CSL3050: DBMS

Assignment 05

By Shashank Shekhar Asthana (B21CS093)

TABLE HOSPITAL

	HOSP_NAME	COUNTRY	ADDRESS
1	King Hamad University Hospital	Kingdom of Bahrain	Sheikh Isa bin Salman Bridge, Al Sayh, Busaiteen, Bahrain
2	Mayo Clinic	United States	4500 San Pablo Road, US
3	Bahrain Defence Force Hospital	Kingdom of Bahrain	Waly Alahed Avenue, West Riffa, Bahrain
4	Salamaniya Hospital	Kindgom of Bahrain	Salmaniya Medical Complex, Manama, Bahrain
5	St Thomas' Hospital London	United Kingdom	Westminster Bridge Rd, London
6	Al-Hilal Hospital	Kingdom of Bahrain	Al-Hilal Hospital, Muharraq, Bahrain
7	King Faisal Specialist Hospital Centre	Kingdom of Saudi Arabia	Al Mathar Ash Shamali, Riyadh 11564, Saudi Arabia
8	Emirates Hospital	United Arab Emirates	Jumeirah Beach Rd, Dubai, UAE
9	American Mission Hospital	Kingdom of Bahrain	Sheikh Essa Road, Manama, Bahrain
10	Dar Al Shifa Hospital	Kuwait	Beirut Street, Hawally, Kuwait

TABLE MEDICINE

	REG_NO	MED_NAME	PRICE	EXP_DATE
1	20	Adol Syrup	0.3	15-MAY-23
2	21	Amoxil Capsules	0.7	10-MAR-23
3	22	Aspirin	1	09-DEC-23
4	23	Bonjela Gel	0.4	23-SEP-24
5	24	Paracetamol	1.2	17-MAY-24
6	25	Atorvastatin	1.5	15-JUN-23
7	26	Coversyl	1	15-MAY-23
8	27	Diamicron	0.9	03-FEB-24
9	28	Lipitor	1.4	12-MAR-25
10	29	Glucophage	1.7	19-JUL-24

TABLE DOCTOR

	DOC_ID	DNAME	GENDER	QUALIFICATION	JOB_SPECIFICATION	HOSP_NAME	NEW_DOC_ID
1	1	Abdullah	M	MBBS	Dermatologist	Bahrain Defence Force Hospital	1
2	2	Ahmed	M	Board Certified	Ophthalmologist	King Hamad University Hospital	2
3	3	Ameera	F	MD	Pediatrician	Salamaniya Hospital	3
4	4	Ali	M	BPT	Physiotherapist	Al-Hilal Hospital	4
5	5	Carolina	F	MS	Surgeon	Mayo Clinic	5
6	6	Sarah	F	MD	Anatomy	Emirates Hospital	6
7	7	Qasim	M	MD	Radiologist	King Hamad University Hospital	7
8	8	Fatema	F	Residency	Anesthetist	Salamaniya Hospital	8
9	9	Khalid	M	MBBS	Psychiatrist	American Mission Hospital	9
10	10	Amal	F	D.M.	Nephrologist	King Hamad University Hospital	10

TABLE NURSE

	NURSE_ID	NAME	GENDER	DOC_ID	HOSP_NAME	NEW_DOC_ID
1	1	Sanaa	F	2	King Hamad University Hospital	2
2	2	Kathy	F	7	King Hamad University Hospital	7
3	3	Mary	F	5	Mayo Clinic	5
4	4	Ibrahim	M	8	Salamaniya Hospital	8
5	5	Anaya	F	3	Salamaniya Hospital	3
6	6	Asha	F	9	American Mission Hospital	9
7	7	Zainab	F	1	Bahrain Defence Force Hospital	1
8	8	Zeshan	M	10	King Hamad University Hospital	10
9	9	Adam	M	4	Al-Hilal Hospital	4
10	10	Hawra	F	6	Emirates Hospital	6

