

Ahmedabad University

DBMS Project Report

Project: Hospital Management System

Group Members:

Name	Enrolment Number
Anshul Mehta	AU1940275
Harsh Patel	AU1940114
Nihal Aggarwal	AU1940217

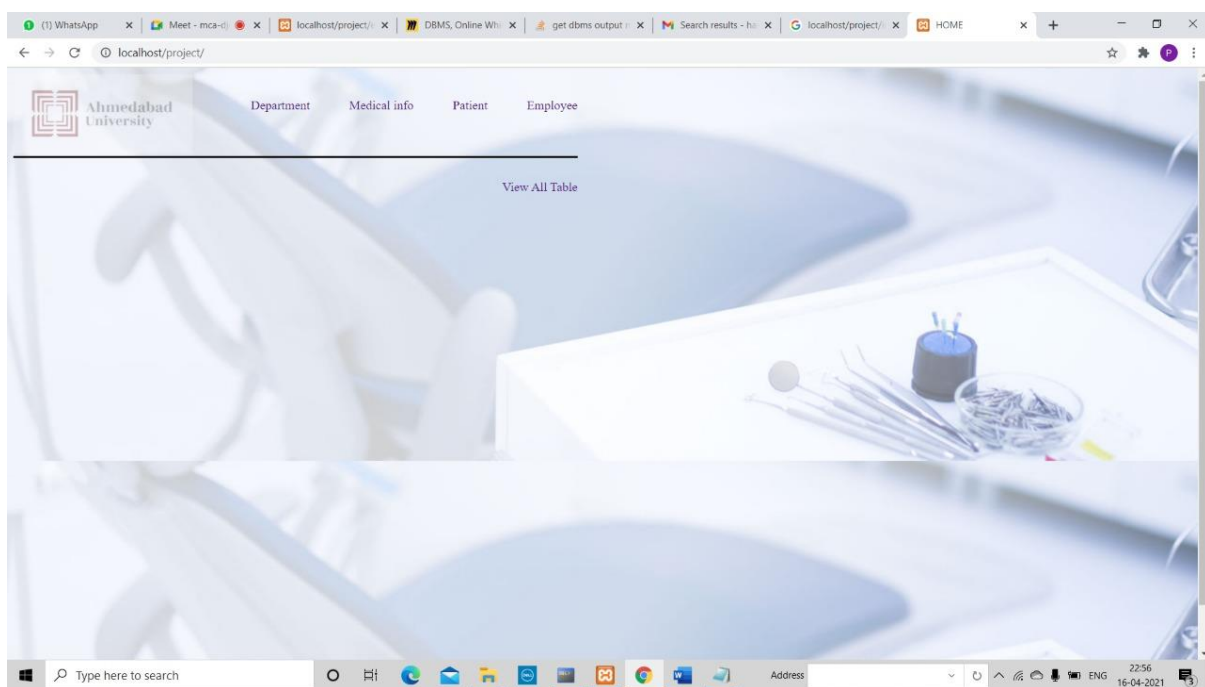
Description:

Our Project Definition is Hospital Management System. There is a ton of data that needs to be processed and retrieved on a daily basis for a massive organization like a hospital. There is details of patients and their data. Employees and their data. They also have to keep a track of inventory, departments, room allocation to patients. They also need to keep a track of all of this data and maintaining a database becomes a very integral part for their functioning. Thus we have tried to model at a small scale the Hospital Database Management System.

Features:

Our Project revolves around making a Hospital Management System. Our primary focus has been to comprehensively cover all the facets of a Hospital Management System. We have the function to Register a Patient, admit a Patient in a Room, Manage the Inventory of Hospital, Maintain Employee Records, Segregate the doctors into departments. We also have the feature of assigning doctors to patients, assigning treatments, maintaining pharmacy and medical cost records. We have also Implemented feature of Calculating Bill taking into consideration all the expenses of doctors' fees of diagnosis, treatment charges etc.

Overview Images:



Browser tabs: (2) WhatsApp, Course: CSE250 Database Mana..., Meet - mca-djef-cob, patient info, ahmedabad university - Google

Address bar: localhost/project/patient.php

Patient Information. 7

First Name :

Last Name :

Age:

Contact:

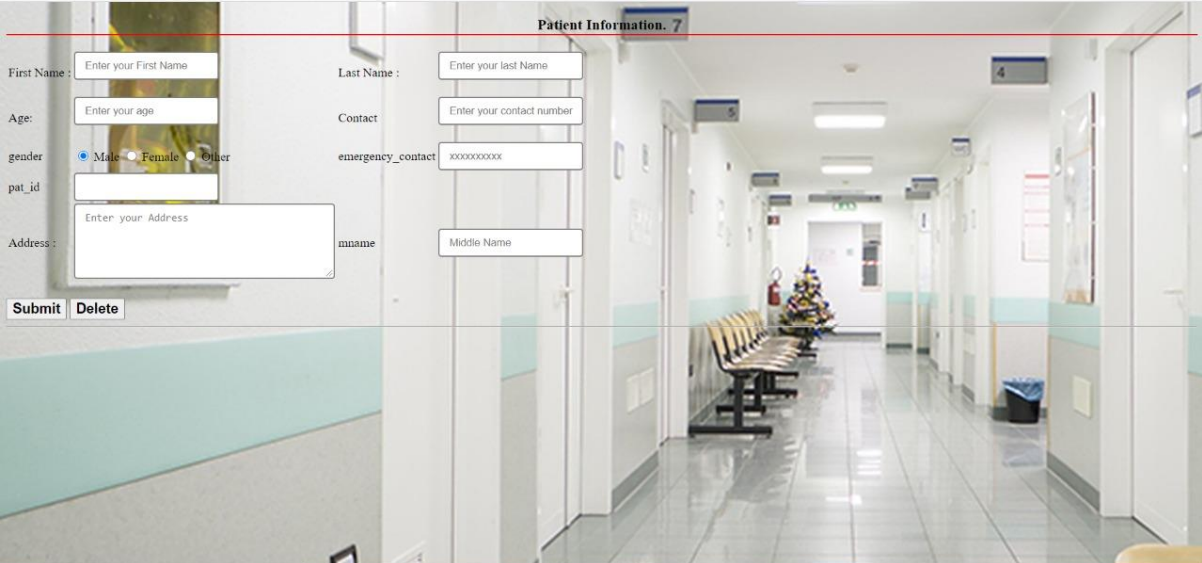
gender: ☒ Male ☐ Female ☐ Other

emergency_contact:

pat_id:

Address :

mname:



Windows taskbar: Type here to search, 23:03, 16-04-2021

Address bar: localhost/project/department.php


Department Information.

Department Name :

Department Id :

Head of department :

Inventory



meet.google.com is sharing your screen.

System Requirements and Implemented Functions:

Backend:

We have created the database as backend in Oracle Database.

Oracle Versions-11c, 12c,18c Community Editions are all Compatible.

Frontend:

The Frontend is done on PHP with XAMMP as the server and Oracle is Connected and Integrated with the Frontend Forms.

We have created CRUD functions for most of the tables.

Tables:

We have a made a CRUD page for almost all pages where we can perform operations on Tables.

We have made 17 tables in the Database namely:

