

**American International University- Bangladesh**

**CSC 2212: ADVANCE DATABASE MANAGEMENT SYSTEMS**

**Midterm Project Report Fall 20-21**

**Project Title: The Healing Infirmary**

**Section: B**

**Group: Squad**

|  |  |
| --- | --- |
| Student Name | Student ID |
| Sultana, Sharmin | 18-37151-1 |
| Aklima Binte Alam | 18-38638-2 |
| Ekhtear Ahmed Fahim | 15-30492-3 |

=============================================

CREATING TABLES NECESSARY FOR DATABASE

=============================================

CREATE TABLE U\_DETAIL (

U\_ID NUMBER NOT NULL CONSTRAINT U\_ID\_PK PRIMARY KEY,

U\_PASS VARCHAR2(20) NOT NULL,

U\_NAME VARCHAR2(30) NOT NULL,

U\_NUMBER VARCHAR2(11) NOT NULL,

U\_MAIL VARCHAR2(20) NOT NULL,

U\_ROLE VARCHAR2(20) NOT NULL,

U\_INFO VARCHAR2(200),

U\_SHIFT VARCHAR2(100)

);

============================================

CREATE TABLE U\_DETAIL\_REC (

UR\_ID NUMBER NOT NULL CONSTRAINT UR\_ID\_PK PRIMARY KEY,

O\_TYPE VARCHAR2(10 BYTE) NOT NULL,

INSERT\_TIME DATE NOT NULL,

U\_ID NUMBER NOT NULL,

U\_NAME VARCHAR2(30 BYTE) NOT NULL,

U\_NUMBER VARCHAR2(11 BYTE) NOT NULL,

U\_MAIL VARCHAR2(20 BYTE) NOT NULL,

U\_ROLE VARCHAR2(20 BYTE) NOT NULL,

U\_INFO VARCHAR2(200 BYTE),

U\_SHIFT VARCHAR2(100 BYTE)

) ;

===========================================

CREATE TABLE PRESCRIPTION

(

P\_ID NUMBER NOT NULL,

U\_ID NUMBER NOT NULL,

P\_INFO VARCHAR2(20) NOT NULL,

CONSTRAINT P\_ID\_PK PRIMARY KEY

(P\_ID) ENABLE );

ALTER TABLE PRESCRIPTION ADD CONSTRAINT U\_ID\_FK1 FOREIGN KEY( U\_ID ) REFERENCES U\_DETAIL( U\_ID )

ENABLE;

==========================================

CREATE TABLE PRESCRIPTION\_REC

(

PR\_ID NUMBER NOT NULL,

O\_TYPE VARCHAR2(10 BYTE) NOT NULL,

INSERT\_TIME DATE NOT NULL,

P\_ID NUMBER NOT NULL,

U\_ID NUMBER NOT NULL,

P\_INFO VARCHAR2(20) NOT NULL,

CONSTRAINT PR\_ID\_PK PRIMARY KEY

(PR\_ID) ENABLE );

==========================================

CREATE TABLE MEDICINE

(

M\_ID NUMBER NOT NULL,

M\_NAME VARCHAR2(20) NOT NULL,

M\_PRICE VARCHAR2(10) NOT NULL,

M\_STATUS VARCHAR2(20) NOT NULL,

CONSTRAINT M\_ID\_PK PRIMARY KEY (M\_ID) ENABLE

);

=========================================

CREATE TABLE MEDICINE\_REC

(

MR\_ID NUMBER NOT NULL,

O\_TYPE VARCHAR(10) NOT NULL,

INSERT\_TIME DATE NOT NULL,

M\_ID NUMBER NOT NULL,

M\_NAME VARCHAR2(20) NOT NULL,

M\_PRICE VARCHAR2(10) NOT NULL,

M\_STATUS VARCHAR2(20) NOT NULL,

CONSTRAINT MR\_ID\_PK PRIMARY KEY (MR\_ID) ENABLE

);

