

Hospital Management System using Oracle Database and Forms

GROUP MEMBERS:-

NAME	COLLEGE	REGISTRATION NO.
1.) GOPAL SHAURYA	HERITAGE INSTITUTE OF TECHNOLOGY	131260110048
2.) SANJEET KUMAR	HERITAGE INSTITUTE OF TECHNOLOGY	131260110090
3.) MAYANK KATARUKA	HERITAGE INSTITUTE OF TECHNOLOGY	131260110064
4.) SUBHADEEP KOLEY	RCC INSTITUTE OF INFORMATION TECHNOLOGY	141170110293

SUMAN ACH



INDEX

SERIAL NO.	CONTAINTS	PAGE NO.
1.	TITLE PAGE	1
2.	INDEX	2
3.	ACKNOWLEDGEMENT	3
4.	REQUIRED SPECIFICATION	4
5.	DATABASE DESIGN	5-13
6.	SCREEN SHOTS	14-22
7.	FUTURE SCOPE	23
8.	CODE	24-55
9.	CERTIFICATE	56-59

SUMAN ARCH



ACKNOWLEDGEMENT

Success is never achieved single handed so, it is our duty to acknowledge all those who have provided a helping hands in making this project successful.

Firstly we would like to express our deep sense of gratitude towards my project guide Mr. Suman Aich for his valuable guidance, co-operation and approval given by him in successful completion of this project. We also like to express our heartiest appreciation and best wishes to our fellow friends and batch mates for their contribution and encouragement during this work.

SUMAN AICH



REQUIRED SPECIFICATION

❖ DOMAIN DESCRIPTION

The Hospital Management System (HMS) is a computer-based system that establishes an orderly and systematic method of managing hospital and clinic patient records more effectively and efficiently for both out-patients and in-patients. It integrates the entire resources of a hospital into an integrated software platform. The HMS proposed by us has two main player one is the **Admin** another is **Operator**. It is capable of doing all job related to admin and operator.

❖ PROBLEM DEFINATION

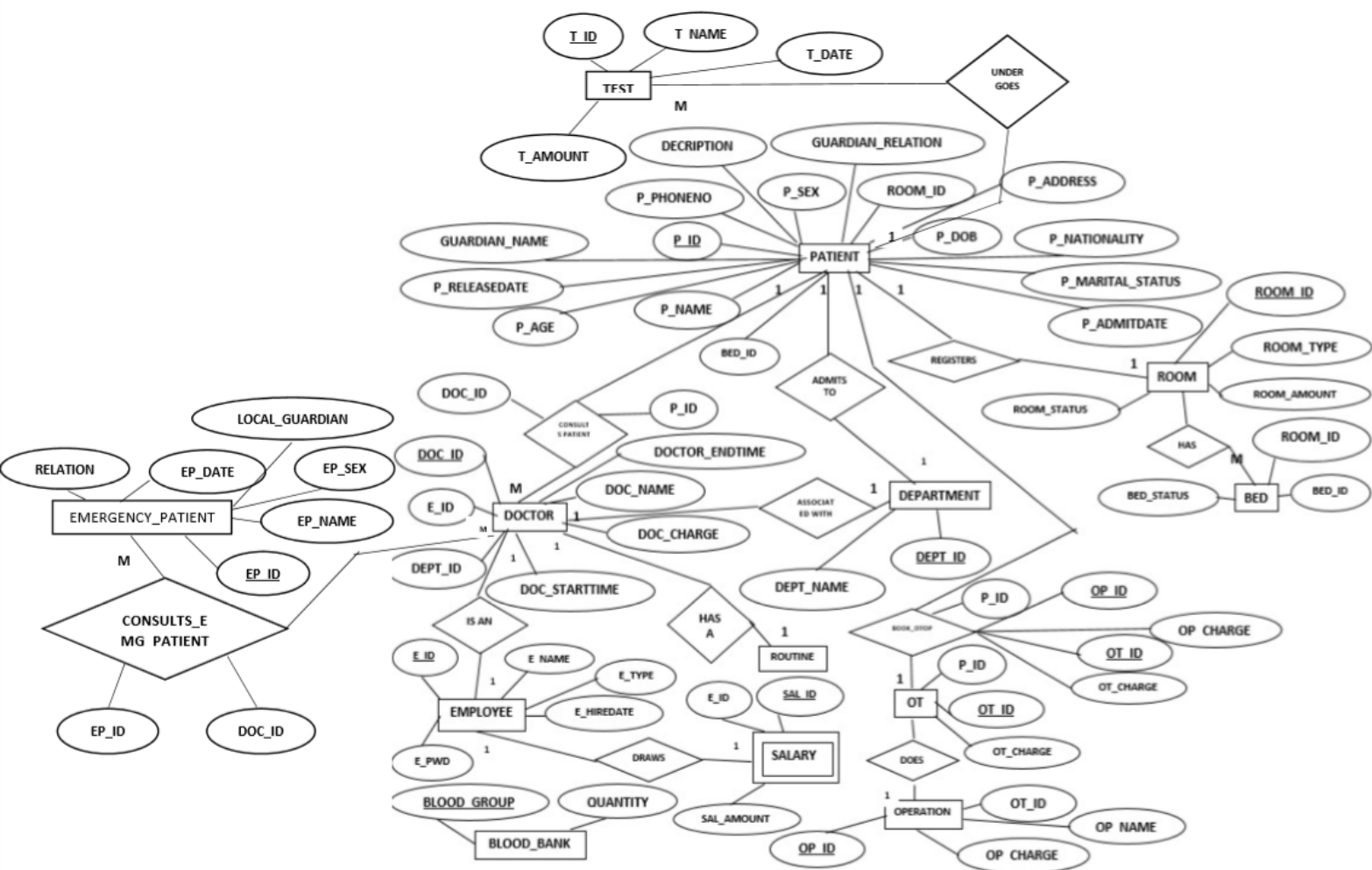
- 1.) Patients are facing difficulties for referring to the right department and to the right doctor
- 2.) Patients are unable to upgrade their services effortlessly.
- 3.) There is lack of proper emergency management system.
- 4.) Lack of proper coordination between the departments resulting in redundancy and inconvenience.
- 5.) The conventional method they are using for the management of the hospital is manually driven and the speed of operation is slow.
- 6.) The system they are using lacks immediate retrieval of records also it consists of error prone manual calculation.

❖ HARDWARE/SOFTWARE REQUIREMENT

- **HARDWARE REQUIREMENT:-**
 - 1.) Intel Core 2 duo processor or higher processor
 - 2.) 4 GB RAM or higher
 - 3.) 500 GB SATA HDD or higher
 - 4.) VGA color monitor with 1024*768 resolution.
- **SOFTWARE REQUIREMENT:-**
 - 1.) Oracle Database 10g Express Edition
 - 2.) Oracle Developer Suite 10g
 - 3.) Jinitiator 1.3.1.22
 - 4.) OC4J Instance
 - 5.) Windows XP service pack 3 or higher

DATABASE DESIGN

❖ E-R DIAGRAM:-



❖ TABLE DESCRIPTION:-

TNAME	TABTYPE	CLUSTERID
EMPLOYEE	TABLE	
DEPARTMENT	TABLE	
SALARY	TABLE	
ROOM	TABLE	

BED	TABLE
PATIENT	TABLE
EMG_PATIENT	TABLE
TEST	TABLE
BLOOD_BANK	TABLE
OT	TABLE
OPERATION	TABLE
DOCTOR	TABLE
CONSULT_PATIENT	TABLE
CONSULT_EMG_PATIENT	TABLE
PATIENT_TEST	TABLE
EMG_PATIENT_TEST	TABLE
ROUTINE	TABLE
BOOK_OTOP	TABLE
BILL	TABLE

SQL> DESC EMPLOYEE;

Name	Null?	Type

E_ID	NOT NULL	NUMBER(5)
E_NAME	NOT NULL	VARCHAR2(30)
E_TYPE	NOT NULL	VARCHAR2(15)
E_PWD		VARCHAR2(20)
E_HIREDATE	NOT NULL	DATE

SQL> DESC DEPARTMENT;

Name	Null?	Type

DEPT_ID	NOT NULL	NUMBER(5)
DEPT_NAME	NOT NULL	VARCHAR2(30)

SQL> DESC SALARY;

Name	Null?	Type

E_ID	NOT NULL	NUMBER(5)
SAL_ID	NOT NULL	NUMBER(5)
SAL_AMOUNT	NOT NULL	NUMBER(10)
SAL_DATE	NOT NULL	DATE

SQL> DESC ROOM;

Name	Null?	Type

ROOM_ID	NOT NULL	NUMBER(5)
ROOM_TYPE	NOT NULL	VARCHAR2(20)
ROOM_AMOUNT	NOT NULL	NUMBER(5)
ROOM_STATUS	NOT NULL	NUMBER(1)

SQL> DESC BED;

Name	Null?	Type

BED_ID	NOT NULL	NUMBER(5)

Suman Ach



ROOM_ID	NOT NULL	NUMBER(5)
BED_STATUS	NOT NULL	NUMBER(1)

SQL> DESC PATIENT;

Name	Null?	Type

P_ID	NOT NULL	NUMBER(5)
P_NAME	NOT NULL	VARCHAR2(30)
P_AGE		NUMBER(3)
P_SEX	NOT NULL	VARCHAR2(10)
P_PHONENO		NUMBER(13)
ROOM_ID		NUMBER(5)
P_ADDRESS		VARCHAR2(100)
NATIONALITY		VARCHAR2(20)
P_MARITAL_STATUS		VARCHAR2(15)
GUARDIAN_NAME		VARCHAR2(30)
GUARDIAN_RELATION		VARCHAR2(30)
P_DOB	NOT NULL	DATE
DESCRIPTION		VARCHAR2(50)
P_ADMIT_DATE	NOT NULL	DATE
P_RELEASE_DATE		DATE

SQL> DESC EMG_PATIENT;

Name	Null?	Type

EP_ID	NOT NULL	NUMBER(5)
EP_NAME		VARCHAR2(30)
LOCAL_GUARDIAN		VARCHAR2(30)
GUARDIAN_RELATION		VARCHAR2(30)
EP_SEX	NOT NULL	VARCHAR2(10)
EP_DATE	NOT NULL	DATE
EP_TIME	NOT NULL	DATE

SQL> DESC TEST;

Name	Null?	Type

T_ID	NOT NULL	NUMBER(5)
T_NAME	NOT NULL	VARCHAR2(30)
T_AMOUNT	NOT NULL	NUMBER(5)

SQL> DESC BLOOD_BANK;

Name	Null?	Type

BLOOD_GROUP	NOT NULL	VARCHAR2(3)
QUANTITY	NOT NULL	NUMBER(5)

SQL> DESC OT;

Name	Null?	Type

OT_ID	NOT NULL	NUMBER(5)

Suman Ach



OT_CHARGE NOT NULL NUMBER(5)

SQL> DESC OPERATION

Name	Null?	Type

OP_ID	NOT NULL	NUMBER(5)
OP_NAME	NOT NULL	VARCHAR2(30)
OP_CHARGE	NOT NULL	NUMBER(5)
OT_ID	NOT NULL	NUMBER(5)

SQL> DESC DOCTOR;

Name	Null?	Type

DOC_ID	NOT NULL	NUMBER(5)
DOC_NAME	NOT NULL	VARCHAR2(30)
DEPT_ID	NOT NULL	NUMBER(5)
E_ID	NOT NULL	NUMBER(5)
DOC_CHARGE	NOT NULL	NUMBER(5)

SQL> DESC CONSULT_PATIENT;

Name	Null?	Type

DOC_ID	NOT NULL	NUMBER(5)
P_ID	NOT NULL	NUMBER(5)

SQL> DESC CONSULT_EMG_PATIENT;

Suman Ach



Name	Null?	Type

DOC_ID	NOT NULL	NUMBER(5)
EP_ID	NOT NULL	NUMBER(5)

SQL> DESC PATIENT_TEST;

Name	Null?	Type

P_ID	NOT NULL	NUMBER(5)
T_ID	NOT NULL	NUMBER(5)
TEST_DATE		DATE

SQL> DESC EMG_PATIENT_TEST;

Name	Null?	Type

EP_ID	NOT NULL	NUMBER(5)
T_ID	NOT NULL	NUMBER(5)
T_DATE		DATE

SQL> DESC ROUTINE;

Name	Null?	Type

DOC_ID	NOT NULL	NUMBER(5)
MONS1	NOT NULL	NUMBER(1)

Suman Ach



MONS2	NOT NULL	NUMBER(1)
MONS3	NOT NULL	NUMBER(1)
MONS4	NOT NULL	NUMBER(1)
TUES1	NOT NULL	NUMBER(1)
TUES2	NOT NULL	NUMBER(1)
TUES3	NOT NULL	NUMBER(1)
TUES4	NOT NULL	NUMBER(1)
WEDS1	NOT NULL	NUMBER(1)
WEDS2	NOT NULL	NUMBER(1)
WEDS3	NOT NULL	NUMBER(1)
WEDS4	NOT NULL	NUMBER(1)
THUS1	NOT NULL	NUMBER(1)
THUS2	NOT NULL	NUMBER(1)
THUS3	NOT NULL	NUMBER(1)
THUS4	NOT NULL	NUMBER(1)
FRIS1	NOT NULL	NUMBER(1)
FRIS2	NOT NULL	NUMBER(1)
FRIS3	NOT NULL	NUMBER(1)
FRIS4	NOT NULL	NUMBER(1)
SATS1	NOT NULL	NUMBER(1)
SATS2	NOT NULL	NUMBER(1)
SATS3	NOT NULL	NUMBER(1)
SATS4	NOT NULL	NUMBER(1)
SUNS1	NOT NULL	NUMBER(1)
SUNS2	NOT NULL	NUMBER(1)

SUNS3	NOT NULL	NUMBER(1)
SUNS4	NOT NULL	NUMBER(1)

SQL> DESC BOOK_OTOP;

Name	Null?	Type

P_ID	NOT NULL	NUMBER(5)
OT_ID	NOT NULL	NUMBER(5)
OP_ID	NOT NULL	NUMBER(5)
OT_DATE	NOT NULL	DATE
OT_CHARGE	NOT NULL	NUMBER(5)
OP_CHARGE	NOT NULL	NUMBER(5)

SQL> DESC BILL;

Name	Null?	Type

BILL_ID	NOT NULL	NUMBER(5)
P_ID	NOT NULL	NUMBER(5)
P_NAME		VARCHAR2(30)
ROOM_CHARGE		NUMBER(5)
TEST_CHARGE		NUMBER(5)
OP_CHARGE		NUMBER(5)
DOC_CHARGE		NUMBER(5)
BILL_DATE		DATE
REMANING_TOTAL		NUMBER(10)
DEPOSIT		NUMBER(10)

❖ Some important screenshots of the application:-

This screenshot shows the login interface of the Narayana Multi-Speciality Hospital application. The browser window displays the URL `http://hp:8889/forms/frmservlet` and the Oracle Application Server F... tab. The application window, titled "WINDOW1", features the hospital's logo and name: "NARAYANA MULTI-SPECIALITY HOSPITAL", along with its certification: "AN ISO 9001:1995 CERTIFIED HOSPITAL" and address: "ANANDPUR, KOLKATA 700107, W.B., INDIA.".