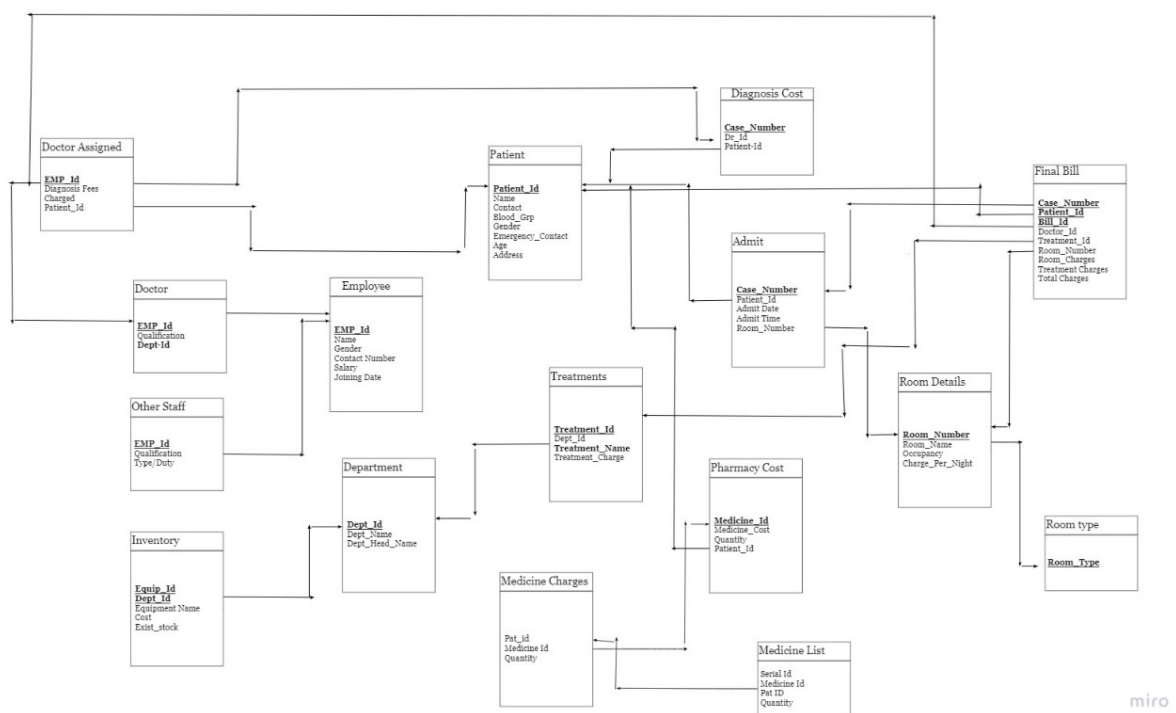
Table Name	The function
Patient	Patient Registration and Details
Employee	Employee Details
Type	Room Types like ICUS, OT etc.
Room Details	Room Details of each Room Number
Treatment Given	The treatment that patient undergoes
Department	Information of Different Departments
Doctor	Doctor Info
Other Staff	General Staff Info
Inventory	Inventory keeps track of med. Equipment
Treatments	List of treatments available dept. wise
Doctors Assigned	List of Doctors and Patients Assigned to them
Diagnosis	The diagnosis table
Admit	Table to admit patients into a room
Pharmacy	List of Medicines available
Medical Charge	The Medical Charge of Patient
Medicine List	List of all the medicines taken by a Patient
Bill	Final billing taking into account all of above expenses

Database Design:

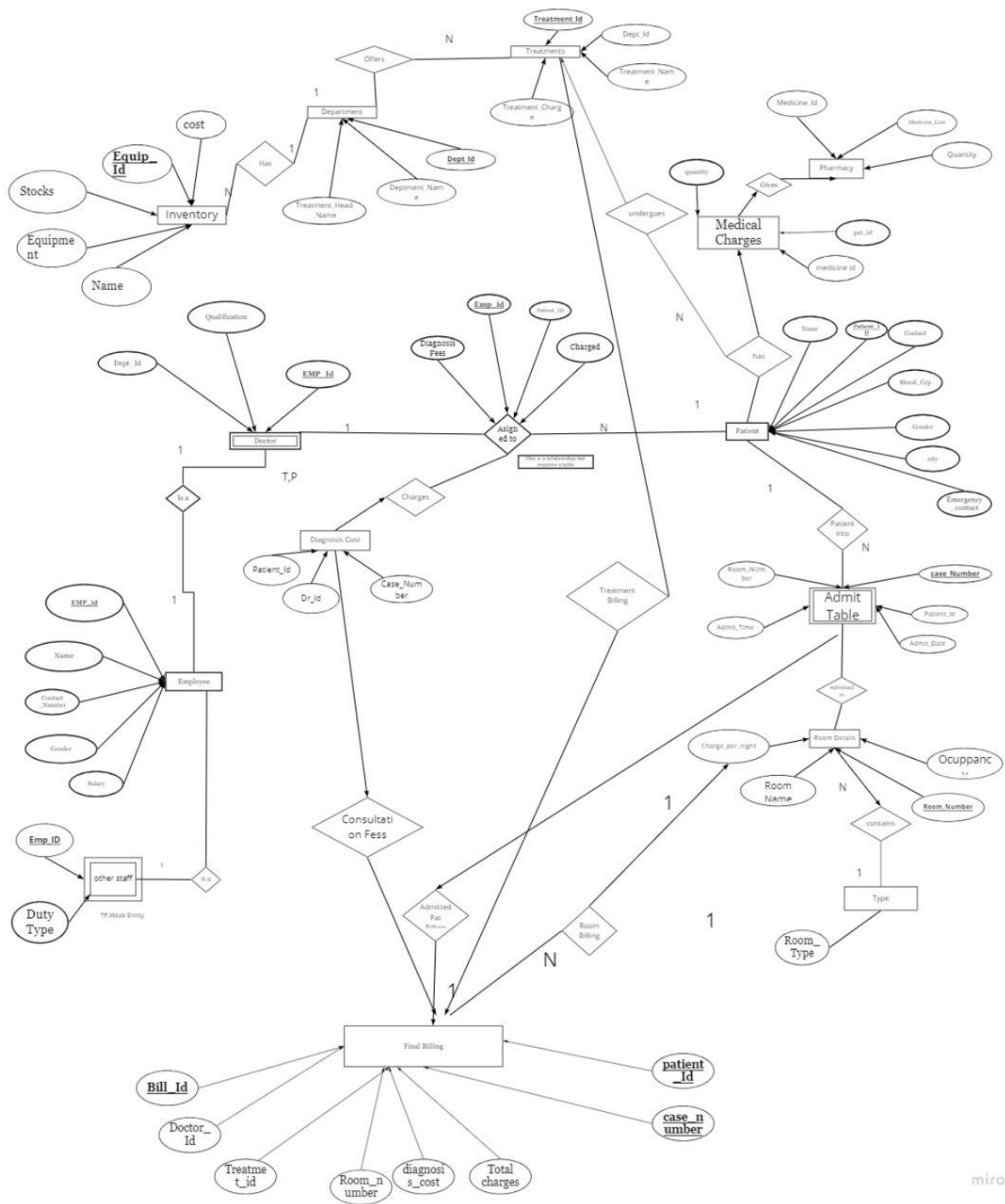
Here are the Schema and ER Diagram.

The List of Normalizations taken into account are mentioned in a separate section,

Schema:



E-R Diagram:



Normalizations and additional tables taken into account:

- 1) All Composite – are split into separate Attributes while implementing create commands
- 2) Tables like Room Details has attribute Room Type that has Partial Dependency on Room Number so We created a separate table that enumerates all the Room Types and then it can be directly referenced
- 3) Table Medical Charges has lot of MVAs (Multi Value Attributes) as a Patient can take lot of medicines and it is a variable number. We created a table Medical List that maps the Patient Id from Medical Charge and then that table can have a list of n number of medicines that are given to patient. There is also an accompanying trigger that inserts the data in Medical List after insertion in Medical Charge.
- 4) We have also checked for Partial and Transitive Dependencies and we found one in Doctor Assigned table which we solve by changing the primary key.
- 5) There is a relation between doctor and patient of assigned to which needs a table to store which doctor has been assigned to which patient.
- 6) Patient Undergoes Treatment is also a relationship that is put in a table.

Queries:

For execution of simple Query execution, we have put a separate page on the Frontend. The Page acts in a manner that it has buttons for named after all the tables. Whenever a Table button is pressed the Data of the Table is Displayed. This helps in retrieving data from all the tables at one place. The data of any table can be read from the page:

localhost/project/SelectAll.php

Successfully connected with Oracle DB

10001	Bhavna	Alkesh	Kumar	Female	1234567890	45	4567891234	103, Arindam Apartments, Opp to Manohar Circle, East Durganagar, Jamshedpur
10002	Rakesh	Shruti	Patel	Male	1452365478	38	1254789256	Block 304, Mathru Chaya East Andheri, Maharashtra
10003	Narail	Aashore	Sharma	Male	1745624589	48	5214756328	104, Dhanaajaya Flats, Opp to PR Circle, Dadar, Maharashtra
10004	akash	Rupesh	Trivedi	Male	4512679635	31	5642173458	Railway Block 302, Fulwadi Chowk, Porbandar, Gujarat
10005	Rupa	Zeel	Shah	Female	7456213578	29	7894521365	Airport Road, Ashvinayak Flats Block no 531, Ahmedabad, Gujarat
10006	Yug	Anantrai	Dave	Male	4579632158	21	7456931245	Kalrav Apartments, Block No 321, Opp to Post Office, Dwarka Nagar, Porbandar, Gujarat
10007	Yogesh	Bhavya	Shah	Male	4215694578	26	4785321689	Block No 124, West Borivili, Sky High Apartments, Maharashtra
10008	Klushi	Sagar	Rathod	Female	4215635648	28	4752963578	Bungalow No 003, Indiranagar, North West Dombivili, Maharashtra
10009	Anjali	Rupesh	Thakur	Female	4896523674	24	4521369856	302, B Wing, Roopeshwarji Flats, Old Rajinder Nagar, Surat, Gujarat
10010	Yash	Aditya	Zala	Male	4562398725	32	7895462156	Iscon Valley, C Wing, Flat No 204, Subhashnagar, Vadodara, Gujarat
10011	Nihal	Rajesh	Aggarwal	M	7984226960	19	4657895421	103, Ashvinayak 3, Bh
10012	Rajesh	aaa	Aggarwal	M	9879009883	45	877966445	jdscyjdcydydw
10012	Harsh	nwslw	Patel	M	848444454	19	5465464654	jewnkjewnkj
45454	Om	kji	Praksh	M	78979854	25	8487468	jawejlfew

