

1.

```
LOCK TABLE NURSE IN SHARE ROW EXCLUSIVE MODE NOWAIT;  
Update NURSE SET SALARY = SALARY+1000 WHERE NURSE_NAME = 'NUR';  
SELECT * FROM NURSE;
```

```
1  LOCK TABLE NURSE IN SHARE ROW EXCLUSIVE MODE NOWAIT;  
2  Update NURSE SET SALARY = SALARY+1000 WHERE NURSE_NAME = 'NUR';  
3  SELECT * FROM NURSE  
4
```

Results	Explain	Describe	Saved SQL	History
0 row(s) updated.				

Exception handling:

1.

```
DECLARE
n_name NURSE.NURSE_NAME%TYPE;
n_NURSE_ID NURSE.NURSE_ID%TYPE := 556;
BEGIN
SELECT NURSE_NAME into n_name
FROM NURSE
WHERE NURSE_ID = n_NURSE_ID;
dbms_output.put_line( 'NURSE Name is ' || n_name );
EXCEPTION
WHEN NO_DATA_FOUND then
dbms_output.put_line('NurseID' ||':'|| n_NURSE_ID || ''|| 'does not exits');
END;
```

```
DECLARE
n_name NURSE.NURSE_NAME%TYPE;
n_NURSE_ID NURSE.NURSE_ID%TYPE := 556;
BEGIN
SELECT NURSE_NAME into n_name
FROM NURSE
WHERE NURSE_ID = n_NURSE_ID;
dbms_output.put_line( 'NURSE Name is ' || n_name );
EXCEPTION
WHEN NO_DATA_FOUND then
dbms_output.put_line('NurseID' ||':'|| n_NURSE_ID || ''|| 'does not exits');
END;
```

Results Explain Describe Saved SQL History

NurseID:556 does not exits

Statement processed.

2. DECLARE

```
d_idDOCTOR.DOCTOR_ID%type := 0;
d_nameDOCTOR.DOCTOR_Name%type;
ex_invalid_id EXCEPTION;
BEGIN
IF d_id<= 0 THEN
  RAISE ex_invalid_id;
ELSE
SELECT DOCTOR_NAME INTO d_name
  FROM DOCTOR
 WHERE DOCTOR_ID = d_id;
  DBMS_OUTPUT.PUT_LINE ('Name: '|| d_name);
END IF;
EXCEPTION
WHEN ex_invalid_id THEN
  dbms_output.put_line('ID must be greater than zero!');
WHEN no_data_found THEN
  dbms_output.put_line('No such customer!');
WHEN others THEN
  dbms_output.put_line('Error!');
END;
```

```
DECLARE
  d_id DOCTOR.DOCTOR_ID%type := 0;
  d_name DOCTOR.DOCTOR_Name%type;
  ex_invalid_id  EXCEPTION;
BEGIN
  IF d_id <= 0 THEN
    RAISE ex_invalid_id;
  ELSE
    SELECT DOCTOR_NAME INTO d_name
    FROM DOCTOR
    WHERE DOCTOR_ID = d_id;
    DBMS_OUTPUT.PUT_LINE ('Name: '|| d_name);
  END IF;
EXCEPTION
  WHEN ex_invalid_id THEN
    dbms_output.put_line('ID must be greater than zero!');
  WHEN no_data_found THEN
    dbms_output.put_line('No such customer!');
  WHEN others THEN
    dbms_output.put_line('Error!');
END;
```

Results Explain Describe Saved SQL History

ID must be greater than zero!

Statement processed.