The login form includes the following fields and controls:

- EMPLOYEE ID:** A text input field containing the value "10029".
- TYPE:** A dropdown menu currently set to "ADMIN".
- PASSWORD:** A text input field with masked characters "*****".
- LOGIN:** A button to submit the login credentials.
- Click to login:** A smaller button located below the main LOGIN button.

At the bottom of the application window, a status bar indicates "Password is case sensitive" and "Record: 1/1".

This screenshot shows the dashboard of the Narayana Multi-Speciality Hospital application after a successful login. The browser window shows the same URL and tab as the previous screenshot. The application window, titled "WINDOW1", displays the hospital's logo and name, and a message indicating the user is logged in: "Logged in as ADMIN" with "Employee ID 10029".

The dashboard includes a navigation menu at the top with the following items: EMPLOYEE, DEPARTMENT, DOCTOR, ROUTINE, SALARY, TEST, BLOOD BANK, ROOM, BED, QT, ACCOUNTS, LOGOUT, and Window. The main content area displays a "WELCOME ADMIN" message and two input fields for user information:

- NAME :** A text input field containing the value "MAYANK".
- HIREDATE:** A text input field containing the value "02-JAN-99".

At the bottom of the application window, a status bar indicates "Record: 1/1".

Oracle Application Server F...

EMPLOYEE DEPARTMENT DOCTOR ROUTINE SALARY TEST BLOOD BANK ROOM BED QT ACCOUNTS LOGOUT Window

ORACLE

WINDOW1

NARAYANA MULTI-SPECIALITY HOSPITAL
AN ISO 9001:1995 CERTIFIED HOSPITAL
ANANDPUR, KOLKATA-700107, W.B., INDIA.

EMPLOYEE DETAILS

EMPLOYEE ID	EMPLOYEE NAME	TYPE	HIREDATE	PASSWORD
10001	DR RAMANAND	DOCTOR	21-JAN-2014	ASDFG
10002	DR JOSHI	DOCTOR	20-FEB-2015	ASDFG
10003	DR BANSALI	DOCTOR	15-DEC-2014	ASDFG
10004	DR KAUSHIK	DOCTOR	02-DEC-2010	ASDFG
10005	DR GOENKA	DOCTOR	10-DEC-2011	ASDFG

INSERT UPDATE SHOW RECORDS DELETE CLEAR BLOCK

Click to show all records

Record: 1/2

Oracle Application Server F...

EMPLOYEE DEPARTMENT DOCTOR ROUTINE SALARY TEST BLOOD BANK ROOM BED QT ACCOUNTS LOGOUT Window

ORACLE

WINDOW1

NARAYANA MULTI-SPECIALITY HOSPITAL
AN ISO 9001:1995 CERTIFIED HOSPITAL
ANANDPUR, KOLKATA-700107, W.B., INDIA.

DEPARTMENT ID	DEPARTMENT NAME
40001	CARDIOLOGY
40002	ORTHOPEDICS
40003	ONCOLOGY
40004	GASTROENTEROLOGY
40005	NEUROSCIENCE

INSERT UPDATE SHOW RECORD DELETE CLEAR BLOCK

Click to delete the selected record

Record: 1/2

http://hp:8889/forms/frmservlet Oracle Application Server F...

EMPLOYEE DEPARTMENT DOCTOR ROUTINE SALARY TEST BLOOD BANK ROOM BED QT ACCOUNTS LOGOUT Window ORACLE

NARAYANA MULTI-SPECIALITY HOSPITAL
AN ISO 9001:1995 CERTIFIED HOSPITAL
ANANDPUR, KOLKATA-700107, W.B., INDIA.

DOCTOR ID: 20001

INSERT UPDATE SHOW RECORD DELETE CLEAR BLOCK
Click to clear the block

MONDAY SLOT1	<input checked="" type="checkbox"/>	TUESDAY SLOT1	<input checked="" type="checkbox"/>	WEDNESDAY SLOT1	<input checked="" type="checkbox"/>	THURSDAY SLOT1	<input checked="" type="checkbox"/>	FRIDAY SLOT 1	<input checked="" type="checkbox"/>	SATURDAY SLOT1	<input checked="" type="checkbox"/>	SUNDAY SLOT1	<input checked="" type="checkbox"/>
MONDAY SLOT2	<input type="checkbox"/>	TUESDAY SLOT2	<input type="checkbox"/>	WEDNESDAY SLOT2	<input type="checkbox"/>	THURSDAY SLOT2	<input type="checkbox"/>	FRIDAY SLOT 2	<input type="checkbox"/>	SATURDAY SLOT2	<input type="checkbox"/>	SUNDAY SLOT2	<input type="checkbox"/>
MONDAY SLOT3	<input checked="" type="checkbox"/>	TUESDAY SLOT3	<input checked="" type="checkbox"/>	WEDNESDAY SLOT13	<input checked="" type="checkbox"/>	THURSDAY SLOT3	<input checked="" type="checkbox"/>	FRIDAY SLOT 3	<input checked="" type="checkbox"/>	SATURDAY SLOT3	<input checked="" type="checkbox"/>	SUNDAY SLOT3	<input checked="" type="checkbox"/>
MONDAY SLOTA	<input type="checkbox"/>	TUESDAY SLOTA	<input type="checkbox"/>	WEDNESDAY SLOTA	<input type="checkbox"/>	THURSDAY SLOTA	<input type="checkbox"/>	FRIDAY SLOTA	<input type="checkbox"/>	SATURDAY SLOTA	<input type="checkbox"/>	SUNDAY SLOTA	<input type="checkbox"/>

Record: 1/? <OSC>

http://hp:8889/forms/frmservlet Oracle Application Server F...

EMPLOYEE DEPARTMENT DOCTOR ROUTINE SALARY TEST BLOOD BANK ROOM BED QT ACCOUNTS LOGOUT Window ORACLE

NARAYANA MULTI-SPECIALITY HOSPITAL
AN ISO 9001:1995 CERTIFIED HOSPITAL
ANANDPUR, KOLKATA-700107, W.B., INDIA.

EMPLOYEE ID	SALARY ID	SALARY AMOUNT	SALARY DATE
10001	80001	50000	28-JAN-2016
10002	80002	50000	28-JAN-2016
10003	80003	50000	28-JAN-2016
10004	80004	50000	28-JAN-2016
10005	80005	50000	28-JAN-2016

INSERT UPDATE SHOW RECORDS DELETE CLEAR BLOCK
Click to insert a record

Record: 1/? <OSC>

http://hp:8889/forms/frmservlet Oracle Application Server F...

EMPLOYEE DEPARTMENT DOCTOR ROUTINE SALARY TEST BLOOD BANK ROOM BED QT ACCOUNTS LOGOUT Window ORACLE

NARAYANA MULTI-SPECIALITY HOSPITAL
AN ISO 9001:1995 CERTIFIED HOSPITAL
ANANDPUR, KOLKATA-700107, W.B., INDIA.

ROOM DETAIL

ROOM ID	ROOM TYPE	ROOM AMOUNT	ROOM STATUS
60001	GENERAL	1200	0
60002	GENERAL	1200	0
60005	AC	3000	0
60006	AC	3000	0
60003	DELUXE	8000	0

INSERT UPDATE SHOW RECORDS DELETE CLEAR BLOCK

Click to insert a record

Record: 1/? <OSC>

http://hp:8889/forms/frmservlet Oracle Application Server F...

PATIENT REG. PATIENT SERVICE EMERGENCY PATIENT EMG PATIENT SERVICES BLOOD BANK BILL GENERATION LOGOUT Window ORACLE

NARAYANA MULTI-SPECIALITY HOSPITAL
AN ISO 9001:1995 CERTIFIED HOSPITAL
ANANDPUR, KOLKATA-700107, W.B., INDIA.

PATIENT REGISTRATION

PATIENT NAME GOPAL SHAURYA GUARDIAN RELATION MOTHER PATIENT ID 1005

PATIENT DOB 19-DEC-1996 GUARDIAN NAME ALKA RANI

PATIENT SEX MALE ROOM TYPE GENERAL ALLOCATE BED ID 70003

PATIENT AGE 20 ADMIT DATE ROOM ID 60001

MARITAL STATUS SINGLE RELEASE DATE

NATIONALITY INDIAN PATIENT ADDRESS

PHONE NUMBER 8017115597

DESCRIPTION HEART VALVE RUPTURE SAVE RECORDS

ROOM IS ALLOCATED

Record: 1/1 <OSC>

http://hp:8889/forms/frmservlet Oracle Application Server F...

PATIENT REG. PATIENT SERVICE EMERGENCY PATIENT EMG PATIENT SERVICES BLOOD BANK BILL GENERATION LOGOUT Window

NARAYANA MULTI-SPECIALITY HOSPITAL
AN ISO 9001:1995 CERTIFIED HOSPITAL
ANANDPUR, KOLKATA-700107, W.B., INDIA.

PATIENT REGISTRATION

PATIENT NAME: GOPAL SHAURVA GUARDIAN RELATION: MOTHER PATIENT ID: 1005

PATIENT DOB: 19-DEC-1996 GUARDIAN NAME: ALKA RANI

PATIENT SEX: MALE ROOM TYPE: GENERAL ALLOCATE BED ID: 70003

PATIENT AGE: 20 ADMIT DATE: 20-JUL-2016 ROOM ID: 60001

MARITAL STATUS: SINGLE RELEASE DATE: 27-JUL-2016

NATIONALITY: INDIAN PATIENT ADDRESS: HIRAPUR, DHANBAD, JHARKHAND-826001.

PHONE NUMBER: 8017115597

DESCRIPTION: HEART VALVE RUPTURE

SAVE RECORDS
Click to save record

FRM-40400: Transaction complete: 1 records applied and saved.
Record: 1/1 <OSC>

http://hp:8889/forms/frmservlet Oracle Application Server F...

PATIENT REG. PATIENT SERVICE EMERGENCY PATIENT EMG PATIENT SERVICES BLOOD BANK BILL GENERATION LOGOUT Window

NARAYANA MULTI-SPECIALITY HOSPITAL
AN ISO 9001:1995 CERTIFIED HOSPITAL
ANANDPUR, KOLKATA-700107, W.B., INDIA.

PATIENT ID: 1005

DOCTOR REGISTRATION

DEPARTMENT: CARDIOLOGY DAY: THURSDAY SLOT: 12:00-18:00 BOOK DOCTOR
Click to book a doctor

DOCTOR BOOKING DETAIL

DOCTOR ID: 20002 DOCTOR NAME: DR JOSHI

TEST REGISTRATION

TEST NAME: REGISTER TEST

OPERATION REGISTRATION

OPERATION NAME: REGISTER OPERATION O.T NUMBER:

DOCTOR IS BOOKED1
Record: 1/1 <OSC>

http://hp:8889/forms/frmservlet Oracle Application Server F...

PATIENT REG. PATIENT SERVICE EMERGENCY PATIENT EMG PATIENT SERVICES BLOOD BANK BILL GENERATION LOGOUT Window

NARAYANA MULTI-SPECIALITY HOSPITAL
AN ISO 9001:1995 CERTIFIED HOSPITAL
ANANDPUR, KOLKATA-700107, W.B., INDIA.

PATIENT ID 1005

DOCTOR REGISTRATION

DEPARTMENT CARDIOLOGY DAY THURSDAY SLOT 12:00-18:00 BOOK DOCTOR

DOCTOR BOOKING DETAIL

DOCTOR ID 20002 DOCTOR NAME DR JOSHI

TEST REGISTRATION

TEST NAME ECG REGISTER TEST

OPERATION REGISTRATION

OPERATION NAME REGISTER OPERATION O.T NUMBER

REGISTRATION CONFIRMED

Record: 1/1 <OSC>

http://hp:8889/forms/frmservlet Oracle Application Server F...

PATIENT REG. PATIENT SERVICE EMERGENCY PATIENT EMG PATIENT SERVICES BLOOD BANK BILL GENERATION LOGOUT Window

NARAYANA MULTI-SPECIALITY HOSPITAL
AN ISO 9001:1995 CERTIFIED HOSPITAL
ANANDPUR, KOLKATA-700107, W.B., INDIA.

PATIENT ID 1005

DOCTOR REGISTRATION

DEPARTMENT CARDIOLOGY DAY THURSDAY SLOT 12:00-18:00 BOOK DOCTOR

DOCTOR BOOKING DETAIL

DOCTOR ID 20002 DOCTOR NAME DR JOSHI

TEST REGISTRATION

TEST NAME ECG REGISTER TEST

OPERATION REGISTRATION

OPERATION NAME CARDIAC SURGERY REGISTER OPERATION O.T NUMBER 90007

Click to register for an operation

Record: 1/1 <OSC>

Oracle Application Server F...

PATIENT REG. PATIENT SERVICE EMERGENCY PATIENT EMG PATIENT SERVICES BLOOD BANK BILL GENERATION LOGOUT Window

NARAYANA MULTI-SPECIALITY HOSPITAL
AN ISO 9001:1995 CERTIFIED HOSPITAL
ANANDPUR, KOLKATA-700107, W.B., INDIA.