TABLE RECEPTION

	REC_ID	TEL_NO	EMAIL	HOSP_NAME
1	41	39123456	khuh@gmail.com	King Hamad University Hospital
2	42	39123456	khuh@gmail.com	King Hamad University Hospital
3	43	39456780	mayoclinic@facebook.com	Mayo Clinic
4	44	33456780	emirateshospital@yahoo.com	Emirates Hospital
5	45	33123456	bdf@yahoo.com	Bahrain Defence Force Hospital
6	46	33678901	alhilal@facebook.com	Al-Hilal Hospital
7	47	39678901	daralshifa@gmail.com	Dar Al Shifa Hospital
8	48	33912045	st.thomashospital@gmail.com	St Thomas' Hospital London
9	49	39912045	kingfaisalshrc@yahoo.com	King Faisal Specialist Hospital Centre
10	50	39678901	daralshifa@gmail.com	Dar Al Shifa Hospital

TABLE PATIENT

	SSN	FNAME	LNAME	AGE	GENDER	NURSE_ID	REC_ID
1	100000001	Sara	Majeed	27	F	7	46
2	100000002	Ahmed	Jamaal	59	M	3	43
3	100000003	Abdulla	Hameed	45	M	9	48
4	100000004	Mariam	Muhammad	40	F	1	41
5	100000005	Fatema	Hasan	64	F	6	47
6	100000006	Zainab	Abdulla	55	F	2	49
7	100000007	Khalil	Ibrahim	35	M	8	44
8	100000008	Alyaa	Husain	57	F	5	42
9	100000009	Khalid	Ahmed	60	M	4	50
10	100000010	Jawad	Ali	20	M	10	45

TABLE APPOINTMENT

	APPOINT_NO	APPOINT_DATE	APPOINT_TIME	REC_ID
1	1	03-MAY-22	01-OCT-23 07.05.03.000000000 AM	45
2	2	17-MAR-22	01-OCT-23 01.30.00.000000000 PM	49
3	3	22-DEC-22	01-OCT-23 09.45.00.000000000 AM	44
4	4	09-JUL-22	01-OCT-23 05.20.00.000000000 PM	48
5	5	03-MAY-22	01-OCT-23 12.15.00.000000000 PM	41
6	6	20-NOV-22	01-OCT-23 08.05.00.000000000 AM	47
7	7	12-JUL-22	01-OCT-23 03.40.00.000000000 PM	42
8	8	01-APR-22	01-OCT-23 02.00.00.000000000 PM	50
9	9	30-AUG-22	01-OCT-23 10.25.00.000000000 AM	43
10	10	15-JUN-22	01-OCT-23 04.50.00.000000000 PM	46

TABLE DIAGNOSIS

	DIAGNOS_NO	ISSUE_DATE	TREATMENT	REMARKS	NURSE_ID	DOC_ID	NEW_DOC_ID
1	31	04-JUN-22	Physiotherapy	Once a month	4	8	8
2	32	29-MAY-22	Aromatherapy	Twice a week	7	1	1
3	33	18-JUN-22	Cyrotherapy	Twice a month	5	3	3
4	34	07-JUL-22	Phototherapy	Once a month	10	6	6
5	35	13-AUG-22	Radiotherapy	Once in 3 months	2	7	7
6	36	03-DEC-22	Immunotherapy	Once a month	6	9	9
7	37	04-JUL-22	Monotherapy	Once a month	1	2	2
8	38	04-JUN-22	Pharmacotherapy	Once a month	8	10	10
9	39	19-JUL-22	Oxygen therapy	Once a week	3	5	5
10	40	25-JUN-22	Gene therapy	Once a month	9	4	4

TABLE VISIT

	VISIT_DATE	SSN	HOSP_NAME
1	10-OCT-21	100000009	Dar Al Shifa Hospital
2	15-MAY-20	100000005	Dar Al Shifa Hospital
3	30-JAN-22	100000006	King Faisal Specialist Hospital Centre
4	04-JUN-19	100000002	Mayo Clinic
5	24-DEC-20	100000010	Bahrain Defence Force Hospital
6	11-NOV-20	100000004	King Hamad University Hospital
7	06-FEB-21	100000001	Al-Hilal Hospital
8	21-APR-22	100000003	St Thomas' Hospital London
9	19-APR-22	100000007	Emirates Hospital
10	07-FEB-21	100000008	King Hamad University Hospital

TABLE PURCHASE

	SSN	REG_NO
1	100000007	20
2	(null)	23
3	100000004	28
4	100000009	25
5	100000003	21
6	100000006	29
7	100000008	22
8	100000002	27
9	100000007	24
10	(null)	26