=========================================

CREATE TABLE AMBULANCE(

A\_ID NUMBER NOT NULL,

CONTACT VARCHAR2(11) NOT NULL,

STATUS VARCHAR2(20) NOT NULL,

CONSTRAINT A\_ID\_PK PRIMARY KEY (A\_ID) ENABLE);

=========================================

CREATE TABLE CONTACT(

C\_NAME VARCHAR2(30) NOT NULL,

C\_NUMBER VARCHAR2(11) NOT NULL,

C\_MAIL VARCHAR2(30) NOT NULL,

C\_MASSAGE VARCHAR2(200) NOT NULL);

======================================

NECESSARY SEQUENCES for PRIMARY KEY

======================================

------------------------------------------------------------------------------

CREATE SEQUENCE A\_ID1 START WITH 1

INCREMENT BY 1 NOCACHE NOCYCLE;

CREATE SEQUENCE U\_ID1 START WITH 1

INCREMENT BY 1 NOCACHE NOCYCLE;

CREATE SEQUENCE UR\_ID1 START WITH 1

INCREMENT BY 1 NOCACHE NOCYCLE;

CREATE SEQUENCE P\_ID1 START WITH 1

INCREMENT BY 1 NOCACHE NOCYCLE;

CREATE SEQUENCE PR\_ID1 START WITH 1

INCREMENT BY 1 NOCACHE NOCYCLE;

CREATE SEQUENCE M\_ID1 START WITH 1

INCREMENT BY 1 NOCACHE NOCYCLE;

CREATE SEQUENCE MR\_ID1 START WITH 1

INCREMENT BY 1 NOCACHE NOCYCLE;

=============================================

CREATING TRIGGERS

=============================================

=============================================

111111111111111111111111111111111111111111111111

=============================================

CREATE TRIGGER U\_DETAIL\_T

AFTER INSERT OR UPDATE ON U\_DETAIL FOR EACH ROW

DECLARE

TYPE\_1 VARCHAR2(10);

BEGIN

TYPE\_1:= CASE

WHEN INSERTING THEN 'INSERT'

WHEN UPDATING THEN 'UPDATE'

END;

INSERT INTO U\_DETAIL\_REC(UR\_ID,O\_TYPE,INSERT\_TIME,U\_ID,U\_NAME,U\_NUMBER,U\_MAIL,U\_ROLE,U\_INFO,U\_SHIFT)

VALUES (UR\_ID1.NEXTVAL,TYPE\_1,sysdate,:NEW.U\_ID,:NEW.U\_NAME,:NEW.U\_NUMBER,:NEW.U\_MAIL,:NEW.U\_ROLE,:NEW.U\_INFO,:NEW.U\_SHIFT);

END;

============================================

22222222222222222222222222222222222222222222222

============================================

CREATE TRIGGER PRESCRIPTION\_T

AFTER INSERT OR UPDATE ON PRESCRIPTION FOR EACH ROW

DECLARE

TYPE\_1 VARCHAR2(10);

BEGIN

TYPE\_1:= CASE

WHEN INSERTING THEN 'INSERT'

WHEN UPDATING THEN 'UPDATE'

END;

INSERT INTO PRESCRIPTION\_REC(PR\_ID,O\_TYPE,INSERT\_TIME,P\_ID, U\_ID, P\_INFO)

VALUES (PR\_ID1.NEXTVAL,TYPE\_1,sysdate,:NEW.P\_ID, :NEW.U\_ID, :NEW.P\_INFO);

END;

============================================

33333333333333333333333333333333333333333333333

============================================

CREATE TRIGGER PRE\_DEL

AFTER DELETE OR UPDATE ON U\_DETAIL FOR EACH ROW

BEGIN

IF :OLD.U\_ID =1 THEN

raise\_application\_error(-20015, 'You can't delete or update the admin row');

END IF;

END;