PATIENT ID 5005

EMERGENCY PATIENT DETAILS

NAME MAYANK KATARUIKA
LOCAL GUARDIAN SANJEET KUMAR
RELATION BROTHER
SEX MALE
ADMISSION DATE 21-JUN-2016

SAVE RECORD

Forms
Do you want to save the changes you have made?
Yes No Cancel

FRM-40400: Transaction complete: 1 records applied and saved.
Record: 0/1 <OSC>

Oracle Application Server F...

PATIENT REG. PATIENT SERVICE EMERGENCY PATIENT EMG PATIENT SERVICES BLOOD BANK BILL GENERATION LOGOUT Window

NARAYANA MULTI-SPECIALITY HOSPITAL
AN ISO 9001:1995 CERTIFIED HOSPITAL
ANANDPUR, KOLKATA-700107, W.B., INDIA.

TRANSACTION ID 7000

ENTER PATIENT ID 1005 SUBMIT

PATIENT NAME GOPAL SHAURYA SEX MALE DOB 19-DEC-96
PATIENT AGE 20 PHONE NO 8017115597 ADMIT DATE 20-JUL-16

ROOM CHARGES

ROOM TYPE	ROOM AMOUNT	TOTAL NUMBER OF DAYS SPENT	AMOUNT
GENERAL	1200	X 10	12000

TEST CHARGE

TEST ID	TEST NAME	TEST AMOUNT	TEST DATE
50001	ECG	4000	29-JUL-16

TOTAL 4000

CONSULT

Record: 1/1 <OSC>

Oracle Application Server F...

PATIENT REG. PATIENT SERVICE EMERGENCY PATIENT EMG PATIENT SERVICES BLOOD BANK BILL GENERATION LOGOUT Window

ORACLE

TEST CHARGE			
TEST ID	TEST NAME	TEST AMOUNT	TEST DATE
50001	ECG	4000	29-JUL-16
TOTAL		4000	

CONSULT		
Doc Id	DOCTOR NAME	DOCTOR CHARGE
20002	DR JOSHI	1500
TOTAL		1500

OPERATION CHARGE			
OT DATE	OT CHARGE	OPERATION CHARGE	OPERATION NAME
29-JUL-2016	30000	50000	CARDIAC SURG
TOTAL		80000	

TOTAL 97500

Record: 1/1 <OSC>

Oracle Application Server F...

PATIENT REG. PATIENT SERVICE EMERGENCY PATIENT EMG PATIENT SERVICES BLOOD BANK BILL GENERATION LOGOUT Window

ORACLE

CONSULT			
Doc Id	DOCTOR NAME	DOCTOR CHARGE	
20002	DR JOSHI	1500	
TOTAL		1500	

OPERATION CHARGE			
OT DATE	OT CHARGE	OPERATION CHARGE	OPERATION NAME
29-JUL-2016	30000	50000	CARDIAC SURG
TOTAL		80000	

TOTAL 97500

UNPAID AMOUNT 97500

ENTER AMOUNT TO BE DEPOSITED

FINAL TOTAL

PRINT BILL PAY BILL RELEASE PATIENT

Record: 1/1 <OSC>

SUMAN ACH

Document sign date :Sep 21, 2016



Oracle Application Server F...

PATIENT REG. PATIENT SERVICE EMERGENCY PATIENT EMG PATIENT SERVICES BLOOD BANK BILL GENERATION LOGOUT Window

ORACLE

TOTAL 4000

CONSULT

Doc Id DOCTOR NAME DOCTOR CHARGE

20002 DR JOSHI 1500

TOTAL 1500

OPERATION CHARGE

OT DATE OT CHARGE OPERATION CHARGE OPERATION NAME

29-JUL-2016 30000 50000 CARDIAC SURG

TOTAL 80000

TOTAL 97500

UNPAID AMOUNT 7500

ENTER AMOUNT TO BE DEPOSITED 90000

FINAL TOTAL

PRINT BILL PAY BILL RELEASE PATIENT

Record: 1/1 <OSC>

Oracle Application Server F...

PATIENT REG. PATIENT SERVICE EMERGENCY PATIENT EMG PATIENT SERVICES BLOOD BANK BILL GENERATION LOGOUT Window

ORACLE

TOTAL 4000

CONSULT

Doc Id DOCTOR NAME DOCTOR CHARGE

20002 DR JOSHI 1500

TOTAL 1500

OPERATION CHARGE

OT DATE OT CHARGE OPERATION CHARGE OPERATION NAME

29-JUL-2016 30000 50000 CARDIAC SURG

Forms

PATIENT IS RELEASED. GET WELL SOON!

OK

TOTAL 97500

UNPAID AMOUNT 0

ENTER AMOUNT TO BE DEPOSITED 7500

FINAL TOTAL

PRINT BILL PAY BILL RELEASE PATIENT

PATIENT IS RELEASED. GET WELL SOON!

Record: 1/1 <OSC>

FUTURE SCOPE

This hospital management system application is built using Oracle 11g database management system and Oracle forms. There is always a scope of betterment and the candidate system is not against this perception. At present the system satisfy most of the basic functions of a hospital.

 The future scopes may be:-

- ✓ Using appropriate hardware we can implement biometric operator and admin login system to increase the security of the database.
- ✓ Generating important reports on day wise or month wise.
- ✓ Reducing manual efforts for maintaining the system.
- ✓ Reducing process time.
- ✓ Emphasis on data accuracy.
- ✓ Making it more robust, user friendly and capable of handling large amount of data.

CODE

❖ DDL:-

```
create user hms1 identified by tiger;
grant connect,resource to hms1;
disc;
conn hms1/tiger;
create table employee(e_id number(5) primary key,e_name varchar2(30) not null, e_type
varchar2(15) not null,e_pwd varchar2(20),e_hired date not null);
CREATE TABLE DEPARTMENT(DEPT_ID NUMBER(5) PRIMARY KEY,DEPT_NAME VARCHAR2(30) NOT
NULL);
CREATE TABLE SALARY(E_ID NUMBER(5) NOT NULL,SAL_ID NUMBER(5) PRIMARY
KEY,SAL_AMOUNT NUMBER(10) NOT NULL,SAL_DATE DATE NOT NULL,CONSTRAINT
FK_SALARY_EID FOREIGN KEY(E_ID) REFERENCES EMPLOYEE(E_ID));
CREATE TABLE ROOM(ROOM_ID NUMBER(5) PRIMARY KEY,ROOM_TYPE VARCHAR2(20) NOT
NULL,ROOM_AMOUNT NUMBER(5) NOT NULL,ROOM_STATUS NUMBER(1) NOT NULL);
CREATE TABLE BED(BED_ID NUMBER(5) PRIMARY KEY,ROOM_ID NUMBER(5) NOT
NULL,BED_STATUS NUMBER(1) NOT NULL,CONSTRAINT FK_BED_ROOMID FOREIGN
KEY(ROOM_ID) REFERENCES ROOM(ROOM_ID));
CREATE TABLE PATIENT(P_ID NUMBER(5) PRIMARY KEY,P_NAME VARCHAR2(30) NOT NULL,P_AGE
NUMBER(3),P_SEX VARCHAR2(10) NOT NULL,P_PHONENO NUMBER(13),ROOM_ID
NUMBER(5),P_ADDRESS VARCHAR2(100) ,NATIONALITY VARCHAR2(20),P_MARITAL_STATUS
VARCHAR2(15),GUARDIAN_NAME VARCHAR2(30),GUARDIAN_RELATION VARCHAR2(30),P_DOB
DATE NOT NULL,DESCRIPTION VARCHAR2(50),P_ADMIT_DATE DATE NOT NULL,P_RELEASE_DATE
DATE,bed_id number(5),CONSTRAINT FK_PATIENT_ROOMID FOREIGN KEY(ROOM_ID) REFERENCES
ROOM(ROOM_ID));
CREATE TABLE PATIENT_TEST(P_ID NUMBER(5),T_ID NUMBER(5),TEST_DATE DATE NOT
NULL,CONSTRAINT PK_PATIENTTEST PRIMARY KEY(P_ID,T_ID),CONSTRAINT FK_PATIENTTEST_PID
FOREIGN KEY(P_ID) REFERENCES PATIENT(P_ID),CONSTRAINT FK_PATIENTTEST_TID FOREIGN
KEY(T_ID) REFERENCES TEST(T_ID));
CREATE TABLE CONSULT_PATIENT(DOC_ID NUMBER(5) NOT NULL,P_ID NUMBER(5) NOT
NULL,CONSTRAINT PK_CONSULTSPATIENT PRIMARY KEY(DOC_ID,P_ID),CONSTRAINT
FK_CONSULTSPATIENT_DOCID FOREIGN KEY(DOC_ID) REFERENCES DOCTOR(DOC_ID),CONSTRAINT
FK_CONSULTSPATIENT_PID FOREIGN KEY(P_ID) REFERENCES PATIENT(P_ID));
```



```

CREATE TABLE EMG_PATIENT(EP_ID NUMBER(5) PRIMARY KEY,EP_NAME
VARCHAR2(30),LOCAL_GUARDIAN VARCHAR(30),GUARDIAN_RELATION VARCHAR2(30),EP_SEX
VARCHAR2(10) NOT NULL,EP_DATE DATE NOT NULL,EP_TIME DATE);
CREATE TABLE EMG_PATIENT_TEST(EP_ID NUMBER(5),T_ID NUMBER(5),T_DATE
DATE,CONSTRAINT PK_EMGPATIENTTEST PRIMARY KEY(EP_ID,T_ID),CONSTRAINT
FK_EMGPATIENTTEST_EPID FOREIGN KEY(EP_ID) REFERENCES EMG_PATIENT(EP_ID),CONSTRAINT
FK_EMGPATIENTTEST_TID FOREIGN KEY(T_ID) REFERENCES TEST(T_ID));
CREATE TABLE CONSULT_EMG_PATIENT(DOC_ID NUMBER(5),EP_ID NUMBER(5),CONSTRAINT
PK_CONSULTSEMGPATIENT PRIMARY KEY(DOC_ID,EP_ID),CONSTRAINT
FK_CONSULTSEMGPATIENT_DOCID FOREIGN KEY(DOC_ID) REFERENCES
DOCTOR(DOC_ID),CONSTRAINT FK_CONSULTSEMGPATIENT_EPID FOREIGN KEY(EP_ID)
REFERENCES EMG_PATIENT(EP_ID));
CREATE TABLE TEST(T_ID NUMBER(5) PRIMARY KEY,T_NAME VARCHAR2(30) NOT
NULL,T_AMOUNT NUMBER(5) NOT NULL);
CREATE TABLE BLOOD_BANK(BLOOD_GROUP VARCHAR2(3) PRIMARY KEY,QUANTITY NUMBER(5)
NOT NULL);
CREATE TABLE OT(OT_ID NUMBER(5) PRIMARY KEY,OT_CHARGE NUMBER(5) NOT NULL);
CREATE TABLE OPERATION(OP_ID NUMBER(5) PRIMARY KEY,OP_NAME VARCHAR2(30) NOT
NULL,OP_CHARGE NUMBER(5) NOT NULL,OT_ID NUMBER(5) NOT NULL,CONSTRAINT
FK_OPERATION_OTID FOREIGN KEY(OT_ID) REFERENCES OT(OT_ID));
CREATE TABLE BOOK_OTOP(P_ID NUMBER(5) NOT NULL,OT_ID NUMBER(5) NOT NULL,OP_ID
NUMBER(5) NOT NULL,OT_DATE DATE NOT NULL,OT_CHARGE NUMBER(5) NOT NULL,OP_CHARGE
NUMBER(5) NOT NULL);
CREATE TABLE BILL(BILL_ID NUMBER(5),P_ID NUMBER(5),P_NAME VARCHAR2(30),ROOM_CHARGE
NUMBER(5),TEST_CHARGE NUMBER(5),OP_CHARGE NUMBER(5),DOC_CHARGE
NUMBER(5),BILL_DATE DATE DEFAULT SYSDATE,REMANING_TOTAL NUMBER(10),DEPOSIT
NUMBER(10),CONSTRAINT PK_BILL PRIMARY KEY(BILL_ID,P_ID));
CREATE TABLE DOCTOR(DOC_ID NUMBER(5) PRIMARY KEY,DOC_NAME VARCHAR2(30) NOT
NULL,DEPT_ID NUMBER(5) NOT NULL,E_ID NUMBER(5) NOT NULL,DOC_CHARGE NUMBER(5) NOT
NULL,CONSTRAINT FK_DOCTOR_EID FOREIGN KEY(E_ID) REFERENCES
EMPLOYEE(E_ID),CONSTRAINT FK_DOCTOR_DEPTID FOREIGN KEY(DEPT_ID) REFERENCES
DEPARTMENT(DEPT_ID));
CREATE TABLE ROUTINE(DOC_ID NUMBER(5) PRIMARY KEY,MONS1 NUMBER(1) NOT NULL,MONS2
NUMBER(1) NOT NULL,MONS3 NUMBER(1) NOT NULL,MONS4 NUMBER(1) NOT NULL,TUES1
NUMBER(1) NOT NULL,TUES2 NUMBER(1) NOT NULL,TUES3 NUMBER(1) NOT NULL,TUES4
NUMBER(1) NOT NULL,WEDS1 NUMBER(1) NOT NULL,WEDS2 NUMBER(1) NOT NULL,WEDS3

```

```

NUMBER(1) NOT NULL,WEDS4 NUMBER(1) NOT NULL,THUS1 NUMBER(1) NOT NULL,THUS2
NUMBER(1) NOT NULL,THUS3 NUMBER(1) NOT NULL,THUS4 NUMBER(1) NOT NULL,FRIS1
NUMBER(1) NOT NULL,FRIS2 NUMBER(1) NOT NULL,FRIS3 NUMBER(1) NOT NULL,FRIS4
NUMBER(1) NOT NULL,SATS1 NUMBER(1) NOT NULL,SATS2 NUMBER(1) NOT NULL,SATS3
NUMBER(1) NOT NULL,SATS4 NUMBER(1) NOT NULL,SUNS1 NUMBER(1) NOT NULL,SUNS2
NUMBER(1) NOT NULL,SUNS3 NUMBER(1) NOT NULL,SUNS4 NUMBER(1) NOT NULL);
CREATE TABLE ABC(DC_ID NUMBER(5));