Printing All Data Entries

Patient Employee Room Type Room Details Department Doctor Inventory

Other Staff Treatments Diagnosis Cost Admission Pharmacy Medical List

meet.google.com is sharing your screen. Stop sharing Hide

localhost/project/SelectAll.php

Successfully connected with Oracle DB

D001	Dr Piyush	Udit	Kumar	25-APR-74	Male	29-AUG-99	7895462145	Block No 1, Sir T Hospital, Opp to Udhagam, Bhavnagar, Gujarat	60000
D002	Dr Shaysi	Harsh	Kamte	27-AUG-67	Male	21-JUN-97	4521635789	Block No 2, Sir T Hospital, Opp to Udhagam, Bhavnagar, Gujarat	65000
D003	Dr Anil	Ankur	Kurekh	21-APR-78	Female	19-SEP-01	4512365478	Block No 3, Sir T Hospital, Opp to Udhagam, Bhavnagar, Gujarat	70000
D004	Dr Anupam	Atal	Vaidya	16-MAR-70	Male	17-DEC-03	4578963214	Block No 4, Sir T Hospital, Opp to Udhagam, Bhavnagar, Gujarat	72000
D005	Dr Uttara	Bhavesh	Patel	04-SEP-67	Female	17-FEB-99	4587963214	Block No 5, Sir T Hospital, Opp to Udhagam, Bhavnagar, Gujarat	75000
D006	Dr Nipun	Antony	Kumar	13-AUG-75	Male	19-JUN-05	4754142548	Block No 6, Sir T Hospital, Opp to Udhagam, Bhavnagar, Gujarat	62000
D007	Dr Uttara	Bhavesh	Patel	29-JUL-78	Female	13-MAY-06	4756321458	Block No 7, Sir T Hospital, Opp to Udhagam, Bhavnagar, Gujarat	69000
D008	Dr Umar	Kumar	Bhavsar	09-JUN-85	Male	19-JUL-10	7456324514	Block No 8, Sir T Hospital, Opp to Udhagam, Bhavnagar, Gujarat	55000
D009	Dr Om	Jagdish	Paramhans	19-SEP-74	Male	16-MAR-98	7546988214	Block No 9, Sir T Hospital, Opp to Udhagam, Bhavnagar, Gujarat	71000
D010	Dr Prem	Ratan	Saraswat	17-APR-79	Male	05-SEP-00	7485214563	Block No 10, Sir T Hospital, Opp to Udhagam, Bhavnagar, Gujarat	64000
D011	Dr Nihal	Rajesh	Aggarwal	13-FEB-87	M	02-APR-21	4578965412	103, Ashvinayak 3	55000

Printing All Data Entries

Patient Employee Room Type Room Details Department Doctor Inventory

Other Staff Treatments Diagnosis Cost Admission Pharmacy Medical List

meet.google.com is sharing your screen. Stop sharing Hide

List of Triggers and Screenshots:

1) Trigger for Handling all the Tables to which Employee is a foreign key

create or replace trigger del_rec before delete on employee

for each row

begin

```
delete from doctor where doctor.emp_id=:old.emp_id;
```

```
delete from other_staff where other_staff.emp_id=:old.emp_id;
```

```
update doctor_assigned set dr_id=null where dr_id=:old.emp_id;
```

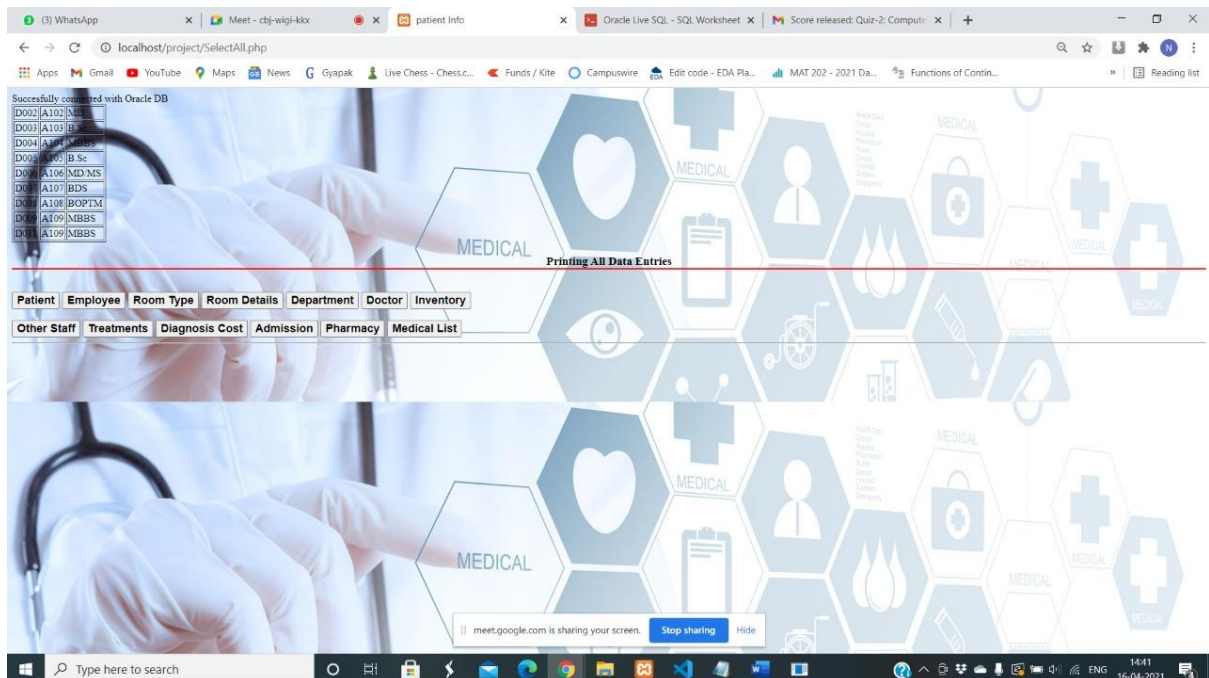
```
update diagnosis_cost set dr_id=null where dr_id=:old.emp_id;
```

end;

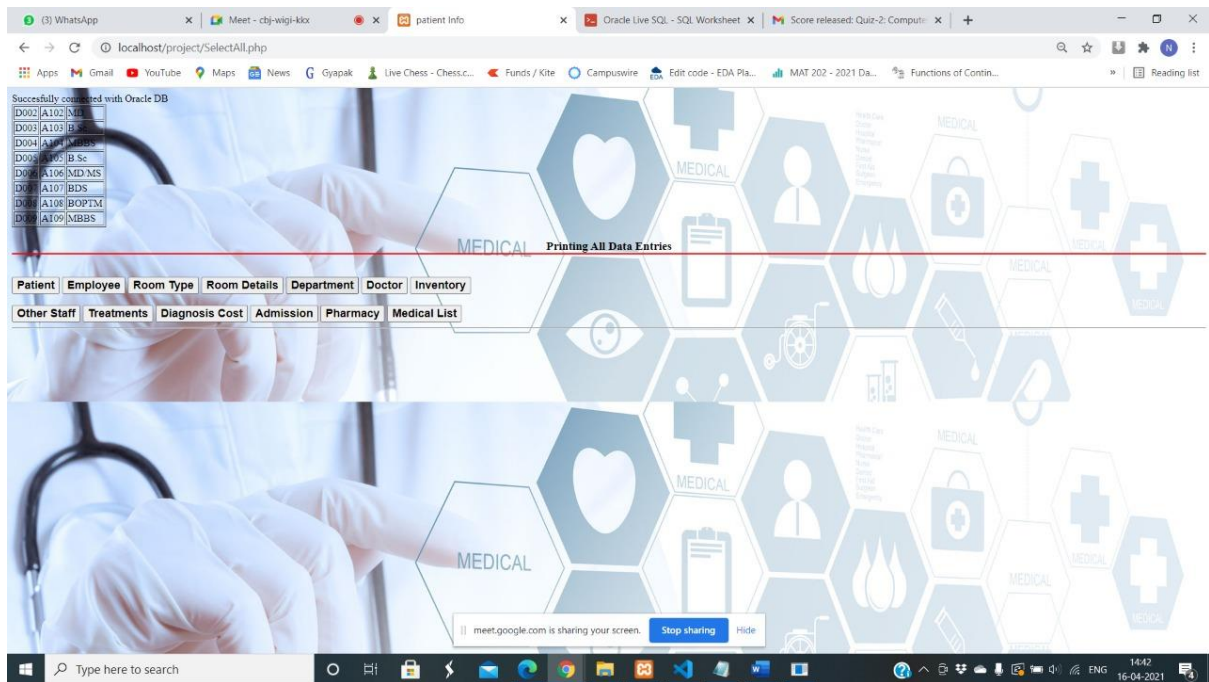
/

Screenshots:

The view of Doctor Table before delete on Employee



The view of doctor Table after delete on Employee



2) Trigger for Handling all child Tables where Patient is a Reference

create or replace trigger del_admitted_pat before delete on patient

for each row

begin

delete from admit where admit.patient_id=:old.pat_id;

update doctor_assigned set pat_id=null where pat_id=:old.pat_id;

update diagnosis_cost set pat_id=null where pat_id=:old.pat_id;

end;

/

Successfully connected with Oracle DB

10001	Bhavna	Akash	Kumar	Female	1234567890	45	4567891234	103, Arindam Apartments, Opp to Manohar Circle, East Durganagar, Jamshedpur
10002	Rakesh	Shubhi	Patel	Male	1452365478	38	1254789256	Block 304, Mathru Chaya East Andheri, Maharashtra
10003	Yugesh	Kishore	Sharma	Male	1745624589	48	5214756328	104, Dhananjaya Flats, Opp to PR Circle, Dadar, Maharashtra
10004	Aakash	Rupesh	Trivedi	Male	4512679635	31	5642173458	Railway Block 302, Fulwadi Chowk, Porbandar, Gujarat
10005	Rupa	Zeel	Shah	Female	7456213578	29	7894521365	Airport Road, Ashvinayak Flats Block no 531, Ahmedabad, Gujarat
10006	Yug	Anantrai	Dave	Male	4579632158	21	7456931245	Kalrav Apartments, Block No 721, Opp to Post Office, Dwarka Nagar, Porbandar, Gujarat
10007	Yogesh	Bhavya	Shah	Male	4215694578	26	4785321689	Block No 124, West Borivili, Sky High Apartments, Maharashtra
10008	Khushi	Sagar	Rathod	Female	4215635648	28	4752963578	Bungalow No 003, Indiranagar, North West Dombivili, Maharashtra
10009	Anjali	Rupesh	Thakur	Female	4896523674	24	4521369856	302, B Wing, Roopeshwarji Flats, Old Rajinder Nagar, Surat, Gujarat
10010	Yash	Aditya	Zala	Male	4562398725	32	7895462156	Iscon Valley, C Wing, Flat No 204, Subhashnagar, Vhododara, Gujarat
10013	Khushi	surya	Shah	F	7546521458	25	4521547856	nsnsnsnsn
10014	Nihal	Rajesh	Aggarwal	M	7984226960	19	4657895421	103, Astvinayak 3, Bh
10012	Harsh	nwskw	Patel	M	848444454	19	5465464654	jewnkJewnkJ

Printing All Data Entries

Patient Employee Room Type Room Details Department Doctor Inventory

Other Staff Treatments Diagnosis Cost Admission Pharmacy Medical List Redo Log

3) Trigger to Insert values into Medical List after Inserting in Medical Charges

create or replace trigger test1 after insert on medical_charge

for each row

declare

cursor count is select count(serial_no) from medicallist;

sid int;

begin

open count;

fetch count into sid;

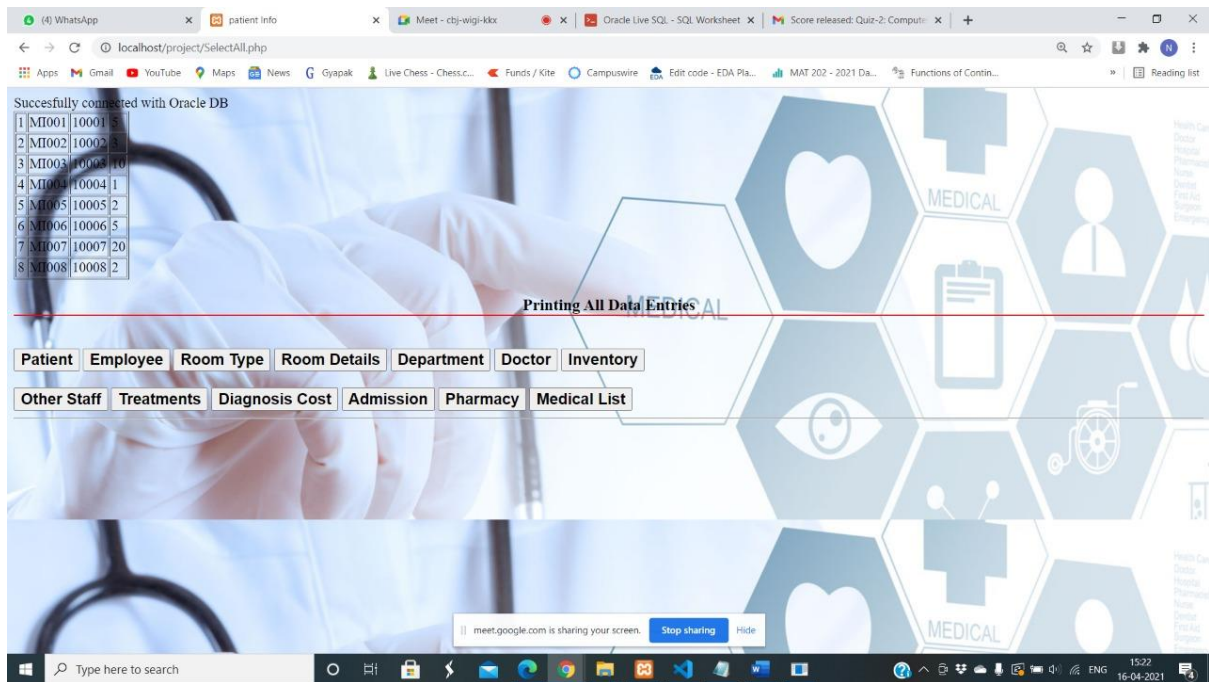
close count;

sid:=sid+1;

insert into medicallist values(sid,:new.medicine_id1,:new.pat_id,:new.quantity);

end;

/



4) Trigger for Intimation on Low Stock Prices

create or replace trigger stockalert after update or insert on inventory

for each row

begin

if (inserting) then

if (:new.stock < 3) then

raise_application_error(-20199, 'Stocks Less Than 3');

end if;

elsif (updating) then

if (:old.stock < 3) then

raise_application_error(-20012, 'Stocks Less Than 3');

else

dbms_output.put_line('Stocks are fine');

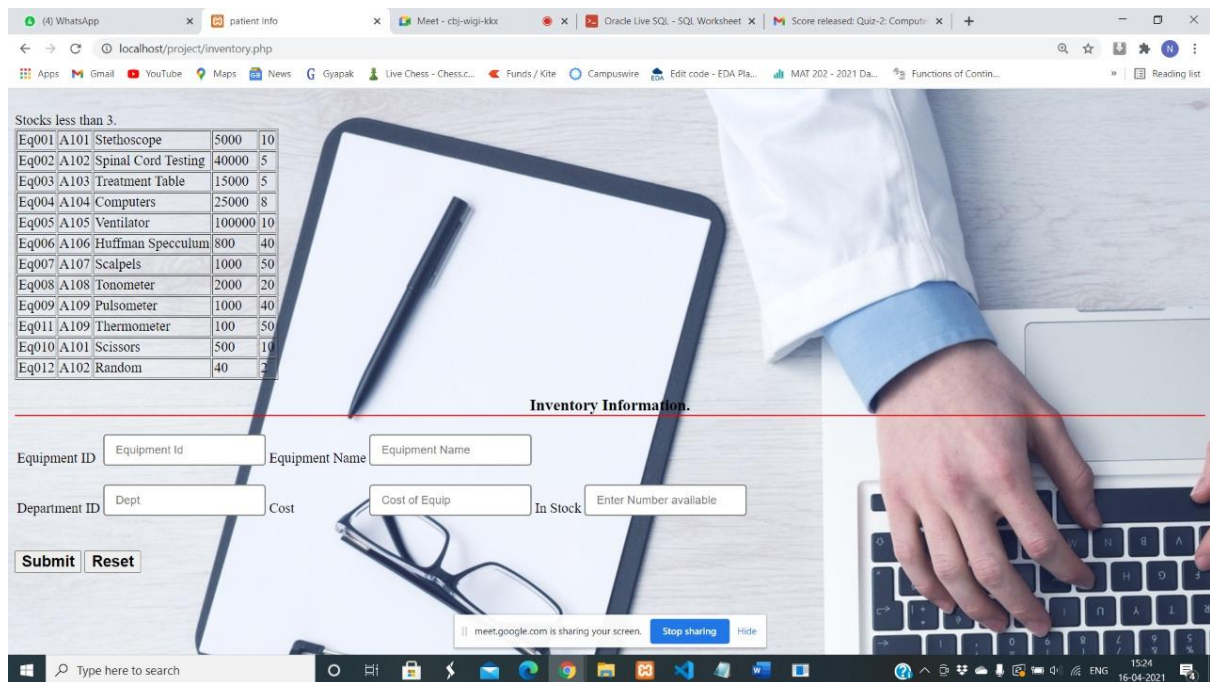
end if;

end if;

end;