===========================================

4444444444444444444444444444444444444444444444

===========================================

CREATE OR REPLACE TRIGGER U\_ROLE\_CONTROL

AFTER INSERT OR UPDATE ON U\_DETAIL FOR EACH ROW

BEGIN

IF :NEW.U\_ROLE NOT IN('admin','doctor','accountant','nurse','patient') THEN

raise\_application\_error(-20015, 'Please choose a valid role as admin or doctor or accountant or nurse or patient');

END IF;

END;

============================================

INSERTING DATA INTO TABLES

=============================================

=============================================

111111111111111111111111111111111111111111111111

=============================================

INSERT INTO U\_DETAIL (U\_ID, U\_PASS, U\_NAME, U\_NUMBER, U\_MAIL, U\_ROLE, U\_INFO, U\_SHIFT) VALUES (U\_ID1.NEXTVAL, '123', 'sharmin', '01234567890', 'sharmin@gmail.com', 'admin', 'hr', 'su-fr-9am-5pm');

INSERT INTO U\_DETAIL (U\_ID, U\_PASS, U\_NAME, U\_NUMBER, U\_MAIL, U\_ROLE, U\_INFO, U\_SHIFT) VALUES (U\_ID1.NEXTVAL, '456', 'fahim', '09876543210', 'ekhtearf@gmail.com', 'doctor', 'sergon', 'mo-we-12pm-5pm');

INSERT INTO U\_DETAIL (U\_ID, U\_PASS, U\_NAME, U\_NUMBER, U\_MAIL, U\_ROLE, U\_INFO, U\_SHIFT) VALUES (U\_ID1.NEXTVAL, '789', 'aklima', '09878654375', 'aklima@gmail.com', 'doctor', 'medicine', 'th-sa-9am-2pm');

INSERT INTO U\_DETAIL (U\_ID, U\_PASS, U\_NAME, U\_NUMBER, U\_MAIL, U\_ROLE, U\_SHIFT) VALUES (U\_ID1.NEXTVAL, '234', 'rohim', '74646578694', 'rohim@gmail.com', 'accountant', 'su-fr-9am-5pm');

INSERT INTO U\_DETAIL (U\_ID, U\_PASS, U\_NAME, U\_NUMBER, U\_MAIL, U\_ROLE, U\_SHIFT) VALUES (U\_ID1.NEXTVAL, '345', 'korima', '73658746578', 'korima@gmail.com', 'nurse', 'su-fr-9am-5pm');

INSERT INTO U\_DETAIL (U\_ID, U\_PASS, U\_NAME, U\_NUMBER, U\_MAIL, U\_ROLE, U\_INFO) VALUES (U\_ID1.NEXTVAL, '321', 'rofiq', '67584756456', 'rofiq@gmail.com', 'patient', 'emergency');

INSERT INTO U\_DETAIL (U\_ID, U\_PASS, U\_NAME, U\_NUMBER, U\_MAIL, U\_ROLE, U\_INFO) VALUES (U\_ID1.NEXTVAL, '654', 'shofiq', '65375867465', 'shofiq@gmail.com', 'patient', 'mild');

INSERT INTO U\_DETAIL (U\_ID, U\_PASS, U\_NAME, U\_NUMBER, U\_MAIL, U\_ROLE, U\_INFO) VALUES (U\_ID1.NEXTVAL, '987', 'raihan', '98765374658', 'raihan@gmail.com', 'patient', 'moderate');

INSERT INTO U\_DETAIL (U\_ID, U\_PASS, U\_NAME, U\_NUMBER, U\_MAIL, U\_ROLE, U\_INFO) VALUES (U\_ID1.NEXTVAL, '876', 'hamid', '36485736475', 'hamid@gmail.com', 'patient', 'emergency');