```

--FUNCTIONS

```

CREATE OR REPLACE FUNCTION BED_ALLOCATE
(ROOM_TYPE1 IN VARCHAR2)
RETURN NUMBER
IS
BED1 NUMBER;
CURSOR C2 IS
SELECT BED_ID FROM ROOM R,BED B WHERE ROOM_TYPE=ROOM_TYPE1 AND
ROOM_STATUS=0 AND BED_STATUS=0 AND B.ROOM_ID=R.ROOM_ID;
BEGIN
FOR I IN
(SELECT BED_ID FROM ROOM R,BED B WHERE ROOM_TYPE=ROOM_TYPE1 AND
ROOM_STATUS=0 AND BED_STATUS=0 AND B.ROOM_ID=R.ROOM_ID)
LOOP
OPEN C2;
FETCH C2 INTO BED1;
IF C2%NOTFOUND THEN
BED1:=0;
DBMS_OUTPUT.PUT_LINE('NOT AVAILABLE');
RETURN BED1;
ELSE
BED1:=I.BED_ID;
DBMS_OUTPUT.PUT_LINE(BED1);
RETURN BED1;
END IF;
END LOOP;

```

```

DBMS_OUTPUT.PUT_LINE('BED1');
END;
/

CREATE OR REPLACE FUNCTION ALLOCATE_BED
(ROOM_TYPE1 IN VARCHAR2)
RETURN NUMBER
IS
A NUMBER;
BEGIN
A := BED_ALLOCATE(ROOM_TYPE1);
    IF(A=0) THEN
        UPDATE ROOM SET ROOM_STATUS=1 WHERE ROOM_ID=(SELECT ROOM_ID FROM BED
WHERE BED_ID = A);
        COMMIT;
        RETURN 0;
    ELSE
        UPDATE BED SET BED_STATUS=1 WHERE BED_ID=A;
        COMMIT;
        RETURN A;
    END IF;
END;

/

CREATE OR REPLACE FUNCTION DOCTOR_INFO
(DNAME IN VARCHAR2 ,PID IN NUMBER,NUM IN NUMBER)
RETURN NUMBER
IS
docid NUMBER;
BEGIN
    CASE NUM
        WHEN 1 THEN
            INSERT INTO CONSULT_PATIENT VALUES((SELECT R.DOC_ID FROM ROUTINE
R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID AND
DE.DEPT_NAME=DNAME AND R.MONS1=1),PID); COMMIT;

```

```

        WHEN 2 THEN
            INSERT INTO CONSULT_PATIENT VALUES((SELECT R.DOC_ID FROM ROUTINE
R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID AND
DE.DEPT_NAME=DNAME AND R.MONS2=1),PID); COMMIT;
        WHEN 3 THEN
            INSERT INTO CONSULT_PATIENT VALUES((SELECT R.DOC_ID FROM ROUTINE
R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID AND
DE.DEPT_NAME=DNAME AND R.MONS3=1),PID); COMMIT;
        WHEN 4 THEN
            INSERT INTO CONSULT_PATIENT VALUES((SELECT R.DOC_ID FROM ROUTINE
R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID AND
DE.DEPT_NAME=DNAME AND R.MONS4=1),PID); COMMIT;
        WHEN 5 THEN
            INSERT INTO CONSULT_PATIENT VALUES((SELECT R.DOC_ID FROM ROUTINE
R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID AND
DE.DEPT_NAME=DNAME AND R.TUES1=1),PID); COMMIT;
        WHEN 6 THEN
            INSERT INTO CONSULT_PATIENT VALUES((SELECT R.DOC_ID FROM ROUTINE
R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID AND
DE.DEPT_NAME=DNAME AND R.TUES2=1),PID); COMMIT;
        WHEN 7 THEN
            INSERT INTO CONSULT_PATIENT VALUES((SELECT R.DOC_ID FROM ROUTINE
R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID AND
DE.DEPT_NAME=DNAME AND R.TUES3=1),PID); COMMIT;
        WHEN 8 THEN
            INSERT INTO CONSULT_PATIENT VALUES((SELECT R.DOC_ID FROM ROUTINE
R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID AND
DE.DEPT_NAME=DNAME AND R.TUES4=1),PID); COMMIT;
        WHEN 9 THEN
            INSERT INTO CONSULT_PATIENT VALUES((SELECT R.DOC_ID FROM ROUTINE
R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID AND
DE.DEPT_NAME=DNAME AND R.WEDS1=1),PID); COMMIT;
        WHEN 10 THEN
            INSERT INTO CONSULT_PATIENT VALUES((SELECT R.DOC_ID FROM ROUTINE
R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID AND
DE.DEPT_NAME=DNAME AND R.WEDS2=1),PID); COMMIT;

```

```

        WHEN 11 THEN
            INSERT INTO CONSULT_PATIENT VALUES((SELECT R.DOC_ID FROM ROUTINE
R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID AND
DE.DEPT_NAME=DNAME AND R.WEDS3=1),PID); COMMIT;
        WHEN 12 THEN
            INSERT INTO CONSULT_PATIENT VALUES((SELECT R.DOC_ID FROM ROUTINE
R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID AND
DE.DEPT_NAME=DNAME AND R.WEDS4=1),PID); COMMIT;
        WHEN 13 THEN
            INSERT INTO CONSULT_PATIENT VALUES((SELECT R.DOC_ID FROM ROUTINE
R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID AND
DE.DEPT_NAME=DNAME AND R.THUS1=1),PID); COMMIT;
        WHEN 14 THEN
            INSERT INTO CONSULT_PATIENT VALUES((SELECT R.DOC_ID FROM ROUTINE
R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID AND
DE.DEPT_NAME=DNAME AND R.THUS2=1),PID); COMMIT;
        WHEN 15 THEN
            INSERT INTO CONSULT_PATIENT VALUES((SELECT R.DOC_ID FROM ROUTINE
R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID AND
DE.DEPT_NAME=DNAME AND R.THUS3=1),PID); COMMIT;
        WHEN 16 THEN
            INSERT INTO CONSULT_PATIENT VALUES((SELECT R.DOC_ID FROM ROUTINE
R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID AND
DE.DEPT_NAME=DNAME AND R.THUS4=1),PID); COMMIT;
        WHEN 17 THEN
            INSERT INTO CONSULT_PATIENT VALUES((SELECT R.DOC_ID FROM ROUTINE
R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID AND
DE.DEPT_NAME=DNAME AND R.FRIS1=1),PID); COMMIT;
        WHEN 18 THEN
            INSERT INTO CONSULT_PATIENT VALUES((SELECT R.DOC_ID FROM ROUTINE
R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID AND
DE.DEPT_NAME=DNAME AND R.FRIS2=1),PID); COMMIT;
        WHEN 19 THEN
            INSERT INTO CONSULT_PATIENT VALUES((SELECT R.DOC_ID FROM ROUTINE
R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID AND
DE.DEPT_NAME=DNAME AND R.FRIS3=1),PID); COMMIT;

```

```

        WHEN 20 THEN
            INSERT INTO CONSULT_PATIENT VALUES((SELECT R.DOC_ID FROM ROUTINE
R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID AND
DE.DEPT_NAME=DNAME AND R.FRIS4=1),PID); COMMIT;
        WHEN 21 THEN
            INSERT INTO CONSULT_PATIENT VALUES((SELECT R.DOC_ID FROM ROUTINE
R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID AND
DE.DEPT_NAME=DNAME AND R.SATS1=1),PID); COMMIT;
        WHEN 22 THEN
            INSERT INTO CONSULT_PATIENT VALUES((SELECT R.DOC_ID FROM ROUTINE
R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID AND
DE.DEPT_NAME=DNAME AND R.SATS2=1),PID); COMMIT;
        WHEN 23 THEN
            INSERT INTO CONSULT_PATIENT VALUES((SELECT R.DOC_ID FROM ROUTINE
R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID AND
DE.DEPT_NAME=DNAME AND R.SATS3=1),PID); COMMIT;
        WHEN 24 THEN
            INSERT INTO CONSULT_PATIENT VALUES((SELECT R.DOC_ID FROM ROUTINE
R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID AND
DE.DEPT_NAME=DNAME AND R.SATS4=1),PID); COMMIT;
        WHEN 25 THEN
            INSERT INTO CONSULT_PATIENT VALUES((SELECT R.DOC_ID FROM ROUTINE
R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID AND
DE.DEPT_NAME=DNAME AND R.SUNS1=1),PID); COMMIT;
        WHEN 26 THEN
            INSERT INTO CONSULT_PATIENT VALUES((SELECT R.DOC_ID FROM ROUTINE
R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID AND
DE.DEPT_NAME=DNAME AND R.SUNS2=1),PID); COMMIT;
        WHEN 27 THEN
            INSERT INTO CONSULT_PATIENT VALUES((SELECT R.DOC_ID FROM ROUTINE
R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID AND
DE.DEPT_NAME=DNAME AND R.SUNS3=1),PID); COMMIT;
        WHEN 28 THEN
            INSERT INTO CONSULT_PATIENT VALUES((SELECT R.DOC_ID FROM ROUTINE
R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID AND
DE.DEPT_NAME=DNAME AND R.SUNS4=1),PID); COMMIT;

```

```
ELSE
DOCID := -1;
```

```
END CASE;
IF(DOCID =-1) THEN
RETURN DOCID;
ELSE
RETURN 1;
END IF;
COMMIT;
```

```
END;
```

```
/
```

```
CREATE OR REPLACE FUNCTION DOCTOR_INFO_EMG
(DNAME IN VARCHAR2 ,PID IN NUMBER,NUM IN NUMBER)
RETURN NUMBER
IS
doc_id NUMBER;
BEGIN
CASE NUM
WHEN 1 THEN
INSERT INTO CONSULT_EMG_PATIENT VALUES((SELECT R.DOC_ID FROM
ROUTINE R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID
AND DE.DEPT_NAME=DNAME AND R.MONS1=1),PID); COMMIT;
WHEN 2 THEN
INSERT INTO CONSULT_EMG_PATIENT VALUES((SELECT R.DOC_ID FROM
ROUTINE R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID
AND DE.DEPT_NAME=DNAME AND R.MONS2=1),PID); COMMIT;
WHEN 3 THEN
INSERT INTO CONSULT_EMG_PATIENT VALUES((SELECT R.DOC_ID FROM
ROUTINE R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID
AND DE.DEPT_NAME=DNAME AND R.MONS3=1),PID); COMMIT;
WHEN 4 THEN
```

```

INSERT INTO CONSULT_EMG_PATIENT VALUES((SELECT R.DOC_ID FROM
ROUTINE R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID
AND DE.DEPT_NAME=DNAME AND R.MONS4=1),PID); COMMIT;
WHEN 5 THEN
INSERT INTO CONSULT_EMG_PATIENT VALUES((SELECT R.DOC_ID FROM
ROUTINE R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID
AND DE.DEPT_NAME=DNAME AND R.TUES1=1),PID); COMMIT;
WHEN 6 THEN
INSERT INTO CONSULT_EMG_PATIENT VALUES((SELECT R.DOC_ID FROM
ROUTINE R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID
AND DE.DEPT_NAME=DNAME AND R.TUES2=1),PID); COMMIT;
WHEN 7 THEN
INSERT INTO CONSULT_EMG_PATIENT VALUES((SELECT R.DOC_ID FROM
ROUTINE R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID
AND DE.DEPT_NAME=DNAME AND R.TUES3=1),PID); COMMIT;
WHEN 8 THEN
INSERT INTO CONSULT_EMG_PATIENT VALUES((SELECT R.DOC_ID FROM
ROUTINE R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID
AND DE.DEPT_NAME=DNAME AND R.TUES4=1),PID); COMMIT;
WHEN 9 THEN
INSERT INTO CONSULT_EMG_PATIENT VALUES((SELECT R.DOC_ID FROM
ROUTINE R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID
AND DE.DEPT_NAME=DNAME AND R.WEDS1=1),PID); COMMIT;
WHEN 10 THEN
INSERT INTO CONSULT_EMG_PATIENT VALUES((SELECT R.DOC_ID FROM
ROUTINE R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID
AND DE.DEPT_NAME=DNAME AND R.WEDS2=1),PID); COMMIT;
WHEN 11 THEN
INSERT INTO CONSULT_EMG_PATIENT VALUES((SELECT R.DOC_ID FROM
ROUTINE R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID
AND DE.DEPT_NAME=DNAME AND R.WEDS3=1),PID); COMMIT;
WHEN 12 THEN
INSERT INTO CONSULT_EMG_PATIENT VALUES((SELECT R.DOC_ID FROM
ROUTINE R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID
AND DE.DEPT_NAME=DNAME AND R.WEDS4=1),PID); COMMIT;
WHEN 13 THEN

```



```

INSERT INTO CONSULT_EMG_PATIENT VALUES((SELECT R.DOC_ID FROM
ROUTINE R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID
AND DE.DEPT_NAME=DNAME AND R.THUS1=1),PID); COMMIT;
    WHEN 14 THEN
        INSERT INTO CONSULT_EMG_PATIENT VALUES((SELECT R.DOC_ID FROM
ROUTINE R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID
AND DE.DEPT_NAME=DNAME AND R.THUS2=1),PID); COMMIT;
    WHEN 15 THEN
        INSERT INTO CONSULT_EMG_PATIENT VALUES((SELECT R.DOC_ID FROM
ROUTINE R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID
AND DE.DEPT_NAME=DNAME AND R.THUS3=1),PID); COMMIT;
    WHEN 16 THEN
        INSERT INTO CONSULT_EMG_PATIENT VALUES((SELECT R.DOC_ID FROM
ROUTINE R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID
AND DE.DEPT_NAME=DNAME AND R.THUS4=1),PID); COMMIT;
    WHEN 17 THEN
        INSERT INTO CONSULT_EMG_PATIENT VALUES((SELECT R.DOC_ID FROM
ROUTINE R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID
AND DE.DEPT_NAME=DNAME AND R.FRIS1=1),PID); COMMIT;
    WHEN 18 THEN
        INSERT INTO CONSULT_EMG_PATIENT VALUES((SELECT R.DOC_ID FROM
ROUTINE R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID
AND DE.DEPT_NAME=DNAME AND R.FRIS2=1),PID); COMMIT;
    WHEN 19 THEN
        INSERT INTO CONSULT_EMG_PATIENT VALUES((SELECT R.DOC_ID FROM
ROUTINE R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID
AND DE.DEPT_NAME=DNAME AND R.FRIS3=1),PID); COMMIT;
    WHEN 20 THEN
        INSERT INTO CONSULT_EMG_PATIENT VALUES((SELECT R.DOC_ID FROM
ROUTINE R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID
AND DE.DEPT_NAME=DNAME AND R.FRIS4=1),PID); COMMIT;
    WHEN 21 THEN
        INSERT INTO CONSULT_EMG_PATIENT VALUES((SELECT R.DOC_ID FROM
ROUTINE R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID
AND DE.DEPT_NAME=DNAME AND R.SATS1=1),PID); COMMIT;
    WHEN 22 THEN