TABLE EXAMINE

	SSN	DOC_ID	NEW_DOC_ID
1	100000004	1	1
2	100000006	2	2
3	100000002	3	3
4	100000009	4	4
5	100000008	5	5
6	100000005	6	6
7	100000001	7	7
8	100000007	8	8
9	100000003	9	9
10	100000010	10	10

TABLE MEDICINE_COUNTRY

	REG_NO	MAN_COUNTRY
1	20	United States
2	21	Australia
3	22	United States
4	23	Germany
5	24	United Kingdom
6	25	United Kingdom
7	26	Kingdom of Saudi Arabia
8	27	Germany
9	28	Switzerland
10	29	Germany

Now we will answer the queries using the PL/SQL syntax

1. Merge the Tables “EXAMINE” and “VISIT” with “DIAGNOSIS” to reduce redundancy. (Hint: Match Doc_ID, SSN and VISIT_DATE/ISSUE_DATE).

Query:-

```
CREATE TABLE CONSULTATION
(
   CONSULTATION_NO NUMBER PRIMARY KEY,
   VISIT_DATE DATE,
   SSN NUMBER,
   DOC_ID NUMBER,
   TREATMENT VARCHAR2(80),
   REMARKS VARCHAR2(80)
);

INSERT INTO CONSULTATION (CONSULTATION_NO, SSN, DOC_ID, TREATMENT, REMARKS)
SELECT ROWNUM, E.SSN, E.DOC_ID, D.TREATMENT, D.REMARKS
FROM EXAMINE E
JOIN DIAGNOSIS D ON E.SSN = D.SSN AND E.DOC_ID = D.DOC_ID;

-- Merge data from VISIT
INSERT INTO CONSULTATION (CONSULTATION_NO, VISIT_DATE, SSN, DOC_ID)
SELECT ROWNUM, V.VISIT_DATE, V.SSN, E.DOC_ID
FROM VISIT V
JOIN EXAMINE E ON V.SSN = E.SSN;

Select * from CONSULTATION;
```

Output:-

	CONSULTATION_NO	VISIT_DATE	SSN	DOC_ID	TREATMENT	REMARKS
1	1	11-NOV-20	100000004	1	(null)	(null)
2	2	30-JAN-22	100000006	2	(null)	(null)
3	3	04-JUN-19	100000002	3	(null)	(null)
4	4	10-OCT-21	100000009	4	(null)	(null)
5	5	07-FEB-21	100000008	5	(null)	(null)
6	6	15-MAY-20	100000005	6	(null)	(null)
7	7	06-FEB-21	100000001	7	(null)	(null)
8	8	19-APR-22	100000007	8	(null)	(null)
9	9	21-APR-22	100000003	9	(null)	(null)
10	10	24-DEC-20	100000010	10	(null)	(null)

2. Output the Doctor Names, Nurse Names and recommended Treatment corresponding to Senior Citizen patients (age > 55).

Query:-

```
DECLARE
  CURSOR SeniorCitizenPatients IS
    SELECT P.FNAME, P.LNAME, D.DNAME, N.NAME, DG.TREATMENT
    FROM PATIENT P
    JOIN DOCTOR D ON P.NURSE_ID = D.DOC_ID
    JOIN NURSE N ON P.NURSE_ID = N.NURSE_ID
    JOIN DIAGNOSIS DG ON P.NURSE_ID = DG.NURSE_ID
    WHERE P.AGE > 55;
BEGIN
  DBMS_OUTPUT.PUT_LINE('Senior Citizen Patients:');
  FOR PatientData IN SeniorCitizenPatients LOOP
    DBMS_OUTPUT.PUT_LINE('Patient Name: ' || PatientData.FNAME || ' ' || PatientData.LNAME);
    DBMS_OUTPUT.PUT_LINE('Doctor Name: ' || PatientData.DNAME);
    DBMS_OUTPUT.PUT_LINE('Nurse Name: ' || PatientData.NAME);
    DBMS_OUTPUT.PUT_LINE('Recommended Treatment: ' || PatientData.TREATMENT);
    DBMS_OUTPUT.PUT_LINE('----------------------------------------');
  END LOOP;
END;
/
```