INSERT INTO U\_DETAIL (U\_ID, U\_PASS, U\_NAME, U\_NUMBER, U\_MAIL, U\_ROLE, U\_INFO) VALUES (U\_ID1.NEXTVAL, '567', 'kobir', '45896325418', 'kobir@gmail.com', 'patient', 'emergency');

============================================

22222222222222222222222222222222222222222222222

============================================

INSERT INTO AMBULANCE(A\_ID,CONTACT, STATUS) VALUES (A\_ID1.NEXTVAL,'47685968475', 'active');

INSERT INTO AMBULANCE (A\_ID,CONTACT, STATUS) VALUES (A\_ID1.NEXTVAL,'76548576894', 'inactive');

INSERT INTO AMBULANCE (A\_ID,CONTACT, STATUS) VALUES (A\_ID1.NEXTVAL,'37495847364', 'active');

INSERT INTO AMBULANCE (A\_ID,CONTACT, STATUS) VALUES (A\_ID1.NEXTVAL,'38457596049', 'active');

INSERT INTO AMBULANCE (A\_ID,CONTACT, STATUS) VALUES (A\_ID1.NEXTVAL,'58695848573', 'inactive');

==========================================

333333333333333333333333333333333333333333333

==========================================

INSERT INTO PRESCRIPTION (P\_ID, U\_ID, P\_INFO) VALUES (P\_ID1.NEXTVAL, '2', 'rofiq-needs-meds');

INSERT INTO PRESCRIPTION (P\_ID, U\_ID, P\_INFO) VALUES (P\_ID1.NEXTVAL, '2', 'hamid-needs-rest');

INSERT INTO PRESCRIPTION (P\_ID, U\_ID, P\_INFO) VALUES (P\_ID1.NEXTVAL, '3', 'raihan-needs-meds');

INSERT INTO PRESCRIPTION (P\_ID, U\_ID, P\_INFO) VALUES (P\_ID1.NEXTVAL, '3', 'kobir-needs-rest');

INSERT INTO PRESCRIPTION (P\_ID, U\_ID, P\_INFO) VALUES (P\_ID1.NEXTVAL, '2', 'shofiq-needs-rest');

INSERT INTO PRESCRIPTION (P\_ID, U\_ID, P\_INFO) VALUES (P\_ID1.NEXTVAL, '2', 'rofiq-needs-rest');

INSERT INTO PRESCRIPTION (P\_ID, U\_ID, P\_INFO) VALUES (P\_ID1.NEXTVAL, '2', 'hamid-needs-meds');

INSERT INTO PRESCRIPTION (P\_ID, U\_ID, P\_INFO) VALUES (P\_ID1.NEXTVAL, '3', 'raihan-needs-rest');

INSERT INTO PRESCRIPTION (P\_ID, U\_ID, P\_INFO) VALUES (P\_ID1.NEXTVAL, '3', 'kobir needs-meds');

INSERT INTO PRESCRIPTION (P\_ID, U\_ID, P\_INFO) VALUES (P\_ID1.NEXTVAL, '3', 'shofiq-needs-meds');

============================================

44444444444444444444444444444444444444444444444

============================================

INSERT INTO MEDICINE (M\_ID, M\_NAME, M\_PRICE, M\_STATUS) VALUES (M\_ID1.NEXTVAL, 'Napa Extra', '2', 'available');

INSERT INTO MEDICINE (M\_ID, M\_NAME, M\_PRICE, M\_STATUS) VALUES(M\_ID1.NEXTVAL, 'Napa Extend', '2', 'available');

INSERT INTO MEDICINE (M\_ID, M\_NAME, M\_PRICE, M\_STATUS) VALUES (M\_ID1.NEXTVAL, 'Seclo', '5', 'available');

INSERT INTO MEDICINE (M\_ID, M\_NAME, M\_PRICE, M\_STATUS) VALUES (M\_ID1.NEXTVAL, 'Phizer', '20000', 'available');