```

```

INSERT INTO CONSULT_EMG_PATIENT VALUES((SELECT R.DOC_ID FROM
ROUTINE R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID
AND DE.DEPT_NAME=DNAME AND R.SATS2=1),PID); COMMIT;
    WHEN 23 THEN
        INSERT INTO CONSULT_EMG_PATIENT VALUES((SELECT R.DOC_ID FROM
ROUTINE R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID
AND DE.DEPT_NAME=DNAME AND R.SATS3=1),PID); COMMIT;
    WHEN 24 THEN
        INSERT INTO CONSULT_EMG_PATIENT VALUES((SELECT R.DOC_ID FROM
ROUTINE R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID
AND DE.DEPT_NAME=DNAME AND R.SATS4=1),PID); COMMIT;
    WHEN 25 THEN
        INSERT INTO CONSULT_EMG_PATIENT VALUES((SELECT R.DOC_ID FROM
ROUTINE R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID
AND DE.DEPT_NAME=DNAME AND R.SUNS1=1),PID); COMMIT;
    WHEN 26 THEN
        INSERT INTO CONSULT_EMG_PATIENT VALUES((SELECT R.DOC_ID FROM
ROUTINE R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID
AND DE.DEPT_NAME=DNAME AND R.SUNS2=1),PID); COMMIT;
    WHEN 27 THEN
        INSERT INTO CONSULT_EMG_PATIENT VALUES((SELECT R.DOC_ID FROM
ROUTINE R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID
AND DE.DEPT_NAME=DNAME AND R.SUNS3=1),PID); COMMIT;
    WHEN 28 THEN
        INSERT INTO CONSULT_EMG_PATIENT VALUES((SELECT R.DOC_ID FROM
ROUTINE R,DOCTOR D,DEPARTMENT DE WHERE R.DOC_ID=D.DOC_ID AND DE.DEPT_ID=D.DEPT_ID
AND DE.DEPT_NAME=DNAME AND R.SUNS4=1),PID); COMMIT;

ELSE
DOCID := -1;

END CASE;
IF(DOCID =-1) THEN
RETURN DOCID;
ELSE

```

```
        RETURN 1;
    END IF;
    COMMIT;
END;
```

/

--TRIGGERS

```
CREATE OR REPLACE TRIGGER CONSULT_TRIGGER
AFTER INSERT
ON CONSULT_PATIENT
FOR EACH ROW
BEGIN
    INSERT INTO ABC VALUES(:NEW.DOC_ID);
END;
```

/

```
CREATE OR REPLACE TRIGGER CONSULT_EMG_TRIGGER
AFTER INSERT
ON CONSULT_EMG_PATIENT
FOR EACH ROW
BEGIN
    INSERT INTO ABC VALUES(:NEW.DOC_ID);
END;
```

/

--Trigger created.



DML:-

TABLE NAME-EMPLOYEE

```
INSERT INTO EMPLOYEE VALUES(10001,'DR RAMANAND','DOCTOR','ASDFG','21-JAN-14');
INSERT INTO EMPLOYEE VALUES(10002,'DR JOSHI','DOCTOR','ASDFG','20-FEB-15');
INSERT INTO EMPLOYEE VALUES(10003,'DR BANSALI','DOCTOR','ASDFG','15-DEC-14');
INSERT INTO EMPLOYEE VALUES(10004,'DR KAUSHIK','DOCTOR','ASDFG','02-DEC-10');
INSERT INTO EMPLOYEE VALUES(10005,'DR GOENKA','DOCTOR','ASDFG','10-DEC-11');
INSERT INTO EMPLOYEE VALUES(10006,'DR MANOJ','DOCTOR','ASDFG','15-FEB-96');
INSERT INTO EMPLOYEE VALUES(10007,'DR ROHIT','DOCTOR','ASDFG','17-JAN-05');
INSERT INTO EMPLOYEE VALUES(10008,'DR RAKESH','DOCTOR','ASDFG','16-DEC-12');
```

```

INSERT INTO EMPLOYEE VALUES(10009,'DR ANAND','DOCTOR','ASDFG','29-FEB-04');
INSERT INTO EMPLOYEE VALUES(10010,'DR ABHISHEK','DOCTOR','ASDFG','07-MAR-94');
INSERT INTO EMPLOYEE VALUES(10011,'DR SHETAL','DOCTOR','ASDFG','09-JUL-96');
INSERT INTO EMPLOYEE VALUES(10012,'DR MAYA','DOCTOR','ASDFG','07-AUG-98');
INSERT INTO EMPLOYEE VALUES(10013,'DR DEEPAK','DOCTOR','ASDFG','07-AUG-05');
INSERT INTO EMPLOYEE VALUES(10014,'DR DIPANKAR','DOCTOR','ASDFG','19-DEC-96');
INSERT INTO EMPLOYEE VALUES(10015,'DR ANKITA','DOCTOR','ASDFG','16-JAN-97');
INSERT INTO EMPLOYEE VALUES(10016,'DR NEHA','DOCTOR','ASDFG','22-JAN-93');
INSERT INTO EMPLOYEE VALUES(10017,'DR RAJ','DOCTOR','ASDFG','26-MAR-95');
INSERT INTO EMPLOYEE VALUES(10018,'DR PIYUSH','DOCTOR','ASDFG','08-OCT-01');
INSERT INTO EMPLOYEE VALUES(10019,'DR ANSHUL','DOCTOR','ASDFG','17-JUN-95');
INSERT INTO EMPLOYEE VALUES(10020,'DR RAHUL','DOCTOR','ASDFG','25-JAN-98');
INSERT INTO EMPLOYEE VALUES(10021,'DR AYUSH','DOCTOR','ASDFG','30-JUN-99');
INSERT INTO EMPLOYEE VALUES(10022,'DR SHAKIL','DOCTOR','ASDFG','01-OCT-05');
INSERT INTO EMPLOYEE VALUES(10023,'DR SAHIL','DOCTOR','ASDFG','10-JUL-99');
INSERT INTO EMPLOYEE VALUES(10024,'DR NANCY','DOCTOR','ASDFG','12-SEP-02');
INSERT INTO EMPLOYEE VALUES(10025,'DR AKASH','DOCTOR','ASDFG','15-SEP-15');
INSERT INTO EMPLOYEE VALUES(10026,'DR ATIF','DOCTOR','ASDFG','19-SEP-05');
INSERT INTO EMPLOYEE VALUES(10027,'DR RUDRASHISH','DOCTOR','ASDFG','25-OCT-98');
INSERT INTO EMPLOYEE VALUES(10028,'DR GOVIND','DOCTOR','ASDFG','02-JAN-99');
INSERT INTO EMPLOYEE VALUES(10029,'MAYANK','ADMIN','ASDFG','02-JAN-99');
INSERT INTO EMPLOYEE VALUES(10030,'SANJEET','ADMIN','ASDFG','02-JAN-99');
INSERT INTO EMPLOYEE VALUES(10031,'GOPAL','ADMIN','ASDFG','02-JAN-99');
INSERT INTO EMPLOYEE VALUES(10032,'SUBHO','ADMIN','ASDFG','02-JAN-99');
INSERT INTO EMPLOYEE VALUES(10033,'PRIYA','RECEPTIONIST','QWERTY','02-JAN-99');
INSERT INTO EMPLOYEE VALUES(10034,'AISHWARIYA','RECEPTIONIST','QWERTY','02-JAN-99');
INSERT INTO EMPLOYEE VALUES(10035,'ANKITA','RECEPTIONIST','QWERTY','10-FEB-01');
INSERT INTO EMPLOYEE VALUES(10036,'DYUTI','RECEPTIONIST','QWERTY','15-FEB-02');
INSERT INTO EMPLOYEE VALUES(10037,'AKANKSHA','RECEPTIONIST','QWERTY','26-SEP-01');
INSERT INTO EMPLOYEE VALUES(10038,'RUPAMA','RECEPTIONIST','QWERTY','11-JAN-02');
INSERT INTO EMPLOYEE VALUES(10039,'CHANDRAYEE','RECEPTIONIST','QWERTY','15-OCT-03');
INSERT INTO EMPLOYEE VALUES(10040,'AKRITI','RECEPTIONIST','QWERTY','25-MAR-02');
INSERT INTO EMPLOYEE VALUES(10041,'SWATI','NURSE','QWERTY','25-MAR-02');
INSERT INTO EMPLOYEE VALUES(10042,'LAXMI','NURSE','QWERTY','28-APR-02');
INSERT INTO EMPLOYEE VALUES(10043,'RANI','NURSE','QWERTY','15-MAY-03');
INSERT INTO EMPLOYEE VALUES(10044,'SONI','NURSE','QWERTY','14-JUN-04');
INSERT INTO EMPLOYEE VALUES(10045,'PUJA','NURSE','QWERTY','07-JUL-04');
INSERT INTO EMPLOYEE VALUES(10046,'NEHA','NURSE','QWERTY','09-AUG-02');
INSERT INTO EMPLOYEE VALUES(10047,'RESHMI','NURSE','QWERTY','08-SEP-03');
INSERT INTO EMPLOYEE VALUES(10048,'RUPALI','NURSE','QWERTY','16-OCT-04');
INSERT INTO EMPLOYEE VALUES(10049,'VIJAY','STAFF','QWERTY','16-OCT-04');
INSERT INTO EMPLOYEE VALUES(10050,'ANKIT','STAFF','QWERTY','16-OCT-04');
INSERT INTO EMPLOYEE VALUES(10051,'VIVEK','STAFF','QWERTY','16-OCT-04');
INSERT INTO EMPLOYEE VALUES(10052,'SUMIT','STAFF','QWERTY','16-OCT-04');
commit;

```

TABLE NAME-DEPARTMENT

```

INSERT INTO DEPARTMENT VALUES(40001,'CARDIOLOGY');
INSERT INTO DEPARTMENT VALUES(40002,'ORTHOPEDICS');

```

```

INSERT INTO DEPARTMENT VALUES(40003,'ONCOLOGY');
INSERT INTO DEPARTMENT VALUES(40004,'GASTROENTEROLOGY');
INSERT INTO DEPARTMENT VALUES(40005,'NEUROSCIENCE');
INSERT INTO DEPARTMENT VALUES(40006,'NEPHROLOGY');
INSERT INTO DEPARTMENT VALUES(40007,'CRITICAL CARE');
INSERT INTO DEPARTMENT VALUES(40008,'PEDIATRICS');
INSERT INTO DEPARTMENT VALUES(40009,'BARIATRIC SURGERY');
INSERT INTO DEPARTMENT VALUES(40010,'PLASTIC SURGERY');
INSERT INTO DEPARTMENT VALUES(40011,'COLORECTAL SURGERY');
INSERT INTO DEPARTMENT VALUES(40012,'ENT');
INSERT INTO DEPARTMENT VALUES(40013,'DENTAL');
INSERT INTO DEPARTMENT VALUES(40014,'EDIOPATHY');
commit;

```

TABLE NAME-DOCTOR

```

INSERT INTO DOCTOR VALUES(20001,'DR RAMANAND',40001,10001,2000);
INSERT INTO DOCTOR VALUES(20002,'DR JOSHI',40001,10002,1500);
INSERT INTO DOCTOR VALUES(20003,'DR BANSALI',40002,10003,2000);
INSERT INTO DOCTOR VALUES(20004,'DR KAUSHIK',40002,10004,1500);
INSERT INTO DOCTOR VALUES(20005,'DR GOENKA',40003,10005,2500);
INSERT INTO DOCTOR VALUES(20006,'DR MANOJ',40003,10006,2000);
INSERT INTO DOCTOR VALUES(20007,'DR ROHIT',40004,10007,2500);
INSERT INTO DOCTOR VALUES(20008,'DR RAKESH',40004,10008,2200);
INSERT INTO DOCTOR VALUES(20009,'DR ANAND',40005,10009,1800);
INSERT INTO DOCTOR VALUES(20010,'DR ABHISHEK',40005,10010,2000);
INSERT INTO DOCTOR VALUES(20011,'DR SHETAL',40006,10011,1500);
INSERT INTO DOCTOR VALUES(20012,'DR MAYA',40006,10012,2000);
INSERT INTO DOCTOR VALUES(20013,'DR DEEPAK',40007,10013,1800);
INSERT INTO DOCTOR VALUES(20014,'DR DIPANKAR',40007,10014,2000);
INSERT INTO DOCTOR VALUES(20015,'DR ANKITA',40008,10015,2500);
INSERT INTO DOCTOR VALUES(20016,'DR NEHA',40008,10016,3000);
INSERT INTO DOCTOR VALUES(20017,'DR RAJ',40009,10017,2200);
INSERT INTO DOCTOR VALUES(20018,'DR PIYUSH',40009,10018,2200);
INSERT INTO DOCTOR VALUES(20019,'DR ANSHUL',40010,10019,1800);
INSERT INTO DOCTOR VALUES(20020,'DR RAHUL',40010,10020,3000);
INSERT INTO DOCTOR VALUES(20021,'DR AYUSH',40011,10021,3500);
INSERT INTO DOCTOR VALUES(20022,'DR SHAKIL',40011,10022,3200);
INSERT INTO DOCTOR VALUES(20023,'DR SAHIL',40012,10023,4000);
INSERT INTO DOCTOR VALUES(20024,'DR NANCY',40012,10024,2800);
INSERT INTO DOCTOR VALUES(20025,'DR AKASH',40013,10025,1800);
INSERT INTO DOCTOR VALUES(20026,'DR ATIF',40013,10026,1700);
INSERT INTO DOCTOR VALUES(20027,'DR RU DRASHISH',40014,10027,4000);
INSERT INTO DOCTOR VALUES(20028,'DR GOVIND',40014,10028,3500);
commit;