Output:-

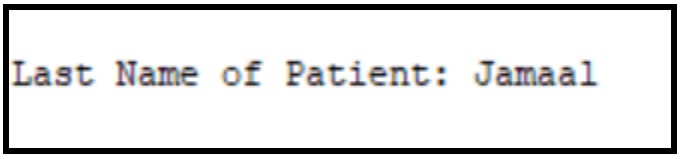
```
Senior Citizen Patients:
Patient Name: Khalid Ahmed
Doctor Name: Ali
Nurse Name: Ibrahim
Recommended Treatment: Physiotherapy
----------------------------------------
Patient Name: Alyaa Husain
Doctor Name: Carolina
Nurse Name: Anaya
Recommended Treatment: Cyrotherapy
----------------------------------------
Patient Name: Fatema Hasan
Doctor Name: Sarah
Nurse Name: Asha
Recommended Treatment: Immunotherapy
----------------------------------------
Patient Name: Ahmed Jamaal
Doctor Name: Ameera
Nurse Name: Mary
Recommended Treatment: Oxygen therapy
----------------------------------------
```

3. Write a PL/SQL code to find the last names of patients who were under the care of female nurses at the “Mayo Clinic”.

Query:-

```
CREATE OR REPLACE PROCEDURE PATIENTS_UNDER_FEMALE_NURSES AS
BEGIN
   FOR rec IN (
      SELECT DISTINCT p.LNAME
      FROM PATIENT p
      JOIN NURSE n ON p.NURSE_ID = n.NURSE_ID
      JOIN RECEPTION r ON n.HOSP_NAME = r.HOSP_NAME
      WHERE n.GENDER = 'F' AND r.HOSP_NAME = 'Mayo Clinic'
   ) LOOP
      DBMS_OUTPUT.PUT_LINE('Last Name of Patient: ' || rec.LNAME);
   END LOOP;
END;
/

SET SERVEROUTPUT ON;

EXEC PATIENTS_UNDER_FEMALE_NURSES;
```

Output:-



4. Write a PL/SQL code to find the name of the hospital having the highest number of “BPT”.

Query:-

```
CREATE OR REPLACE PROCEDURE HOSPITAL_WITH_HIGHEST_BPT AS
   v_max_count NUMBER;
   v_hospital_name VARCHAR2(30);
BEGIN
   SELECT COUNT(DOC_ID), HOSP_NAME
   INTO v_max_count, v_hospital_name
   FROM DOCTOR
   WHERE HOSP_NAME=(
      SELECT d.HOSP_NAME
      FROM DOCTOR d
      WHERE d.QUALIFICATION = 'BPT'
   )
```

```
    GROUP BY hosp_name;

    DBMS_OUTPUT.PUT_LINE('Hospital with Highest Number of "BPT" Doctors:');
    DBMS_OUTPUT.PUT_LINE('Hospital Name: ' || v_hospital_name);
    DBMS_OUTPUT.PUT_LINE('Number of "BPT" Doctors: ' || v_max_count);
END;
/

SET SERVEROUTPUT ON;

EXEC HOSPITAL_WITH_HIGHEST_BPT;
```

Output:-

```
Hospital with Highest Number of "BPT" Doctors:
Hospital Name: Al-Hilal Hospital
Number of "BPT" Doctors: 1
```

5. Check the validity of given email_ids (output the valid ones). Conditions for Valid email_ids:
a. End with '@gmail.com', '@yahoo.com', '@outlook.com', '@iCloud.com'
b. Allowed characters in Prefix: letters (a-z), numbers, underscores, periods, and dashes.
c. In Prefix, an underscore, period, or dash must be followed by one or more letters or numbers.

Query:-

```
CREATE OR REPLACE PROCEDURE VALID_EMAIL_CHECK AS
    v_email VARCHAR2(100);
BEGIN

    FOR email_rec IN (SELECT DISTINCT EMAIL FROM RECEPTION) LOOP
        v_email := email_rec.EMAIL;


        IF REGEXP_LIKE(v_email, '(@gmail\.com|@yahoo\.com|@outlook\.com|@iCloud\.com)$', 'i') THEN

            DBMS_OUTPUT.PUT_LINE('Valid Email: ' || v_email);
            IF REGEXP_LIKE(v_email, '^[\w\.-]+@[a-z0-9]+([._-][a-z0-9]+)*$', 'i') THEN
                DBMS_OUTPUT.PUT_LINE('Valid Email: ' || v_email);
```

```
        END IF;
      END IF;
   END LOOP;
END;
/

SET SERVEROUTPUT ON;

EXEC VALID_EMAIL_CHECK;
```