/

Screenshot



5)Trigger for keeping a Log of the Audit and Before After Stocks on Inventory

create table redolog_values(equip_id varchar(10) , equip_name varchar(30), before_stock int,after_stock int);

create or replace trigger chk_redolog after update on inventory

for each row

begin

if(:new.stock<>:old.stock) then

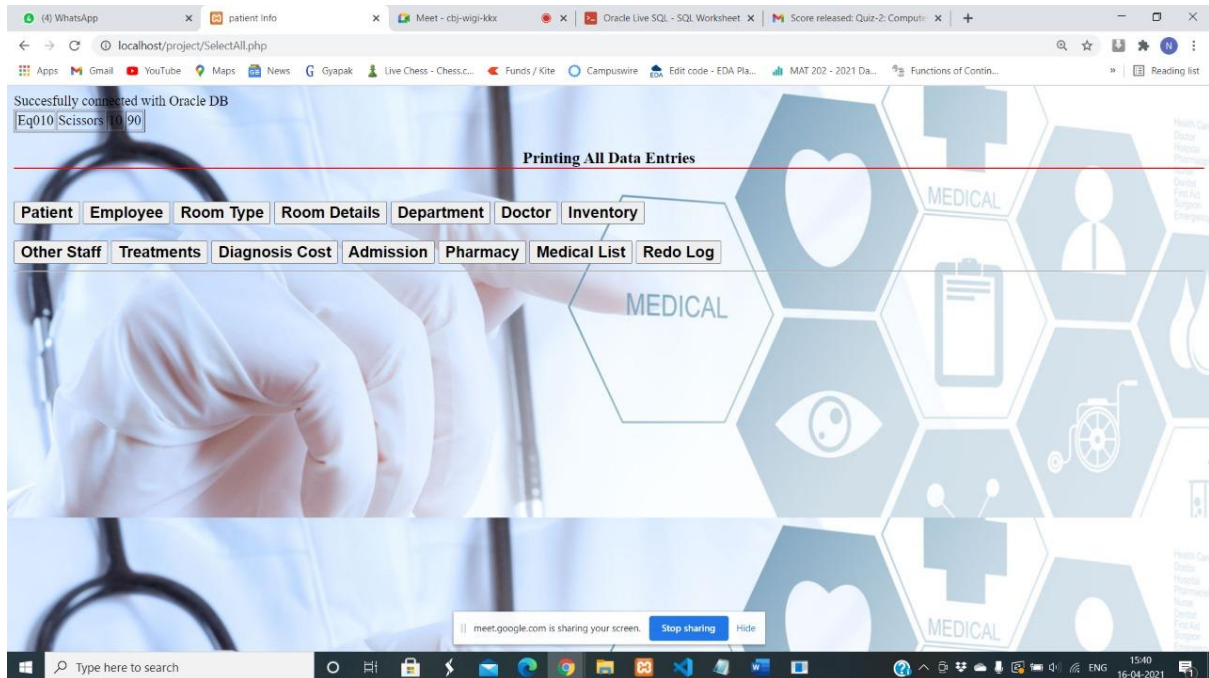
insert into redolog_values values(:old.equip_id,:old.equip_name,:old.stock,:new.stock);

end if;

end;

/

Output : Alert that Stocks less than 3



6)Trigger for Security Breach and Data Entry in Doctor Table

```
create table securtiy(user_name varchar2(20),current_date varchar(20), time varchar2(20));
```

```
create or replace trigger chk_trap after insert or update or delete on doctor
```

```
for each row
```

```
begin
```

```
    if(to_char(sysdate,'dy')='sat' or to_char(sysdate,'dy')='sun' or  
to_number(to_char(sysdate,'HH24'))<6 or to_number(to_char(sysdate,'HH24'))>22) then
```

```
        insert into security values(user,to_char(sysdate),SYSTIMESTAMP);
```

```
    end if;
```

```
end;
```

```
/
```


Is Applicable Only When Data is inserted or updates after 10 PM or Saturdays Sundays

7)Checking Valid Medicine Names

create or replace trigger chk_cost before insert or update on pharmacy

for each row

begin

if (:new.medicine_name is null) then

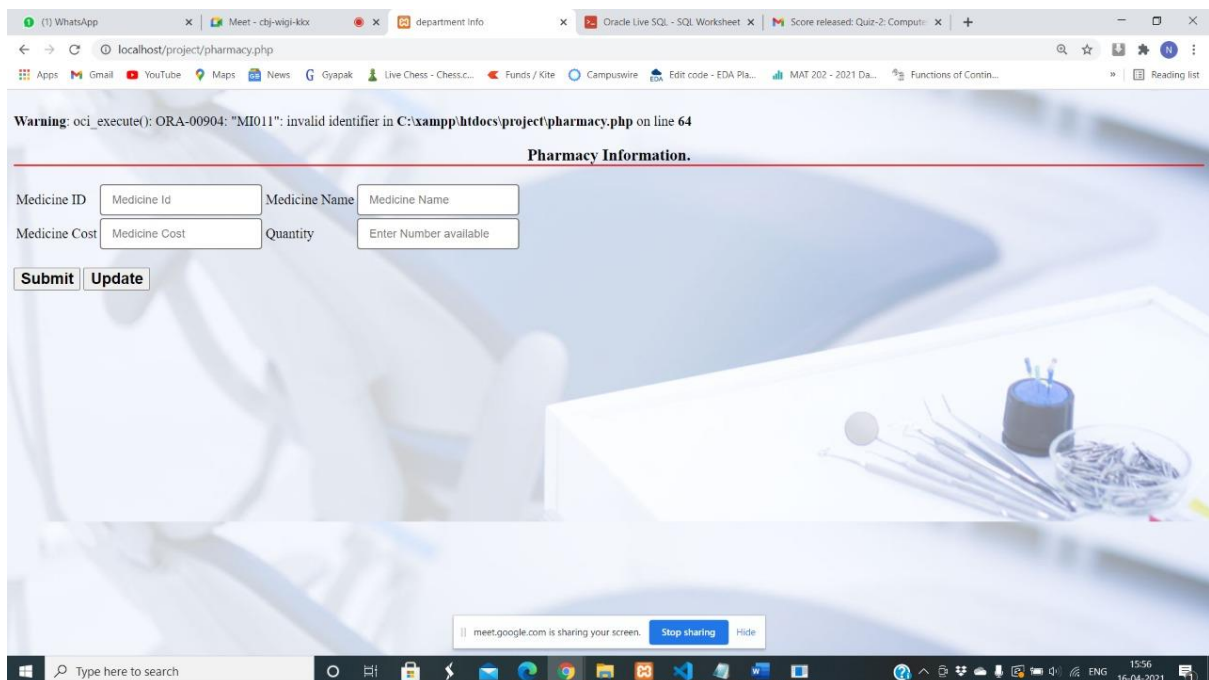
raise_application_error(-200016,'Enter a Proper Medicine Name ');

end if;

end;

/

Screenshot:



9) Trigger for Changing Occupancy Status of a Room when allocated to a Particular Patient

```
create or replace trigger changestatus after insert or update on admit
for each row
declare
    cursor c_status is select * from room_details;
    r_status c_status%rowtype;
begin
    open c_status ;
    loop
        fetch c_status into r_status;
        if c_status%NOTFOUND then
            exit;
        end if;
        if (r_status.room_no=:new.room_no) then
            update room_details set occupancy='Y' where room_no=:new.room_no;
        end if;
    end loop;
    close c_status;
end;
/
```

Screen Shot:

Successfully connected with Oracle DB

R003	Ward	Y	4	15000
R007	Ward	Y	4	15000
R008	Ward	N	4	15000
R00100	Ward	N	4	15000

Printing All Data Entries

Patient Employee Room Type Room Details Department Doctor Inventory

Other Staff Treatments Diagnosis Cost Admission Pharmacy Medical List

After inserting into admit the Occupancy Status is Changed to Y:

Successfully connected with Oracle DB

R003	Ward	Y	4	15000
R007	Ward	Y	4	15000
R008	Ward	N	4	15000
R00100	Ward	Y	4	15000

Printing All Data Entries

Patient Employee Room Type Room Details Department Doctor Inventory

Other Staff Treatments Diagnosis Cost Admission Pharmacy Medical List

10) To Set Salary minimum cap of 10000

create or replace trigger check_employee_salary

before insert or update on employee

for each row

begin

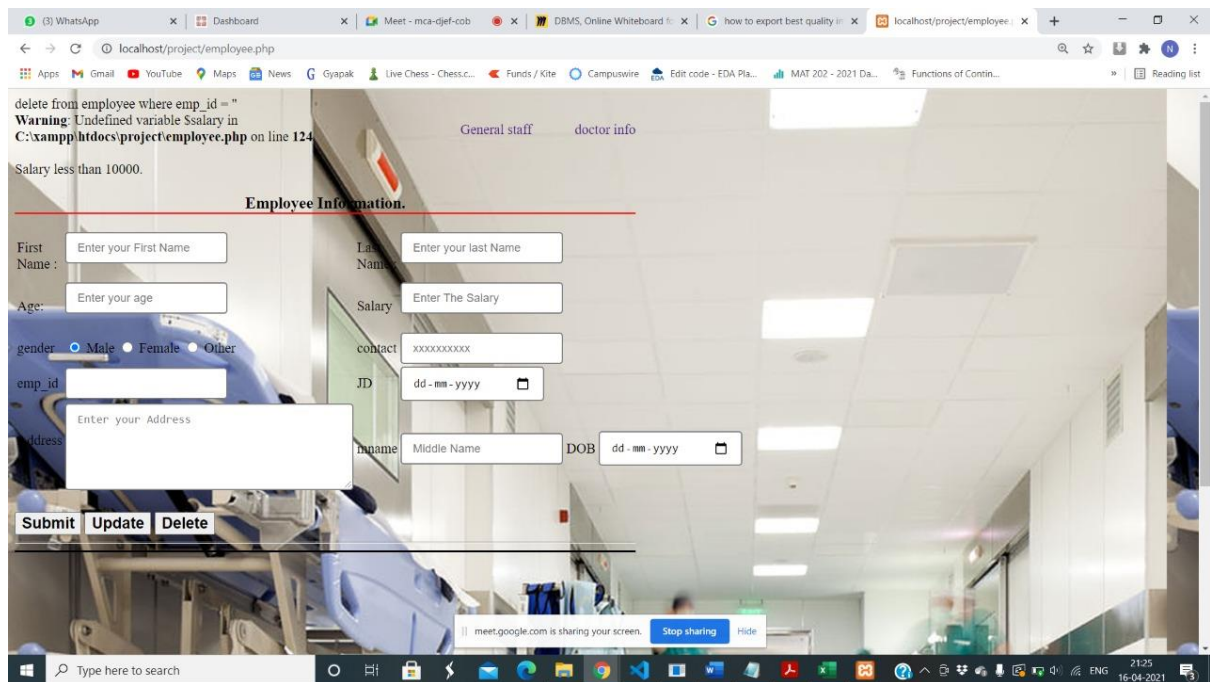
if (:NEW.salary < 10000) then

raise_application_error(-20005,'The entered value in salary is less than 10000!!');

end if;

end;

/



List of Stored Procedures:

1) Procedure to Input Room Type and Get List of Rooms in that Types and their Details

create or replace procedure roomdetails as

cursor c_type is select * from type;

```

r_type c_type%rowtype;

cursor c_room_type (typename type.room_typename%type) is select * from room_details
where room_type=typename;

r_room_type c_room_type%rowtype;

begin

open c_type;

loop

fetch c_type into r_type;

if c_type%notfound then

exit;

end if;

open c_room_type(r_type.room_typename);

loop

fetch c_room_type into r_room_type;

if c_room_type%notfound then

exit;

end if;

dbms_output.put_line(r_type.room_typename||' '||r_room_type.room_no ||'
'||r_room_type.occupancy||" "||r_room_type.occupancy_days);

end loop;

close c_room_type;

end loop;

close c_type;

end;

/

```

```
SQL> create or replace procedure roondetails as
2 cursor c_type is select * from type;
3 r_type c_type%rowtype;
4 cursor c_room_type (typename type.room_type%type) is select * from room_details where room_type=typename;
5 r_room_type c_room_type%rowtype;
6 begin
7 open c_type;
8 loop
9 fetch c_type into r_type;
10 if c_type%notfound then
11 exit;
12 end if;
13 open c_room_type(r_type.room_type);
14 loop
15 fetch c_room_type into r_room_type;
16 if c_room_type%notfound then
17 exit;
18 end if;
19 dbms_output.put_line(r_type.room_type||' '||r_room_type.room_no ||' '||r_room_type.occupancy||' '||r_room_type.occupancy_days);
20 end loop;
21 close c_room_type;
22 end loop;
23 close c_type;
24 end;
25 /

Procedure created.

SQL> execute roondetails;
Emer_Room R004 Y1
ICU R001 Y5
Surgery R005 Y1
Surgery R007 N2
Surgery R0011 N2
Theatre R002 Y2
Wait_Room R006 Y2
Ward R003 Y4

PL/SQL procedure successfully completed.

SQL>
```

2)For Giving List of Doctors and Patients Assigned to Them

create or replace procedure doctorassigned as

cursor c_doctor is select * from doctor;

r_doctor c_doctor%rowtype;

cursor c_docpat (doctorid doctor.emp_id%type) is select * from doctor_assigned where
dr_id=doctorid;

r_docpat c_docpat%rowtype;

begin

open c_doctor;

loop

fetch c_doctor into r_doctor;

if c_doctor%notfound then

exit;

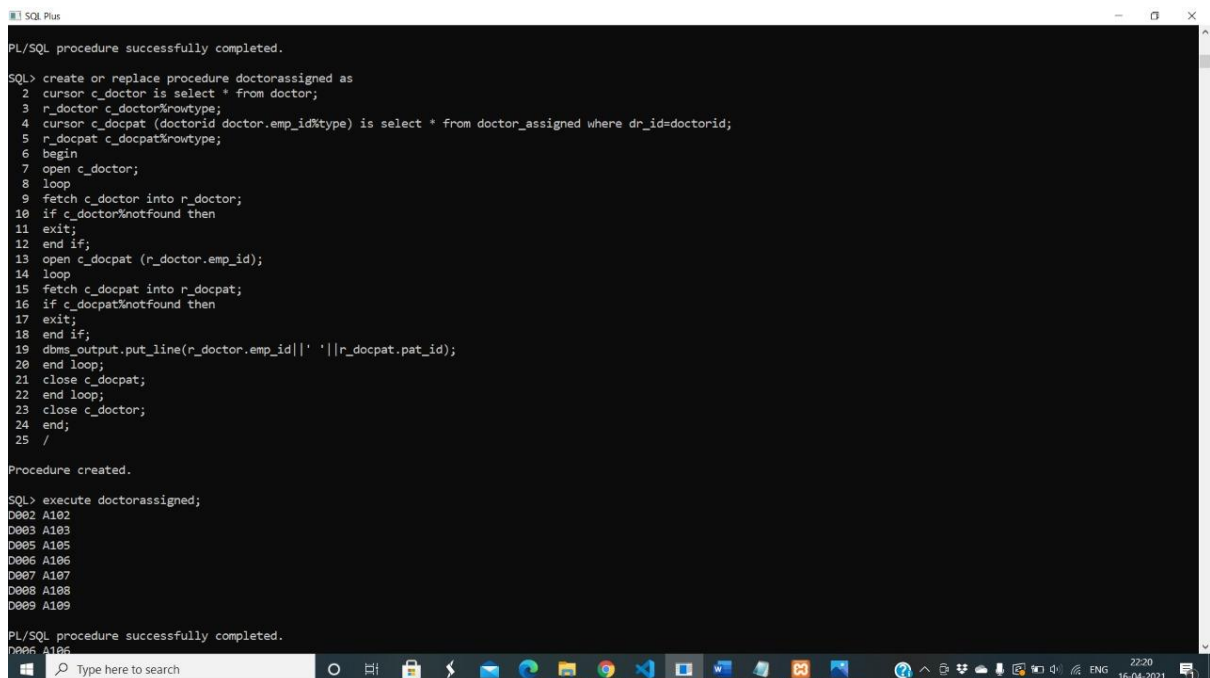
end if;

open c_docpat (r_doctor.emp_id);


```

loop
fetch c_docpat into r_docpat;
if c_docpat%notfound then
exit;
end if;
dbms_output.put_line(r_doctor.emp_id||' '||r_docpat.pat_id);
end loop;
close c_docpat;
end loop;
close c_doctor;
end;
/