```

TABLE NAME-TEST

```

INSERT INTO TEST VALUES(50001,'ECG',4000);
INSERT INTO TEST VALUES(50002,'ENDOSCOPY',5000);
INSERT INTO TEST VALUES(50003,'XRAY',6000);

```

```

INSERT INTO TEST VALUES(50004,'BLOOD TEST',2000);
INSERT INTO TEST VALUES(50005,'UREA TEST',8000);
INSERT INTO TEST VALUES(50006,'CT SCAN',4000);
INSERT INTO TEST VALUES(50007,'MRI SCAN',8000);
INSERT INTO TEST VALUES(50008,'BIOPSY',3000);
INSERT INTO TEST VALUES(50009,'AUDIOMETRY',9000);
INSERT INTO TEST VALUES(50010,'BLOOD PRESSURE',1000);
commit;

```

TABLE NAME- ROOM

```

INSERT INTO ROOM VALUES(60001,'GENERAL',1200,0);
INSERT INTO ROOM VALUES(60002,'GENERAL',1200,0);
INSERT INTO ROOM VALUES(60005,'AC',3000,0);
INSERT INTO ROOM VALUES(60006,'AC',3000,0);
INSERT INTO ROOM VALUES(60003,'DELUXE',8000,0);
INSERT INTO ROOM VALUES(60004,'DELUXE',8000,0);
INSERT INTO ROOM VALUES(60007,'SUITE',10000,0);
INSERT INTO ROOM VALUES(60008,'SUITE',10000,0);
COMMIT;

```

TABLE NAME-BED

```

INSERT INTO BED VALUES(70001,60001,0);
INSERT INTO BED VALUES(70002,60001,0);
INSERT INTO BED VALUES(70003,60001,0);
INSERT INTO BED VALUES(70004,60001,0);
INSERT INTO BED VALUES(70005,60002,0);
INSERT INTO BED VALUES(70006,60002,0);
INSERT INTO BED VALUES(70007,60002,0);
INSERT INTO BED VALUES(70008,60002,0);
INSERT INTO BED VALUES(70009,60003,0);
INSERT INTO BED VALUES(70010,60003,0);
INSERT INTO BED VALUES(70011,60004,0);
INSERT INTO BED VALUES(70012,60004,0);
INSERT INTO BED VALUES(70013,60005,0);
INSERT INTO BED VALUES(70014,60005,0);
INSERT INTO BED VALUES(70015,60006,0);
INSERT INTO BED VALUES(70016,60006,0);
INSERT INTO BED VALUES(70017,60007,0);
INSERT INTO BED VALUES(70018,60008,0);
commit;

```

TABLE NAME-SALARY

```

INSERT INTO SALARY VALUES(10001,80001,50000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10002,80002,50000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10003,80003,50000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10004,80004,50000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10005,80005,50000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10006,80006,50000,'28-JAN-2016');

```

```

INSERT INTO SALARY VALUES(10007,80007,50000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10008,80008,50000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10009,80009,50000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10010,80010,50000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10011,80011,50000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10012,80012,50000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10013,80013,50000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10014,80014,50000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10015,80015,50000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10016,80016,50000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10017,80017,50000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10018,80018,50000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10019,80019,50000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10020,80020,50000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10021,80021,50000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10022,80022,50000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10023,80023,50000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10024,80024,50000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10025,80025,50000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10026,80026,50000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10027,80027,50000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10028,80028,50000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10029,80029,30000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10030,80030,30000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10031,80031,30000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10032,80032,30000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10033,80033,20000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10034,80034,20000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10035,80035,20000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10036,80036,20000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10037,80037,20000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10038,80038,20000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10039,80039,20000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10040,80040,20000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10041,80041,20000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10042,80042,20000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10043,80043,20000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10044,80044,20000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10045,80045,20000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10046,80046,20000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10047,80047,20000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10048,80048,20000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10049,80049,20000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10050,80050,20000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10051,80051,20000,'28-JAN-2016');
INSERT INTO SALARY VALUES(10052,80052,20000,'28-JAN-2016');
commit;

```

TABLE NAME-BLOOD BANK

```

INSERT INTO BLOOD_BANK VALUES('A+',300);

```

SUMAN ARCH




```

INSERT INTO BLOOD_BANK VALUES('A-',300);
INSERT INTO BLOOD_BANK VALUES('B+',300);
INSERT INTO BLOOD_BANK VALUES('B-',300);
INSERT INTO BLOOD_BANK VALUES('O+',300);
INSERT INTO BLOOD_BANK VALUES('O-',200);
INSERT INTO BLOOD_BANK VALUES('AB+',300);
INSERT INTO BLOOD_BANK VALUES('AB-',300);
commit;

```

TABLE NAME-OT

```

INSERT INTO OT(OT_ID,OT_CHARGE) VALUES(90001,30000);
INSERT INTO OT(OT_ID,OT_CHARGE) VALUES(90002,30000);
INSERT INTO OT(OT_ID,OT_CHARGE) VALUES(90003,30000);
INSERT INTO OT(OT_ID,OT_CHARGE) VALUES(90004,30000);
INSERT INTO OT(OT_ID,OT_CHARGE) VALUES(90005,30000);
INSERT INTO OT(OT_ID,OT_CHARGE) VALUES(90006,30000);
INSERT INTO OT(OT_ID,OT_CHARGE) VALUES(90007,30000);
INSERT INTO OT(OT_ID,OT_CHARGE) VALUES(90008,30000);
INSERT INTO OT(OT_ID,OT_CHARGE) VALUES(90009,30000);
INSERT INTO OT(OT_ID,OT_CHARGE) VALUES(90010,30000);
INSERT INTO OT(OT_ID,OT_CHARGE) VALUES(90011,30000);
INSERT INTO OT(OT_ID,OT_CHARGE) VALUES(90012,30000);
commit;

```

TABLE NAME-OPERATION

```

INSERT INTO OPERATION VALUES(11001,'ENDOCRINE SURGERY',30000,90001);
INSERT INTO OPERATION VALUES(11002,'GENERAL SURGERY',30000,90002);
INSERT INTO OPERATION VALUES(11003,'GYNECOLOGICAL SURGERY',30000,90003);
INSERT INTO OPERATION VALUES(11004,'HERNIA SURGERY',30000,90004);
INSERT INTO OPERATION VALUES(11005,'NEURO SURGERY',50000,90005);
INSERT INTO OPERATION VALUES(11006,'ORTHOPEDIC SURGERY',50000,90006);
INSERT INTO OPERATION VALUES(11007,'CARDIAC SURGERY',50000,90007);
INSERT INTO OPERATION VALUES(11008,'PEDIATRIC SURGERY',50000,90008);
INSERT INTO OPERATION VALUES(11009,'PLASTIC SURGERY',50000,90009);
INSERT INTO OPERATION VALUES(11010,'THORACIC SURGERY',80000,90010);
INSERT INTO OPERATION VALUES(11011,'TRAUMA SURGERY',80000,90011);
INSERT INTO OPERATION VALUES(11012,'ROBOTIC SURGERY',80000,90012);
commit;

```

TABLE NAME-ROUTINE

```

INSERT INTO ROUTINE VALUES(20001,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0);
INSERT INTO ROUTINE VALUES(20002,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1);
INSERT INTO ROUTINE VALUES(20003,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0);
INSERT INTO ROUTINE VALUES(20004,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1);
INSERT INTO ROUTINE VALUES(20005,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0);
INSERT INTO ROUTINE VALUES(20006,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1);
INSERT INTO ROUTINE VALUES(20007,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0);
INSERT INTO ROUTINE VALUES(20008,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1);

```



```

        MESSAGE('ACCESS DENIED');
MESSAGE('ACCESS DENIED');
END;
END;
2.)BED ----> BLOCK6-----> PUSH_BUTTON9 ---->WHEN-BUTTON-PRESSED ---->
COMMIT_FORM;
3.) BED ----> BLOCK6-----> PUSH_BUTTON7 ---->WHEN-BUTTON-PRESSED ---->
COMMIT_FORM;
4.) BED ----> BLOCK6-----> PUSH_BUTTON10 ---->WHEN-BUTTON-PRESSED ---->
GO_BLOCK('BED');
EXECUTE_QUERY;
5.) BED ----> BLOCK6-----> PUSH_BUTTON8 ---->WHEN-BUTTON-PRESSED ---->
GO_BLOCK('BED');
DELETE_RECORD;
COMMIT;
6.) BED ----> BLOCK6-----> PUSH_BUTTON11 ---->WHEN-BUTTON-PRESSED ---->
GO_BLOCK('BED');
CLEAR_BLOCK(NO_VALIDATE);
7.)BLOOD_BANK ----> BLOCK6 -----> PUSH_BUTTON8 ---> WHEN-BUTTON-PRESSED ---->
COMMIT_FORM;
8.) BLOOD_BANK ----> BLOCK6 -----> PUSH_BUTTON9 ---> WHEN-BUTTON-PRESSED ---->
GO_BLOCK('BLOOD_BANK');
EXECUTE_QUERY;
9.) BLOOD_BANK ----> BLOCK6 -----> PUSH_BUTTON11 ---> WHEN-BUTTON-PRESSED ---->
GO_BLOCK('BLOOD_BANK');
CLEAR_BLOCK(NO_VALIDATE);
10.) DEPARTMENT ---> BLOCK8 ---> PUSH_BUTTON12 ---->WHEN-BUTTON-PRESSED---->
COMMIT_FORM;
11.) DEPARTMENT ---> BLOCK8 ---> PUSH_BUTTON9 ---->WHEN-BUTTON-PRESSED---->
COMMIT_FORM;
12.) DEPARTMENT ---> BLOCK8 ---> PUSH_BUTTON10 ---->WHEN-BUTTON-PRESSED---->
GO_BLOCK('DEPARTMENT');
EXECUTE_QUERY;
13.) DEPARTMENT ---> BLOCK8 ---> PUSH_BUTTON13 ---->WHEN-BUTTON-PRESSED---->
GO_BLOCK('DEPARTMENT');
CLEAR_BLOCK(NO_VALIDATE);
14.) DEPARTMENT ---> BLOCK8 ---> PUSH_BUTTON11 ---->WHEN-BUTTON-PRESSED---->
GO_BLOCK('DEPARTMENT');
DELETE_RECORD;
COMMIT;
15.)DOCTOR--->BLOCK6--->PUSH_BUTTON7-----> WHEN-BUTTON-PRESSED-----> COMMIT_FORM;
16.) DOCTOR--->BLOCK6--->PUSH_BUTTON8-----> WHEN-BUTTON-PRESSED-----> COMMIT_FORM;
17.) DOCTOR--->BLOCK6--->PUSH_BUTTON9-----> WHEN-BUTTON-PRESSED----->
GO_BLOCK('DOCTOR');
EXECUTE_QUERY;
18.) DOCTOR--->BLOCK6--->PUSH_BUTTON10-----> WHEN-BUTTON-PRESSED----->

```

```

GO_BLOCK('DOCTOR');
DELETE_RECORD;
COMMIT;
19.) DOCTOR--->BLOCK6---->PUSH_BUTTON10-----> WHEN-BUTTON-PRESSED----->
GO_BLOCK('DOCTOR');
CLEAR_BLOCK('NO_VALIDATE');
20.) EMG_PATIENT--->EMG_PATIENT---->PUSH_BUTTON16-----> WHEN-BUTTON-PRESSED----->
GO_BLOCK('EMG_PATIENT');
COMMIT_FORM;
CLEAR_BLOCK;
21.)PATIENT_REGISTER---->PATIENT---->P_NAME----->POST-TEXT-ITEM----->
DECLARE
    VAR_TRANSACTION NUMBER;
    BEGIN
FOR I IN (SELECT P_ID FROM PATIENT WHERE P_ID=( SELECT MAX(P_ID) FROM PATIENT))
LOOP
    VAR_TRANSACTION :=I.P_ID;
END LOOP;
VAR_TRANSACTION:= VAR_TRANSACTION + 1;
:P_ID := NVL(VAR_TRANSACTION,1000);
END;
22.) PATIENT_REGISTER---->PATIENT---->P_DOB----->POST-TEXT-ITEM----->
BEGIN
:P_AGE:=ROUND((TO_CHAR(SYSDATE-:P_DOB)/365),0);
END;
23.)PATIENT_REGISTER---->PATIENT---->PUSH_BUTTON26----->WHEN-BUTTON-PRESSED----->
DECLARE
    A NUMBER;
BEGIN
    IF(:TEXT_ITEM27 IS NULL) THEN
        MESSAGE('PLEASE SELECT ROOM TYPE FIRST');
    ELSE
        A:=ALLOCATE_BED(:TEXT_ITEM27);
        IF A=0 THEN
            MESSAGE('ROOM IS FULL');
            MESSAGE('ROOM IS FULL');
        ELSE
            MESSAGE('ROOM IS ALLOCATED');
            :BED_ID:=A;
FOR I IN (SELECT ROOM_ID FROM BED WHERE BED_ID=A)
LOOP
    :ROOM_ID := I.ROOM_ID;
END LOOP;
END IF;
END IF;
END;