Output:-

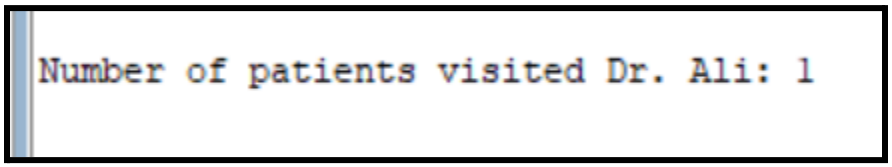
```
Valid Email: khuh@gmail.com
Valid Email: st.thomashospital@gmail.com
Valid Email: bdf@yahoo.com
Valid Email: daralshifa@gmail.com
Valid Email: emirateshospital@yahoo.com
Valid Email: kingfaisalshrc@yahoo.com
```

6. Write a PL/SQL code to count the number of patients who have visited Dr. “Ali”.

Query:-

```
DECLARE
   v_doctor_name VARCHAR2(30) := 'Ali';
   v_patient_count NUMBER;
BEGIN

   SELECT COUNT(DISTINCT P.SSN)
   INTO v_patient_count
   FROM PATIENT P
   JOIN EXAMINE E ON P.SSN = E.SSN
   JOIN DOCTOR D ON E.DOC_ID = D.DOC_ID
   WHERE D.DNAME = v_doctor_name;


   DBMS_OUTPUT.PUT_LINE('Number of patients visited Dr. ' || v_doctor_name || ': ' || v_patient_count);
END;
/

SET SERVEROUTPUT ON;

BEGIN
   NULL;
END;
/
```

Output:-



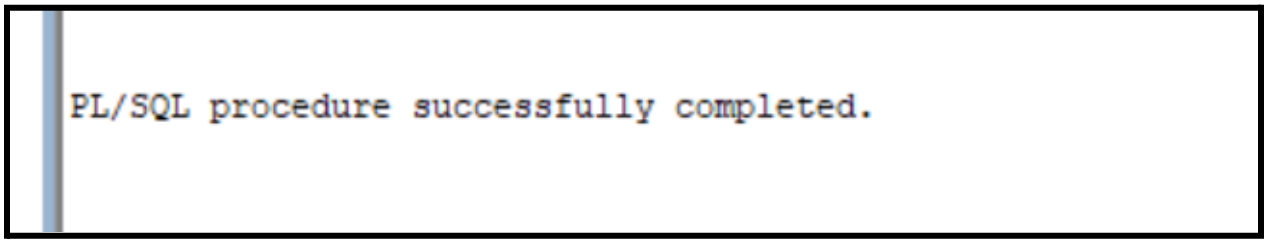
7. Change the format of Doc_ID to “[First 4 English letters of Hospital Name]_[4_digit_no]”. For e.g., change Doc_ID “1” to “BAHR_0001”. Update the same for all tables where Doc_ID is referenced.

Query:-

```
DECLARE
   v_hospital_name VARCHAR2(80);
   v_new_doc_id VARCHAR2(80);
BEGIN
   -- Looping through each doctor record
   FOR doctor_rec IN (SELECT DOC_ID, HOSP_NAME FROM DOCTOR)
   LOOP

      v_hospital_name := SUBSTR(REGEXP_REPLACE(doctor_rec.HOSP_NAME, '[^a-zA-Z]', ''), 1, 4);


      v_new_doc_id := RPAD(v_hospital_name, 4, ' ') || '_' || LPAD(doctor_rec.DOC_ID, 4, '0');


      UPDATE DOCTOR
      SET DOC_ID = v_new_doc_id
      WHERE DOC_ID = doctor_rec.DOC_ID;


      UPDATE NURSE
      SET DOC_ID = v_new_doc_id
      WHERE DOC_ID = doctor_rec.DOC_ID;

      UPDATE EXAMINE
      SET DOC_ID = v_new_doc_id
      WHERE DOC_ID = doctor_rec.DOC_ID;


      COMMIT;
   END LOOP;

   DBMS_OUTPUT.PUT_LINE('Doc_IDs have been updated.');
EXCEPTION
   WHEN OTHERS THEN
      DBMS_OUTPUT.PUT_LINE('Error: ' || SQLERRM);
```

```
END;
/
```