```



```

SQL Plus
PL/SQL procedure successfully completed.

SQL> create or replace procedure doctorassigned as
2  cursor c_doctor is select * from doctor;
3  r_doctor c_doctor%rowtype;
4  cursor c_docpat (doctorid doctor.emp_idtype) is select * from doctor_assigned where dr_id=doctorid;
5  r_docpat c_docpat%rowtype;
6  begin
7  open c_doctor;
8  loop
9  fetch c_doctor into r_doctor;
10 if c_doctor%notfound then
11 exit;
12 end if;
13 open c_docpat (r_doctor.emp_id);
14 loop
15 fetch c_docpat into r_docpat;
16 if c_docpat%notfound then
17 exit;
18 end if;
19 dbms_output.put_line(r_doctor.emp_id||' '||r_docpat.pat_id);
20 end loop;
21 close c_docpat;
22 end loop;
23 close c_doctor;
24 end;
25 /

Procedure created.

SQL> execute doctorassigned;
D002 A102
D003 A103
D005 A105
D006 A106
D007 A107
D008 A108
D009 A109

PL/SQL procedure successfully completed.
D006 A106

```

3) For Calculating Final Bill

```
create or replace procedure finalbill(pat_id varchar, treatment_id varchar) as

cursor c_patient is select * from patient;

r_patient c_patient%rowtype;

cursor c_treatments is select * from treatments;

r_treatments c_treatments%rowtype;

cursor c_admit(patientid patient.pat_id%type) is select * from admit where
pat_id=patientid;

r_admit c_admit%rowtype;

cursor c_room_details(roomno admit.room_no%type) is select * from room_details;

r_room_details c_room_details%rowtype;

patientid varchar(20);

treatmentid varchar(20);

case_number varchar(20);

billing_date date;

room_no varchar(20);

room_charges int;

treatment_charge int;

tot int;

begin

for r_patient in c_patient loop

if (r_patient.pat_id=pat_id) then

patientid:=r_patient.pat_id;

end if;

end loop;

for r_treatments in c_treatments loop

if(r_treatments.treatment_id=treatment_id) then

treatmentid:=r_treatments.treatment_id;

treatment_charge:=r_treatments.charge;

end if;
```

```

end loop;
open c_admit (patientid);
loop
fetch c_admit into r_admit;
if c_admit%notfound then
exit;
end if;
case_number:=r_admit.case_number;
room_no:=r_admit.room_no;
end loop;
close c_admit;
open c_room_details (room_no);
loop
fetch c_room_details into r_room_details;
if c_room_details%notfound then
exit;
end if;
room_charges:=r_room_details.occupancy_days*r_room_details.charge_of_room;
end loop;
close c_room_details;
tot:=room_charges+treatment_charge;
dbms_output.put_line(tot);
insert into final_bill values()
end;
/

```

```
SQL Plus
53 dbms_output.put_line('Billing Date'||' '||billing_date);
54 dbms_output.put_line('Patient ID'||' '||patientid);
55 dbms_output.put_line('ROOM Charges'||' '||room_charges);
56 dbms_output.put_line('Treatment '||' '||treatmentid);
57 dbms_output.put_line('Treatment Charges'||' '||treatment_charge);
58 dbms_output.put_line('Amount Payable'||' '||tot);
59 end;
60 /

Procedure created.

SQL> execute finalbill('10001','Tr01');
** Final Bill**
Case Number CS05
ROOM Number R005
Billing Date
Patient ID 10001
ROOM Charges 40000
Treatment Tr01
Treatment Charges 5000
Amount Payable 45000

PL/SQL procedure successfully completed.

SQL> _
```

4) Procedure to Input Patient ID and retrieve data of Room in which he is admitted, Diagnosis Cost and Other Details

create or replace procedure patdetail(pat_id varchar) as

cursor c_patient is select * from patient;

r_patient c_patient%rowtype;

cursor c_admit (patid patient.pat_id%type) is select * from admit where pat_id=patid;

r_admit c_admit%rowtype;

cursor c_diagnosiscost (patid patient.pat_id%type) is select * from diagnosis_cost where pat_id=patid;

r_diagnosiscost c_diagnosiscost%rowtype;

begin

open c_patient;

loop

fetch c_patient into r_patient;

if c_patient%notfound then

exit;

```

end if;

open c_admit (r_patient.pat_id);

loop

fetch c_admit into r_admit;

if c_admit%notfound then

exit;

end if;

dbms_output.put_line(r_patient.pat_id||' ' ||r_patient.fname||' ' ||r_patient.lname||'
'|| r_admit.room_no||' ' ||r_admit.case_number);

end loop;

close c_admit;

end loop;

close c_patient;

end;

/

```

```

SQL Plus
PL/SQL: Statement ignored

SQL> create or replace procedure patdetail(pat_id varchar) as
2 cursor c_patient is select * from patient;
3 r_patient c_patient%rowtype;
4 cursor c_admit (patid patient.pat_id%type) is select * from admit where pat_id=patid;
5 r_admit c_admit%rowtype;
6 cursor c_diagnosiscost (patid patient.pat_id%type) is select * from diagnosis_cost where pat_id=patid;
7 r_diagnosiscost c_diagnosiscost%rowtype;
8 begin
9 open c_patient;
10 loop
11 fetch c_patient into r_patient;
12 if c_patient%notfound then
13 exit;
14 end if;
15 open c_admit (r_patient.pat_id);
16 loop
17 fetch c_admit into r_admit;
18 if c_admit%notfound then
19 exit;
20 end if;
21 dbms_output.put_line(r_patient.pat_id||' ' ||r_patient.fname||' ' ||r_patient.lname||' ' || r_admit.room_no||' ' ||r_admit.case_number);
22 end loop;
23 close c_admit;
24 end loop;
25 close c_patient;
26 end;
27 /

procedure created.

SQL> execute patdetail('10002');
10002 Rakesh Patel R001 CS01
10002 Rakesh Patel R002 CS02
10002 Rakesh Patel R003 CS03
10002 Rakesh Patel R004 CS04
10002 Rakesh Patel R005 CS05

PL/SQL procedure successfully completed.

SQL>

```

meet.google.com is sharing your screen. [Stop sharing](#) [Hide](#)

Type here to search

22:30 16-04-2021

5) For accessing List of Employees in a department

```
create or replace procedure doclist (dept_id varchar) as
cursor c_dept is select * from department ;
r_dept c_dept%rowtype;
cursor c_doc (deptid department.dept_id%type) is select * from doctor where
dept_id=deptid;
r_doc c_doc%rowtype;
dc int;
begin
open c_dept;
loop
dc:=0;
fetch c_dept into r_dept;
if c_dept%notfound then
exit;
end if;
open c_doc(r_dept.dept_id);
loop
fetch c_doc into r_doc;
if c_doc%notfound then
exit;
end if;
dc:=dc+1;
dbms_output.put_line(r_dept.dept_name||' '|| r_doc.emp_id);
end loop;
dbms_output.put_line('Number of Doctors in Department' ||' '||dc);
close c_doc;
end loop;
close c_dept;
end;
```


/

```
SQL Plus
5 r_doc c_doc%rowtype;
6 dc int;
7 begin
8 open c_dept;
9 loop
10 dc:=0;
11 fetch c_dept into r_dept;
12 if c_dept%notfound then
13 exit;
14 end if;
15 open c_doc(r_dept.dept_id);
16 loop
17 fetch c_doc into r_doc;
18 if c_doc%notfound then
19 exit;
20 end if;
21 dc:=dc+1;
22 dbms_output.put_line(r_dept.dept_name||' '|| r_doc.emp_id);
23 end loop;
24 dbms_output.put_line('Number of Doctors in Department'||' '||dc);
25 close c_doc;
26 end loop;
27 close c_dept;
28 end;
29 /

Procedure created.

SQL> execute doclist('D003');
Number of Doctors in Department 0
Neuro depart D002
Number of Doctors in Department 1
Physio depart D003
Number of Doctors in Department 1
Medical depart D004
Number of Doctors in Department 1
Ich depart D005
Number of Doctors in Department 1
Gynec depart D006
Number of Doctors in Department 1
Dental depart D007
Number of Doctors in Department 1
Optical depart D008
Number of Doctors in Department 1
Cardiac depart D009
Number of Doctors in Department 1

PL/SQL procedure successfully completed.

SQL>
```