INSERT INTO MEDICINE(M\_ID, M\_NAME, M\_PRICE, M\_STATUS) VALUES (M\_ID1.NEXTVAL, 'Moderna', '19000', 'available');

INSERT INTO MEDICINE (M\_ID, M\_NAME, M\_PRICE, M\_STATUS) VALUES (M\_ID1.NEXTVAL, 'Astrazeneca', '22000', 'available');

INSERT INTO MEDICINE (M\_ID, M\_NAME, M\_PRICE, M\_STATUS) VALUES (M\_ID1.NEXTVAL, 'Vicadine', '200', 'unavailable');

INSERT INTO MEDICINE (M\_ID, M\_NAME, M\_PRICE, M\_STATUS) VALUES (M\_ID1.NEXTVAL, 'Xennex', '200', 'unavailable');

INSERT INTO MEDICINE (M\_ID, M\_NAME, M\_PRICE, M\_STATUS) VALUES (M\_ID1.NEXTVAL, 'Tushka', '70', 'unavailable');

INSERT INTO MEDICINE (M\_ID, M\_NAME, M\_PRICE, M\_STATUS) VALUES (M\_ID1.NEXTVAL, 'Paracitamol', '2', 'unavailable');

INSERT INTO MEDICINE (M\_ID, M\_NAME, M\_PRICE, M\_STATUS) VALUES (M\_ID1.NEXTVAL, 'Histacine', '2', 'unavailable');

INSERT INTO MEDICINE (M\_ID, M\_NAME, M\_PRICE, M\_STATUS) VALUES (M\_ID1.NEXTVAL, 'Exium', '24', 'unavailable');

===========================================

5555555555555555555555555555555555555555555555

===========================================

INSERT INTO CONTACT (C\_NAME, C\_NUMBER, C\_MAIL, C\_MASSAGE) VALUES ('raihan', '98765374658', 'raihan@gmail.com', 'I need to see a doctor');

INSERT INTO CONTACT (C\_NAME, C\_NUMBER, C\_MAIL, C\_MASSAGE) VALUES ('rofiq', '67584756456', 'rofiq@gmail.com', 'I am very sick, can someone send an ambulance?');

INSERT INTO CONTACT (C\_NAME, C\_NUMBER, C\_MAIL, C\_MASSAGE) VALUES ('shofiq', '65375867465', 'shofiq@gmail.com', 'Is there any doctors who specializes in surgery?')

-----------------------------------------------------------------------------

============================================

CREATING VIEWS

============================================

Read First: First Run the CRATE VIEW Queries on the workspace....then Run the select \* queries that are located above each view.

...................

1............................................................................

see the list of all the active and inactive ambulances

...................

select \* from AMB\_ACT

....................

CREATE OR REPLACE VIEW AMB\_ACT AS SELECT STATUS,LISTAGG(CONTACT,', ') WITHIN GROUP (ORDER BY CONTACT) AS "PHONE NUMBERS" FROM AMBULANCE GROUP BY STATUS;

...................

2................................................................

See the details of all the doctors and nurses

....................

Select \* from doc\_nur\_det

....................................

CREATE VIEW DOC\_NUR\_DET

AS SELECT u\_detail.u\_name,u\_detail.u\_role,u\_detail.u\_info,u\_detail.u\_number,u\_detail.u\_shift,u\_detail.u\_mail

FROM

u\_detail

WHERE

u\_detail.u\_role in ('Doctor','nurse')

GROUP BY

u\_detail.u\_name,u\_detail.u\_role,u\_detail.u\_info,u\_detail.u\_number,u\_detail.u\_shift,u\_detail.u\_mail;

...................

3......................................................................................

See the details of all the prescriptions and doctors that prescribed them

.......................

select \* from doc\_pr\_det

...................