Output:-



8. Write a PL/SQL code to compute the total expenditure of a patient spent on Medicines.

Query:-

```
CREATE OR REPLACE PROCEDURE CalculateMedicineExpenditure(p_ssn NUMBER) IS
   v_total_expenditure NUMBER := 0;

BEGIN

   SELECT SUM(M.PRICE)
   INTO v_total_expenditure
   FROM PURCHASE P
   JOIN MEDICINE M ON P.REG_NO = M.REG_NO
   WHERE P.SSN = p_ssn;


   DBMS_OUTPUT.PUT_LINE('Total expenditure on medicines for SSN ' || p_ssn || ': ' || v_total_expenditure);

EXCEPTION
   WHEN NO_DATA_FOUND THEN
      DBMS_OUTPUT.PUT_LINE('No data found for SSN ' || p_ssn);
   WHEN OTHERS THEN
      DBMS_OUTPUT.PUT_LINE('Error: ' || SQLERRM);
END CalculateMedicineExpenditure;
/


BEGIN
   CalculateMedicineExpenditure(100000002);
END;
/
```

Output:-

```
Total expenditure on medicines for SSN 100000002: .9
```

9. Write a PL/SQL code to find the first and last name, age and nurse_id of all those female patients who have an appointment after July. (Take care of APPOINT_DATE).

Query:-

```
DECLARE
   CURSOR female_patients_cur IS
      SELECT P.FNAME, P.LNAME, P.AGE, P.NURSE_ID
      FROM PATIENT P
      JOIN APPOINTMENT A ON P.REC_ID = A.REC_ID
      WHERE P.GENDER = 'F' AND TO_CHAR(A.APPOINT_DATE, 'MM') > '07';
BEGIN
   FOR female_patient_rec IN female_patients_cur
   LOOP
      DBMS_OUTPUT.PUT_LINE('First Name: ' || female_patient_rec.FNAME ||
         ', Last Name: ' || female_patient_rec.LNAME ||
         ', Age: ' || female_patient_rec.AGE ||
         ', Nurse ID: ' || female_patient_rec.NURSE_ID);
   END LOOP; -- Added the missing LOOP here
END;
/
```

Output:-

```
First Name: Fatema, Last Name: Hasan, Age: 64, Nurse ID: 6
```

10. Write a PL/SQL code to give the medicine_country whose medicines are prescribed the most by each doctor respectively.

Query:-

```
DECLARE
   CURSOR doctor_cursor IS
```

```
    SELECT DISTINCT DOC_ID
    FROM DOCTOR;

  doctor_id NUMBER;
BEGIN
  FOR doctor_rec IN doctor_cursor
  LOOP
    doctor_id := doctor_rec.DOC_ID;
    DBMS_OUTPUT.PUT_LINE('Doctor ID: ' || doctor_id);

    FOR medicine_country_rec IN (SELECT MC.MAN_COUNTRY, COUNT(*) AS PRESCRIPTION_COUNT
                                 FROM MEDICINE_COUNTRY MC
                                 JOIN PURCHASE P ON MC.REG_NO = P.REG_NO
                                 JOIN EXAMINE E ON P.SSN = E.SSN
                                 WHERE E.DOC_ID = doctor_id
                                 GROUP BY MC.MAN_COUNTRY
                                 ORDER BY PRESCRIPTION_COUNT DESC)
    LOOP
      DBMS_OUTPUT.PUT_LINE('    Medicine Country: ' || medicine_country_rec.MAN_COUNTRY ||
        ', Prescription Count: ' || medicine_country_rec.PRESCRIPTION_COUNT);
      EXIT; -- Exit the loop after the first (most prescribed) medicine country
    END LOOP;
  END LOOP;
END;
/
```