```

```

COMMIT;
EXCEPTION
    WHEN OTHERS THEN
        MESSAGE('ALL ROOMS ARE BOOKED');
        MESSAGE('ALL ROOMS ARE BOOKED');
24.) PATIENT_REGISTER---->PATIENT----->PUSH_BUTTON18----->WHEN-BUTTON-PRESSED----->
BEGIN
    GO_BLOCK('PATIENT');
COMMIT;
CLEAR_BLOCK('PATIENT');
EXCEPTION WHEN OTHERS THEN
    NULL;
END;
25.) EMG_PATIENT_SERVICES ---> BLOCK3 ----->TEXT_ITEM50 ----->POST-TEXT-ITEM ----->
BEGIN
    FOR I IN (SELECT COUNT(*) C FROM EMG_PATIENT WHERE EP_ID=:TEXT_ITEM50)
    LOOP
        IF (I.C=0) THEN
            MESSAGE('PLEASE ENTER A VALID EMG PATIENT ID');
            MESSAGE('PLEASE ENTER A VALID EMG PATIENT ID');
        END IF;
    END LOOP;
END;
26.) EMG_PATIENT_SERVICES ---> BLOCK3 ----->PUSH_BUTTON49 ----->WHEN-BUTTON-PRESSED ----->
DECLARE
    N1 NUMBER;
    N2 NUMBER;
    N3 NUMBER;
    FLAG NUMBER;

BEGIN
    N1:= :TEXT_ITEM32;
    N2:= :TEXT_ITEM44;
    N3 :=(N1-1) * 4 + N2;
    FLAG := DOCTOR_INFO_EMG(:DISPLAY_ITEM5, :TEXT_ITEM50, N3);
    IF(FLAG = -1) THEN
        MESSAGE('NO DOCTOR IS AVAILABLE');
        MESSAGE('NO DOCTOR IS AVAILABLE');
    ELSIF(FLAG= 0) THEN
        MESSAGE('YOU HAVE ALREADY REGISTERED');

        MESSAGE('YOU HAVE ALREADY REGISTERED');
    ELSIF(FLAG=1) THEN
        MESSAGE('DOCTOR IS BOOKED');
        MESSAGE('DOCTOR IS BOOKED');

```

```

DECLARE
    DOCID1 NUMBER;
BEGIN
FOR I IN (SELECT DC_ID FROM ABC)
LOOP
    DOCID1 :=I.DC_ID;

END LOOP;
DELETE FROM ABC;
STANDARD.COMMIT;
:TEXT_ITEM82:=DOCID1;
DECLARE
    DOCNAME VARCHAR2(30);
BEGIN
FOR I IN (SELECT DOC_NAME FROM DOCTOR WHERE DOC_ID = DOCID1)
LOOP
    DOCNAME:=I.DOC_NAME;
END LOOP;
:TEXT_ITEM86:=DOCNAME;
END;
END;
END IF;
EXCEPTION WHEN OTHERS THEN
    MESSAGE('YOU HAVE ALREADY REGISTERED');
    MESSAGE('YOU HAVE ALREADY REGISTERED');
END;
27.) EMG_PATIENT_SERVICES ---> BLOCK3 ----->PUSH_BUTTON76 ----->WHEN-BUTTON-PRESSED ----->
BEGIN
    INSERT INTO EMG_PATIENT_TEST VALUES(:TEXT_ITEM50,:TEXT_ITEM58,SYSDATE);
    MESSAGE(' REGISTRATION CONFIRMED');
    MESSAGE('REGISTRATION CONFIRMED');
    STANDARD.COMMIT;
EXCEPTION
    WHEN OTHERS THEN
        MESSAGE('ALREADY REGISTERED');
        MESSAGE('ALREADY REGISTERED');
    END;
28.) EMG_PATIENT_SERVICES ---> BLOCK3 ----->PUSH_BUTTON113 ----->WHEN-BUTTON-PRESSED ----->
DECLARE
    A NUMBER;
    B NUMBER;
    C NUMBER;
BEGIN
FOR I IN (SELECT OT_ID,OP_CHARGE FROM OPERATION WHERE OP_ID = :TEXT_ITEM100)
LOOP
    A := I.OT_ID;

```

```

:TEXT_ITEM114 := A;
B := I.OP_CHARGE;
END LOOP;
BEGIN
FOR I IN (SELECT OT_CHARGE FROM OT WHERE OT_ID = A)
LOOP
C := I.OT_CHARGE;
INSERT INTO BOOK_OTOP VALUES(:TEXT_ITEM50,A,:TEXT_ITEM100,SYSDATE,C,B);
STANDARD.COMMIT;
END LOOP;
END;
END;
29.)EMPLOYEE ----> BLOCK7-----> PUSH_BUTTON6 ---->WHEN-BUTTON-PRESSED ---->
COMMIT_FORM;
30.) EMPLOYEE ----> BLOCK7-----> PUSH_BUTTON8 ---->WHEN-BUTTON-PRESSED ---->
COMMIT_FORM;
31.) EMPLOYEE ----> BLOCK7-----> PUSH_BUTTON9 ---->WHEN-BUTTON-PRESSED ---->
GO_BLOCK(' EMPLOYEE ');
EXECUTE_QUERY;
32.) EMPLOYEE ----> BLOCK7-----> PUSH_BUTTON10 ---->WHEN-BUTTON-PRESSED ---->
GO_BLOCK(' EMPLOYEE ');
DELETE_RECORD;
COMMIT;
34.) EMPLOYEE ----> BLOCK7-----> PUSH_BUTTON22 ---->WHEN-BUTTON-PRESSED ---->
GO_BLOCK(' EMPLOYEE');
CLEAR_BLOCK(NO_VALIDATE);
35.) OPERATION ----> BLOCK6-----> PUSH_BUTTON7 ---->WHEN-BUTTON-PRESSED ---->
COMMIT_FORM;
36.) OPERATION ----> BLOCK6-----> PUSH_BUTTON8 ---->WHEN-BUTTON-PRESSED ---->
COMMIT_FORM;
37.) OPERATION ----> BLOCK6-----> PUSH_BUTTON9 ---->WHEN-BUTTON-PRESSED ---->
GO_BLOCK(' OPERATION ');
EXECUTE_QUERY;
38.) OPERATION ----> BLOCK6-----> PUSH_BUTTON10 ---->WHEN-BUTTON-PRESSED ---->
GO_BLOCK(' OPERATION ');
DELETE_RECORD;
COMMIT;
39.) OPERATION ----> BLOCK6-----> PUSH_BUTTON11 ---->WHEN-BUTTON-PRESSED ---->
GO_BLOCK(' OPERATION');
CLEAR_BLOCK(NO_VALIDATE);
40.) OT ----> BLOCK6-----> PUSH_BUTTON7 ---->WHEN-BUTTON-PRESSED ---->
COMMIT_FORM;
41.) OT ----> BLOCK6-----> PUSH_BUTTON8 ---->WHEN-BUTTON-PRESSED ---->
COMMIT_FORM;
42.) OT ----> BLOCK6-----> PUSH_BUTTON9 ---->WHEN-BUTTON-PRESSED ---->
GO_BLOCK(' OT ');

```

```

EXECUTE_QUERY;
43.) OT ----> BLOCK6-----> PUSH_BUTTON10 ---->WHEN-BUTTON-PRESSED ---->
GO_BLOCK(' OT ');
DELETE_RECORD;
COMMIT;
44.) OT ----> BLOCK6-----> PUSH_BUTTON11 ---->WHEN-BUTTON-PRESSED ---->
GO_BLOCK(' OT ');
CLEAR_BLOCK(NO_VALIDATE);
45.) ROOM ----> BLOCK6-----> PUSH_BUTTON9 ---->WHEN-BUTTON-PRESSED ---->
COMMIT_FORM;
46.) ROOM ----> BLOCK6-----> PUSH_BUTTON7 ---->WHEN-BUTTON-PRESSED ---->
COMMIT_FORM;
47.) ROOM ----> BLOCK6-----> PUSH_BUTTON8 ---->WHEN-BUTTON-PRESSED ---->
GO_BLOCK(' ROOM ');
EXECUTE_QUERY;
48.) ROOM ----> BLOCK6-----> PUSH_BUTTON10 ---->WHEN-BUTTON-PRESSED ---->
GO_BLOCK(' ROOM ');
DELETE_RECORD;
COMMIT;
49.) ROOM ----> BLOCK6-----> PUSH_BUTTON11 ---->WHEN-BUTTON-PRESSED ---->
GO_BLOCK(' ROOM ');
CLEAR_BLOCK(NO_VALIDATE);
50.) ROUTINE ----> BLOCK3-----> PUSH_BUTTON4 ---->WHEN-BUTTON-PRESSED ---->
COMMIT_FORM;
51.) ROUTINE ----> BLOCK3-----> PUSH_BUTTON5 ---->WHEN-BUTTON-PRESSED ---->
COMMIT_FORM;
52.) ROUTINE ----> BLOCK3-----> PUSH_BUTTON6 ---->WHEN-BUTTON-PRESSED ---->
GO_BLOCK(' ROUTINE ');
EXECUTE_QUERY;
53.) ROUTINE ----> BLOCK3-----> PUSH_BUTTON7 ---->WHEN-BUTTON-PRESSED ---->
GO_BLOCK(' ROUTINE ');
DELETE_RECORD;
COMMIT;
54.) ROUTINE ----> ROUTINE-----> PUSH_BUTTON8 ---->WHEN-BUTTON-PRESSED ---->
GO_BLOCK(' ROUTINE ');
CLEAR_BLOCK(NO_VALIDATE);
55.)SALARY ----> BLOCK6-----> PUSH_BUTTON7 ---->WHEN-BUTTON-PRESSED ---->
COMMIT_FORM;
56.) SALARY ----> BLOCK6-----> PUSH_BUTTON8----->WHEN-BUTTON-PRESSED ---->
COMMIT_FORM;
57.) SALARY ----> BLOCK6-----> PUSH_BUTTON9 ---->WHEN-BUTTON-PRESSED ---->
GO_BLOCK(' SALARY ');
EXECUTE_QUERY;
58.) SALARY ----> BLOCK6-----> PUSH_BUTTON10 ---->WHEN-BUTTON-PRESSED ---->
GO_BLOCK(' SALARY ');
DELETE_RECORD;

```

```

COMMIT;
59.) SALARY ----> BLOCK6-----> PUSH_BUTTON12 ---->WHEN-BUTTON-PRESSED ---->
GO_BLOCK(' SALARY ');
CLEAR_BLOCK(NO_VALIDATE);
60.) TEST----> BLOCK6-----> PUSH_BUTTON7 ---->WHEN-BUTTON-PRESSED ---->
COMMIT_FORM;
61.) TEST ----> BLOCK6-----> PUSH_BUTTON8----->WHEN-BUTTON-PRESSED ---->
COMMIT_FORM;
62.) TEST ----> BLOCK6-----> PUSH_BUTTON9 ---->WHEN-BUTTON-PRESSED ---->
GO_BLOCK(' TEST ');
EXECUTE_QUERY;
63.) TEST ----> BLOCK6-----> PUSH_BUTTON10 ---->WHEN-BUTTON-PRESSED ---->
GO_BLOCK(' TEST ');
DELETE_RECORD;
COMMIT;
63.) TEST ----> BLOCK6-----> PUSH_BUTTON11 ---->WHEN-BUTTON-PRESSED ---->
GO_BLOCK(' TEST ');
CLEAR_BLOCK(NO_VALIDATE);
64.) PATIENT_SERVICES ----->BLOCK3 -----> TEXT_ITEM50 ----->POST-TEXT-ITEM----->
BEGIN
    FOR I IN (SELECT COUNT(*) C FROM PATIENT WHERE P_ID=:TEXT_ITEM50)
    LOOP
        IF (I.C=0) THEN
            MESSAGE('PLEASE ENTER A VALID PATIENT ID');
            MESSAGE('PLEASE ENTER A VALID PATIENT ID');
        END IF;
    END LOOP
END;
65.) PATIENT_SERVICES ----->BLOCK3 -----> PUSH_BUTTON49 ----->WHEN-BUTTON-PRESSED----->
DECLARE
    N1 NUMBER;
    N2 NUMBER;
    N3 NUMBER;
    FLAG NUMBER;
BEGIN
    N1:= :TEXT_ITEM32;
    N2:= :TEXT_ITEM44;
    N3 :=(N1-1) * 4 + N2;
    FLAG := DOCTOR_INFO(:DISPLAY_ITEM5, :TEXT_ITEM50, N3);
    IF(FLAG = -1) THEN
        MESSAGE('NO DOCTOR IS AVAILABLE');
        MESSAGE('NO DOCTOR IS AVAILABLE');
    ELSIF(FLAG= 0) THEN
        MESSAGE('YOU HAVE ALREADY REGISTERED');

        MESSAGE('YOU HAVE ALREADY REGISTERED');

```



```

        ELSIF(FLAG=1) THEN
        MESSAGE('DOCTOR IS BOOKED' || FLAG);
DECLARE
        DOCID1 NUMBER;
BEGIN
FOR I IN (SELECT DC_ID FROM ABC)
LOOP
        DOCID1 :=I.DC_ID;

END LOOP;
DELETE FROM ABC;
STANDARD.COMMIT;
:TEXT_ITEM82:=DOCID1;
DECLARE
        DOCNAME VARCHAR2(30);
BEGIN
FOR I IN (SELECT DOC_NAME FROM DOCTOR WHERE DOC_ID = DOCID1)
LOOP
        DOCNAME:=I.DOC_NAME;
END LOOP;
:TEXT_ITEM86:=DOCNAME;
END;
END;
END IF;
EXCEPTION WHEN OTHERS THEN
        MESSAGE('YOU HAVE ALREADY REGISTERED');
        MESSAGE('YOU HAVE ALREADY REGISTERED');
END;
65.) PATIENT_SERVICES ----->BLOCK3 ----> PUSH_BUTTON76 ----->WHEN-BUTTON-PRESSED----->
BEGIN
        INSERT INTO PATIENT_TEST VALUES(:TEXT_ITEM50,:TEXT_ITEM58,SYSDATE);
        MESSAGE(' REGISTRATION CONFIRMED');
        MESSAGE('REGISTRATION CONFIRMED');
        STANDARD.COMMIT;
EXCEPTION
        WHEN OTHERS THEN
        MESSAGE('ALREADY REGISTERED');
        MESSAGE('ALREADY REGISTERED');
        END;
66.) PATIENT_SERVICES ----->BLOCK3 ----> PUSH_BUTTON113 ----->WHEN-BUTTON-PRESSED----->
DECLARE
        A NUMBER;
        B NUMBER;
        C NUMBER;
BEGIN
FOR I IN (SELECT OT_ID,OP_CHARGE FROM OPERATION WHERE OP_ID = :TEXT_ITEM100)