CREATE VIEW DOC\_PR\_DET

AS SELECT

PRESCRIPTION.P\_ID,

U\_DETAIL.U\_NAME,

U\_DETAIL.U\_INFO,

PRESCRIPTION.P\_INFO,

U\_DETAIL.U\_SHIFT

FROM

U\_DETAIL,PRESCRIPTION

WHERE

U\_DETAIL.U\_ID=PRESCRIPTION.U\_ID;

.....................................................................................

4......................................................................................

See all the details of people who contacted you if their number exists on U\_DETAIL table

.......................

select \* from con\_det

...................

CREATE VIEW CON\_DET

AS SELECT \* FROM

U\_DETAIL,CONTACT

WHERE

U\_DETAIL.U\_NUMBER=CONTACT.C\_NUMBER ;

============================================

ALL PROCEDURES

=============================================

.......................................................................

111111111111111111111111111111111111

.........................................................................

create or replace procedure staff\_up(user\_name u\_detail.u\_name%type)

is

staff\_shift u\_detail.u\_name%type;

begin

select u\_shift into staff\_shift from u\_detail where u\_name=user\_name;

dbms\_output.put\_line('The staff works in these days '||staff\_shift);

end staff\_up;

-------------------------------------------------------------

set serveroutput on;

exec staff\_up('fahim');

..................................................................

222222222222222222222222222222222

..................................................................

create or replace PROCEDURE secure\_dml

IS

BEGIN

IF TO\_CHAR (SYSDATE, 'HH24:MI') NOT BETWEEN '08:00' AND '18:00'

OR TO\_CHAR (SYSDATE, 'DY') IN ('SAT', 'SUN') THEN

RAISE\_APPLICATION\_ERROR (-20205,

'You may only make changes during normal office hours');

END IF;

END secure\_dml;

---------------------------------------------------

exec secure\_dml;

................................................................

33333333333333333333333333333333333

................................................................

create or replace PROCEDURE print\_contact(in\_u\_id NUMBER)

IS

udet u\_detail%ROWTYPE;

BEGIN

SELECT \* INTO udet FROM u\_detail WHERE u\_id = in\_u\_id;

dbms\_output.put\_line(udet.u\_name || ' ' || udet.u\_number || '<' || udet.u\_mail ||'>' );

EXCEPTION

WHEN OTHERS THEN dbms\_output.put\_line( SQLERRM );

END;

-------------------------------------------------------

exec print\_contact(2);

....................................................................

444444444444444444444444444444444444

................................................................

CREATE OR REPLACE PROCEDURE

getLastPatientName

AS patient\_name u\_detail%rowtype;

BEGIN

SELECT \* INTO patient\_name from u\_detail where

u\_id=(select max(u\_id) from u\_detail where u\_role='patient');

dbms\_output.put\_line(patient\_name.u\_name);

end;

-------------------------------------------------------

EXECUTE getLastPatientName;

............................................................................

5555555555555555555555555555555555555555

.............................................................................

CREATE OR REPLACE PROCEDURE TODAY\_IS

AS

BEGIN

DBMS\_OUTPUT.PUT\_LINE('Date of today is ' || TO\_CHAR(SYSDATE, 'DL'));

END TODAY\_IS;

-----------------------------------------------------------

EXEC TODAY\_IS;

............................................................................

6666666666666666666666666666666666666666

............................................................................

create or replace procedure nurse\_sal\_calc(work\_hour IN number)

is

salary number(10);

begin

salary := work\_hour \* 150;

dbms\_output.put\_line('Nurses Salary is '||salary||' tk');

end nurse\_sal\_calc;

--------------------------------------------------------

exec nurse\_sal\_calc(8);

===========================================

=========================================

ALL FUNCTIONS

=========================================

----------------------------------------------------------------------

1111111111111111111111111111111111111111111

----------------------------------------------------------------------

CREATE OR REPLACE FUNCTION totalPatients

RETURN number IS

total number(2) := 0;

BEGIN

SELECT count(\*) into total

FROM u\_detail where u\_role='patient';

RETURN total;

END;