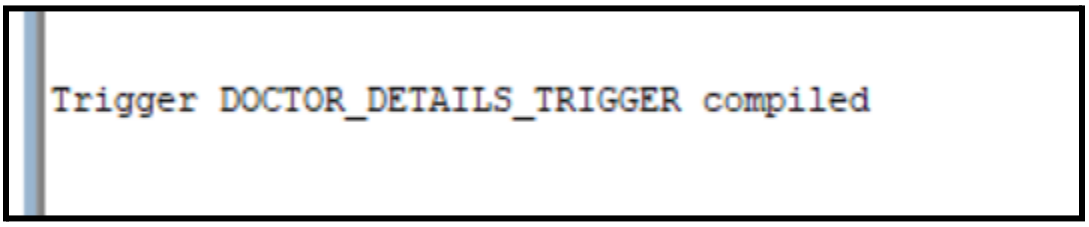
Output:-

```
Doctor ID: 1
    Medicine Country: Switzerland, Prescription Count: 1
Doctor ID: 2
    Medicine Country: Germany, Prescription Count: 1
Doctor ID: 3
    Medicine Country: Germany, Prescription Count: 1
Doctor ID: 4
    Medicine Country: United Kingdom, Prescription Count: 1
Doctor ID: 5
    Medicine Country: United States, Prescription Count: 1
Doctor ID: 6
Doctor ID: 7
Doctor ID: 8
    Medicine Country: United Kingdom, Prescription Count: 1
Doctor ID: 9
    Medicine Country: Australia, Prescription Count: 1
Doctor ID: 10
```

11. Write a PL/SQL code so that whenever a record is inserted or updated in Doctor Table, that doctor's details (all attributes) are displayed in output. (Hint: Trigger)

Query:-

```
CREATE OR REPLACE TRIGGER doctor_details_trigger
AFTER INSERT OR UPDATE ON DOCTOR
FOR EACH ROW
DECLARE
BEGIN
   DBMS_OUTPUT.PUT_LINE('Doctor Details -');
   DBMS_OUTPUT.PUT_LINE('Doctor ID: ' || :new.DOC_ID);
   DBMS_OUTPUT.PUT_LINE('Name: ' || :new.DNAME);
   DBMS_OUTPUT.PUT_LINE('Gender: ' || :new.GENDER);
   DBMS_OUTPUT.PUT_LINE('Qualification: ' || :new.QUALIFICATION);
   DBMS_OUTPUT.PUT_LINE('Job Specification: ' || :new.JOB_SPECIFICATION);
   DBMS_OUTPUT.PUT_LINE('Hospital Name: ' || :new.HOSP_NAME);
   DBMS_OUTPUT.PUT_LINE('Timestamp: ' || TO_CHAR(SYSDATE, 'YYYY-MM-DD HH24:MI:SS'));
END;
/

SET SERVEROUTPUT ON;
```



12. Insert any two new medicines in the Medicine Table (Like Diazepam, or Lorazepam, etc.) with some arbitrary attributes. Now using CURSOR, write a PL/SQL code to enter a new record in the Medicine_Country Table corresponding to the medicines names from the Medicine Table which are not listed in Medicine_Country. For these new entries, take the Man_Country as "Unknown" (e.g., Insert ("Diazepam", "Unknown")).

Query:-

```
INSERT INTO MEDICINE (REG_NO, MED_NAME, PRICE, EXP_DATE)
VALUES (30, 'Diazepam', 2.500, TO_DATE('2023-12-31', 'YYYY-MM-DD'));

INSERT INTO MEDICINE (REG_NO, MED_NAME, PRICE, EXP_DATE)
VALUES (31, 'Lorazepam', 2.800, TO_DATE('2023-12-31', 'YYYY-MM-DD'));
```

```
-- PL/SQL code to insert new records in Medicine_Country for medicines not listed with 'Unknown' as Man_Country
DECLARE
   v_med_name MEDICINE.MED_NAME%TYPE;
   v_man_country MEDICINE_COUNTRY.MAN_COUNTRY%TYPE := 'Unknown';
   v_med_count NUMBER := 0;
BEGIN
   -- Open a cursor to fetch medicine names not in Medicine_Country
   FOR med_rec IN (SELECT DISTINCT MED_NAME FROM MEDICINE WHERE MED_NAME NOT IN (SELECT
MED_NAME FROM MEDICINE_COUNTRY))
   LOOP
      -- Insert the new record in Medicine_Country
      INSERT INTO MEDICINE_COUNTRY (REG_NO, MAN_COUNTRY)
      SELECT REG_NO, v_man_country
      FROM MEDICINE
      WHERE MED_NAME = med_rec.MED_NAME;
      v_med_count := v_med_count + 1;
   END LOOP;

   -- Output the number of new records inserted
   DBMS_OUTPUT.PUT_LINE(v_med_count || ' new records inserted into Medicine_Country.');
END;
/
```

Output:-