6)Input Department and get list of available Treatments

create or replace procedure availabletreatments(dept_id varchar) as

cursor c_dept is select * from department ;

r_dept c_dept%rowtype;

cursor c_treatment(deptid department.dept_id%type) is select * from treatments where
dept_id=deptid;

r_treatment c_treatment%rowtype;

begin

open c_dept;

loop

fetch c_dept into r_dept;

```

if c_dept%notfound then

exit;

end if;

open c_treatment(r_dept.dept_id);

loop

fetch c_treatment into r_treatment;

if c_treatment%notfound then

exit;

end if;

dbms_output.put_line(r_dept.dept_id||' '||r_dept.dept_name||'
'||r_treatment.treatment_name);

end loop;

close c_treatment;

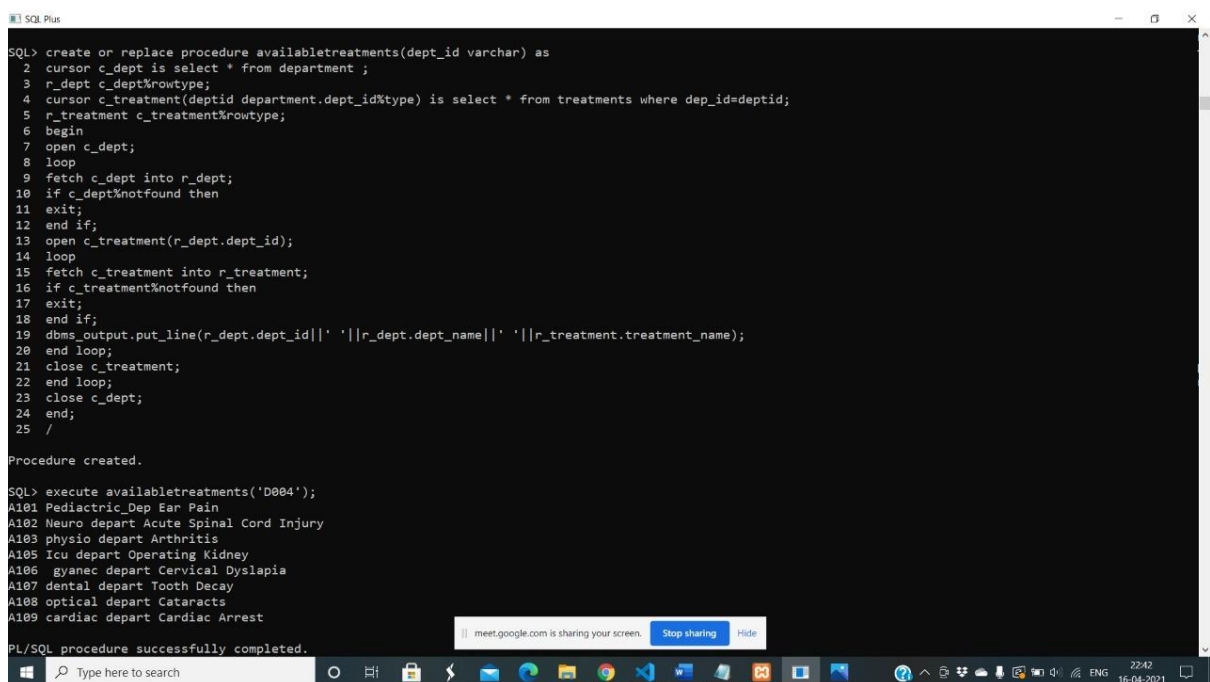
end loop;

close c_dept;

end;

/

```



The screenshot shows a SQL Plus window with a dark background. The window title is "SQL Plus". The main area displays the following text:

```

SQL> create or replace procedure availabletreatments(dept_id varchar) as
2 cursor c_dept is select * from department ;
3 r_dept c_dept%rowtype;
4 cursor c_treatment(deptid department.dept_id%type) is select * from treatments where dep_id=deptid;
5 r_treatment c_treatment%rowtype;
6 begin
7 open c_dept;
8 loop
9 fetch c_dept into r_dept;
10 if c_dept%notfound then
11 exit;
12 end if;
13 open c_treatment(r_dept.dept_id);
14 loop
15 fetch c_treatment into r_treatment;
16 if c_treatment%notfound then
17 exit;
18 end if;
19 dbms_output.put_line(r_dept.dept_id||' '||r_dept.dept_name||' '||r_treatment.treatment_name);
20 end loop;
21 close c_treatment;
22 end loop;
23 close c_dept;
24 end;
25 /

Procedure created.

SQL> execute availabletreatments('D004');
A101 Pediatric Dep Ear Pain
A102 Neuro depart Acute Spinal Cord Injury
A103 physio depart Arthritis
A105 Icu depart Operating Kidney
A106 gyanec depart Cervical Dyslapia
A107 dental depart Tooth Decay
A108 optical depart Cataracts
A109 cardiac depart Cardiac Arrest

PL/SQL procedure successfully completed.

```

At the bottom of the window, there is a status bar with the text "meet.google.com is sharing your screen." and buttons for "Stop sharing" and "Hide". The Windows taskbar is visible at the very bottom, showing the search bar and various application icons.

7)Input Department and Get all necessary Inventory details Department wise

create or replace procedure Department_equip(dept_name varchar) as

cursor C_depart is select * from department;

cursor C_equip is select * from inventory;

r_depart c_depart%rowtype;

r_equip c_equip%rowtype;

begin

for r_depart in c_depart loop

if(r_depart.dept_name = dept_name)then

dbms_output.put_line('Department name =' || dept_name);

dbms_output.put_line('Equipment'||' '||'Equipment id'||' '||'cost'||'
'||'Existing stock');

for r_equip in C_equip loop

if(r_equip.dept_id = r_depart.dept_id) then

dbms_output.put_line(r_equip.equip_name||' '||r_equip.equip_id||'
'||r_equip.cost||' '||r_equip.stock);

end if;

end loop;

end if;

end loop;

end;

/

```

SQL> create or replace procedure Department_equip(dept_name varchar) as
1  cursor C_depart is select * from department; cursor C_equip is select * from inventory;
2  in c_depart loop
3      if( r_depart.dept_name = dept_name)then
4          dbms_output.put_line('Department name = ' || dept_name);
5          for r_equip in C_equip loop
6              if(r_equip.dept_id = r_depart.dept_id) then
7                  dbms_output.put_line('Equipment id' || ' ' || 'cost' || ' ' || 'Existing stock' );
8              end if;
9          end loop;
10     end if;
11 end loop;
12 /
SQL>
SQL> create or replace procedure Department_equip(dept_name varchar) as
1  cursor C_depart is select * from department;
2  cursor C_equip is select * from inventory;
3  r_depart c_depart%rowtype;
4  r_equip c_equip%rowtype;
5  begin
6      for r_depart in c_depart loop
7          if( r_depart.dept_name = dept_name)then
8              dbms_output.put_line('Department name = ' || dept_name);
9              dbms_output.put_line('Equipment id' || ' ' || 'cost' || ' ' || 'Existing stock' );
10             for r_equip in C_equip loop
11                 if(r_equip.dept_id = r_depart.dept_id) then
12                     dbms_output.put_line(r_equip.equip_name || ' ' || r_equip.equip_id || ' ' || r_equip.cost || ' ' || r_equip.stock);
13                 end if;
14             end loop;
15         end if;
16     end loop;
17 end;
18 /
SQL>
Procedure created.

SQL> execute Department_equip('Neuro depart');
Department name =Neuro depart
Equipment id      cost Existing stock
Spinal Cord Testing  Eq002  40000  5
Random            Eq012  40  90

PL/SQL procedure successfully completed.

SQL>

```

8)Function to print the number of Rooms Available

create or replace function room_available return int as

cursor c_room is select * from room_details;

r_room c_room%rowtype;

d int := 0;

begin

for r_room in c_room loop

if(r_room.occupancy = 'N') then

d := d + 1;

end if;

end loop;

return d;

end;

/

```
SQL Plus
SQL> create or replace function room_available return int as
2   cursor c_room is select * from room_details;
3   r_room c_room%rowtype;
4   d int := 0;
5   begin
6       for r_room in c_room loop
7           if(r_room.occupancy = 'N') then
8               d := d + 1;
9           end if;
10        end loop;
11        return d;
12    end;
13 /

Function created.

SQL> select room_available from dual;

ROOM_AVAILABLE
-----
2

SQL>
```