-----------------------------------------------------------------------

DECLARE

c number(2);

BEGIN

c := totalPatients();

dbms\_output.put\_line('Total number of Patients: ' || c);

END;

-----------------------------------------------------------------------

select totalpatients from dual;

------------------------------------------------------------------------

22222222222222222222222222222222222222222222

------------------------------------------------------------------------

CREATE OR REPLACE FUNCTION welcome\_msg ( p\_name IN VARCHAR2)

RETURN VARCHAR2

IS

BEGIN

RETURN ('Welcome '|| p\_name);

END;

------------------------------------------------------------------------

DECLARE

lv\_msg VARCHAR2(250);

BEGIN

lv\_msg := welcome\_msg ('Fahim');

dbms\_output.put\_line(lv\_msg);

END;

-------------------------------------------------------------------------

select welcome\_msg('Fahim') from dual;

-------------------------------------------------------------------------

333333333333333333333333333333333333333333333

-------------------------------------------------------------------------

create or replace FUNCTION calculati (x NUMBER,z VARCHAR2,y NUMBER)

RETURN NUMBER IS

w NUMBER;

BEGIN

IF z NOT IN ('+','-','\*','/')

THEN

raise\_application\_error(-2000,'PLEASE PUT THE RIGHT OPERATOR');

END IF;

CASE z

WHEN '+' THEN

w := x+y;

WHEN '-' THEN

w := x-y;

WHEN '\*' THEN

w := x\*y;

ELSE

w := x/y;

END CASE;

RETURN w;

END calculati;

------------------------------------------------------------------------

select calculati(100,'/',10) from dual;

------------------------------------------------------------------------

44444444444444444444444444444444444444444444

------------------------------------------------------------------------

CREATE OR REPLACE FUNCTION GET\_REC(USER\_N IN VARCHAR2)

RETURN DATE

IS INSERT\_DATE DATE;

BEGIN

SELECT MIN(INSERT\_TIME) INTO INSERT\_DATE

FROM U\_DETAIL\_REC

WHERE U\_NAME = USER\_N;

RETURN(INSERT\_DATE);

END;

-----------------------------------------------------------------------

select GET\_REC('fahim') from dual;

------------------------------------------------------------------------

55555555555555555555555555555555555555555555

-----------------------------------------------------------------------

create or replace FUNCTION get\_complete\_details

(in\_person\_id IN NUMBER)

RETURN VARCHAR2

IS person\_details VARCHAR2(230);

BEGIN

SELECT 'User ID-'||u\_detail.u\_id||' NAME-'|| u\_detail.u\_name||' Role-'|| u\_detail.u\_role ||', Prescription-'||prescription.p\_info||', Prescription ID-'||prescription.p\_id

INTO person\_details FROM prescription,u\_detail WHERE prescription.u\_id = u\_detail.u\_id and prescription.p\_id = in\_person\_id;

RETURN(person\_details);

END get\_complete\_details;

---------------------------------------------------------------------

select get\_complete\_details(9) from dual;

----------------------------------------------------------------------

6666666666666666666666666666666666666666666

-----------------------------------------------------------------------

CREATE OR REPLACE FUNCTION CALCULATE\_TAX (x NUMBER)

RETURN NUMBER

IS t NUMBER;

BEGIN

IF x<300000 THEN

raise\_application\_error(-20999,'As your income is less than 3 lakhs you do not have to pay tax');

END IF;

IF x<400000

THEN t := (5\*x)/100;

END IF;

IF x<700000

THEN t := (10\*x)/100;

END IF;

IF x<1100000

THEN t := (15\*x)/100;

END IF;

IF x<1600000

THEN t := (20\*x)/100;

END If;

IF x>1600000

THEN t := (25\*x)/100;

END IF;

RETURN (t);

END calculate\_tax;

-------------------------------------------------------------------------

select calculate\_tax(300000) from dual;

==========================================

END

=======================================