```

```

LOOP
    A := I.OT_ID;
    :TEXT_ITEM114 := A;
    B := I.OP_CHARGE;
    END LOOP;
BEGIN
FOR I IN (SELECT OT_CHARGE FROM OT WHERE OT_ID = A)
LOOP
    C := I.OT_CHARGE;
    INSERT INTO BOOK_OTOP VALUES(:TEXT_ITEM50,A,:TEXT_ITEM100,SYSDATE,C,B);
    STANDARD.COMMIT;
    END LOOP;
END;
END;

```

67.)BILLING ----->BLOCK6----->PUSH_BUTTON8 ----->WHEN-BUTTON-PRESSED----->

```

DECLARE
    VAR_DEPOSIT NUMBER;
    VAR_TRANSACTION NUMBER;
    A NUMBER;
    VAR_ROOMID NUMBER;
    VAR_ROOMAMOUNT NUMBER;
    VAR_TESTTOTAL NUMBER;
    VAR_TOTALCONS NUMBER;
    VAR_TOTALOP NUMBER;
BEGIN
    A:=:TEXT_ITEM7;
    VAR_TESTTOTAL:=0;
    VAR_TOTALCONS:=0;
    VAR_TOTALOP:=0;
    FOR I IN (SELECT * FROM PATIENT WHERE P_ID=A)
        LOOP
            :DISPLAY_ITEM15:=I.P_NAME;
            :DISPLAY_ITEM18:=I.P_AGE;
            :DISPLAY_ITEM23:=I.P_SEX;
            :DISPLAY_ITEM24:=I.P_PHONENO;
            :DISPLAY_ITEM26:=I.P_DOB;
            :DISPLAY_ITEM28:=I.P_ADMIT_DATE;

            VAR_ROOMID:=I.ROOM_ID;

        END LOOP;

        FOR J IN (SELECT * FROM ROOM WHERE ROOM_ID=VAR_ROOMID)
            LOOP

```

```

:ROOM_TYPE:=J.ROOM_TYPE;
:ROOM_AMOUNT:=J.ROOM_AMOUNT;
VAR_ROOMAMOUNT:=J.ROOM_AMOUNT;
END LOOP;

FOR K IN ( SELECT CEIL(SYSDATE-P_ADMIT_DATE) AS DAYS FROM PATIENT WHERE P_ID=A)
LOOP
:DISPLAY_ITEM44:=K.DAYS;
END LOOP;
:DISPLAY_ITEM48:=NVL(VAR_ROOMAMOUNT,0)*NVL(:DISPLAY_ITEM44,0);

GO_BLOCK('TEST');
CLEAR_BLOCK(NO_VALIDATE);
FIRST_RECORD;

FOR SK IN ( SELECT P.T_ID X,P.TEST_DATE P,T.T_NAME Y,T.T_AMOUNT Z FROM PATIENT_TEST P,TEST T
WHERE P.P_ID=A AND P.T_ID=T.T_ID)
LOOP

:TEST.T_ID:=SK.X;
:TEST.T_NAME:=SK.Y;
:TEST.T_AMOUNT:=SK.Z;
:TEST.TEXT_ITEM84:=SK.P;
VAR_TESTTOTAL:=NVL(VAR_TESTTOTAL,0)+NVL(SK.Z,0);
NEXT_RECORD;
END LOOP;
FIRST_RECORD;
:DISPLAY_ITEM95:=VAR_TESTTOTAL;
:DISPLAY_ITEM97:=VAR_TESTTOTAL;
VAR_TESTTOTAL:=0;

GO_BLOCK('CONSULT_PATIENT');
CLEAR_BLOCK(NO_VALIDATE);
FIRST_RECORD;

FOR I IN (SELECT C.DOC_ID X,D.DOC_NAME Y ,D.DOC_CHARGE Z FROM CONSULT_PATIENT C,DOCTOR D
WHERE C.P_ID=A AND C.DOC_ID=D.DOC_ID)
LOOP

:CONSULT_PATIENT.DOC_ID:=I.X;
:CONSULT_PATIENT.DOCTOR_NAME:=I.Y;
:CONSULT_PATIENT.DOC_CHARGE:=I.Z;
VAR_TOTALCONS:=NVL(VAR_TOTALCONS,0)+NVL(I.Z,0);
NEXT_RECORD;
END LOOP;
FIRST_RECORD;

```

```

        :DISPLAY_ITEM111:=VAR_TOTALCONS;
:DISPLAY_ITEM113:=VAR_TOTALCONS;
VAR_TOTALCONS:=0;

GO_BLOCK('BOOK_OTOP');
CLEAR_BLOCK(NO_VALIDATE);
FIRST_RECORD;

FOR I IN (SELECT OP.OP_NAME A,B.OT_DATE B,B.OT_CHARGE C,B.OP_CHARGE D FROM OPERATION
OP,BOOK_OTOP B WHERE B.P_ID=A AND B.OP_ID=OP.OP_ID)
LOOP
    :BOOK_OTOP.OT_DATE:=I.B;
    :BOOK_OTOP.OT_CHARGE:=I.C;
    :BOOK_OTOP.OP_CHARGE:=I.D;
    :BOOK_OTOP.OP_NAME:=I.A;
    VAR_TOTALOP:=NVL(VAR_TOTALOP,0)+NVL(I.C,0)+NVL(I.D,0);
    NEXT_RECORD;
END LOOP;

FIRST_RECORD;
:DISPLAY_ITEM121:=VAR_TOTALOP;
:DISPLAY_ITEM122:=VAR_TOTALOP;
VAR_TOTALCONS:=0;

:DISPLAY_ITEM126:=
NVL(:DISPLAY_ITEM48,0)+NVL(:DISPLAY_ITEM97,0)+NVL(:DISPLAY_ITEM113,0)+NVL(:DISPLAY_ITEM122,0);

BEGIN
FOR I IN (SELECT BILL_ID FROM BILL WHERE BILL_ID=( SELECT MAX(BILL_ID) FROM BILL))
LOOP
    VAR_TRANSACTION :=I.BILL_ID;

END LOOP;
:TEXT_ITEM155:=NVL(VAR_TRANSACTION+1,7000);
BEGIN
FOR I IN ( SELECT SUM(DEPOSIT) B FROM BILL WHERE P_ID=:TEXT_ITEM7 GROUP BY P_ID)
LOOP
    VAR_DEPOSIT :=I.B;
END LOOP;
IF((:DISPLAY_ITEM126-nvl(VAR_DEPOSIT,0))<0) THEN
    :TEXT_ITEM160:=0;
ELSE
:TEXT_ITEM160:=:DISPLAY_ITEM126-nvl(VAR_DEPOSIT,0);
END IF;

```

```

END;

END;
END;
68.) BILLING ----->BLOCK6----->PUSH_BUTTON154 ----->WHEN-BUTTON-PRESSED----->
BEGIN
    DO_KEY('PRINT');
END;
69.) BILLING ----->ROOM_CHARGES----->PUSH_BUTTON159 ----->WHEN-BUTTON-PRESSED----->
DECLARE
    VAR_TRANSACTION NUMBER;
BEGIN
FOR I IN (SELECT BILL_ID FROM BILL WHERE BILL_ID=( SELECT MAX(BILL_ID) FROM BILL))
LOOP
    VAR_TRANSACTION :=I.BILL_ID;
END LOOP;
:TEXT_ITEM155:=NVL(VAR_TRANSACTION+1,7000);
IF(:TEXT_ITEM160= 0) THEN
    MESSAGE('BILL ALREADY PAID');
    MESSAGE('BILL ALREADY PAID');
ELSE
    INSERT INTO BILL
VALUES(:TEXT_ITEM155,:TEXT_ITEM7,:DISPLAY_ITEM15,:DISPLAY_ITEM48,:DISPLAY_ITEM97,:DISPLAY_ITEM122,:DISP
LAY_ITEM113,SYSDATE,:TEXT_ITEM160,:DISPLAY_ITEM138);
STANDARD.COMMIT;
:TEXT_ITEM160:=:TEXT_ITEM160-:DISPLAY_ITEM138;

END IF;
END;
70.) BILLING ----->ROOM_CHARGES----->RELEASE_PATIENT ----->WHEN-BUTTON-PRESSED----->
BEGIN
IF(:TEXT_ITEM160 = 0) THEN
    DELETE FROM BOOK_OTOP WHERE P_ID = :TEXT_ITEM7;
    UPDATE BED SET BED_STATUS=1 WHERE BED_ID=(SELECT BED_ID FROM PATIENT WHERE P_ID =
:TEXT_ITEM7);
    UPDATE PATIENT SET ROOM_ID = NULL, BED_ID=NULL WHERE P_ID=:TEXT_ITEM7;
    DELETE FROM CONSULT_PATIENT WHERE P_ID=:TEXT_ITEM7;
    DELETE FROM PATIENT_TEST WHERE P_ID=:TEXT_ITEM7;
    STANDARD.COMMIT;
        MESSAGE('PATIENT IS RELEASED. GET WELL SOON!');
        MESSAGE('PATIENT IS RELEASED. GET WELL SOON!');
ELSE
    MESSAGE('PLEASE PAY THE TOTAL BILL');
    MESSAGE('PLEASE PAY THE TOTAL BILL');
END IF;
END;

```

TRIGGERS ON MENU MODULES:-

```
1.) ADMIN_MENU -----> MENU1----->EMPLOYEE
BEGIN
    CALL_FORM('D:\EMPLOYEE', HIDE, DO_REPLACE, NO_QUERY_ONLY);
END;
2.) ADMIN_MENU -----> MENU1----->DEPARTMENT
BEGIN
    CALL_FORM('D:\DEPARTMENT', HIDE, DO_REPLACE, NO_QUERY_ONLY);
END;
3.) ADMIN_MENU -----> MENU1----->DOCTOR
BEGIN
    CALL_FORM('D:\DOCTOR', HIDE, DO_REPLACE, NO_QUERY_ONLY);
END;
4.) ADMIN_MENU -----> MENU1----->SALARY
BEGIN
    CALL_FORM('D:\SALARY', HIDE, DO_REPLACE, NO_QUERY_ONLY);
END;
5.) ADMIN_MENU -----> MENU1----->TEST
BEGIN
    CALL_FORM('D:\TEST', HIDE, DO_REPLACE, NO_QUERY_ONLY);
END;
6.) ADMIN_MENU -----> MENU1----->BLOOD BANK
BEGIN
    CALL_FORM('D:\BLOOD_BANK', HIDE, DO_REPLACE, NO_QUERY_ONLY);
END;
7.) ADMIN_MENU -----> MENU1----->ROOM
BEGIN
    CALL_FORM('D:\ROOM', HIDE, DO_REPLACE, NO_QUERY_ONLY);
END;
8.) ADMIN_MENU -----> MENU1----->BED
BEGIN
    CALL_FORM('D:\BED', HIDE, DO_REPLACE, NO_QUERY_ONLY);
END;
9.) ADMIN_MENU -----> MENU1----->OT
BEGIN
    CALL_FORM('D:\OT', HIDE, DO_REPLACE, NO_QUERY_ONLY);
END;
10.) ADMIN_MENU -----> MENU1----->ACCOUNTS
BEGIN
    CALL_FORM('D:\ACCOUNTS', HIDE, DO_REPLACE, NO_QUERY_ONLY);
END;
11.) ADMIN_MENU -----> MENU1----->LOGOUT
BEGIN
    CALL_FORM('D:\LOGIN_PAGE', HIDE, DO_REPLACE, NO_QUERY_ONLY);
END;
```

```

12) OPERATOR_MENU----->MENU1----->PATIENT REG.
BEGIN
    CALL_FORM('D:\PATIENT_REGISTER', HIDE, DO_REPLACE, NO_QUERY_ONLY);
END;
13.) OPERATOR_MENU----->MENU1----->PATIENT SERVICES
BEGIN
    CALL_FORM('D:\PATIENT_SERVICES', HIDE, DO_REPLACE, NO_QUERY_ONLY);
END;
14.) OPERATOR_MENU----->MENU1----->EMERGENCY PATIENT
BEGIN
    CALL_FORM('D:\EMG_PATIENT', HIDE, DO_REPLACE, NO_QUERY_ONLY);
END;
15.) OPERATOR_MENU----->MENU1----->EMG PATIENT SERVICES
BEGIN
    CALL_FORM('D:\EMG_PATIENT_SERVICES', HIDE, DO_REPLACE, NO_QUERY_ONLY);
END;
16.) OPERATOR_MENU----->MENU1----->BLOOD BANK
BEGIN
    CALL_FORM('D:\BLOOD_BANK', HIDE, NO_REPLACE, NO_QUERY_ONLY);
END;
17.) OPERATOR_MENU----->MENU1----->BILL GENERATION
BEGIN
    CALL_FORM('D:\BILLING', HIDE, DO_REPLACE, NO_QUERY_ONLY);
END;
18.) OPERATOR_MENU----->MENU1----->LOGOUT
BEGIN
    CALL_FORM('D:\LOGIN_PAGE', HIDE, DO_REPLACE, NO_QUERY_ONLY);
END;

```

Certificate

This is to certify that Mr. Mayank Kataruka of Heritage Institute Technology, Registration number: 131260110064, has successfully completed a project on Hospital Management System using Oracle Forms under the guidance of Mr. Suman Aich.

Suman Aich.

Globsyn Finishing School

SUMAN AICH



Certificate

This is to certify that Mr. Sanjeet Kumar of Heritage Institute Technology, Registration number: 131260110090, has successfully completed a project on Hospital Management System using Oracle Forms under the guidance of Mr. Suman Aich.

Suman Aich.

Globsyn Finishing School

SUMAN AICH



Certificate

This is to certify that Mr. Gopal Shaurya of Heritage Institute Technology, Registration number: 131260110048, has successfully completed a project on Hospital Management System using Oracle Forms under the guidance of Mr. Suman Aich.

Suman Aich.

Globsyn Finishing School

Suman Aich



Certificate

This is to certify that Mr. Subhadeep Koley of RCC Institute of Information Technology, Registration number: 141170110293, has successfully completed a project on Hospital Management System using Oracle Forms under the guidance of Mr. Suman Aich.

Suman Aich.

Globsyn Finishing School

SUMAN AICH

