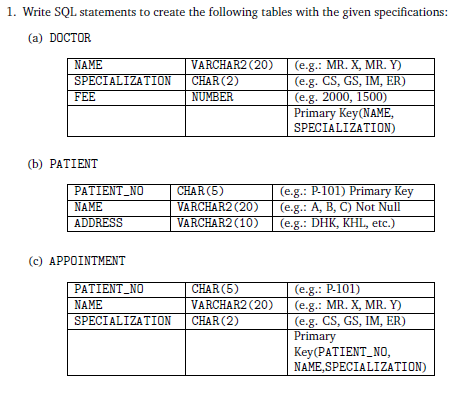
|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | | | | | | | |
|  | Report on Lab-03  DATABASE MANAGEMENT SYSTEMS LAB | | | | | |  | |
|  |  | | | | | | |  | | |
| Submitted by  Adid-Al-Mahamud Shazid  Student Id: 210042172  Department: CSE  Programme: SWE  Course Title: CSE 4308  Submitted to  Zannatun Naim Sristy  Lecturer, Department of CSE  August 31, 2023 | |  | | | | |
|  |
|  | | |  |
|  | |  | | | |  | | | |

**Introduction**

In the lab class, we were given five tasks to solve using SQL command line to understand the basics of data definition and data manipulation.

**Task 1**



* 1. **Solution**

CREATE TABLE DOCTOR

(

    NAME VARCHAR2(20),

    SPECIALIZATION CHAR(2),

    FEE NUMBER NOT NULL,

    CONSTRAINT PK\_DOCTOR PRIMARY KEY (NAME,SPECIALIZATION)

);

CREATE TABLE PATIENT

(

    PATIENT\_NO CHAR(5),

    NAME VARCHAR2(20) NOT NULL,

    ADDRESS VARCHAR2(10) NOT NULL,

    CONSTRAINT PK\_PATIENT PRIMARY KEY (PATIENT\_NO)

);

CREATE TABLE APPOINTMENT

(

    PATIENT\_NO CHAR(5),

    NAME VARCHAR2(20),

    SPECIALIZATION CHAR(2),

    CONSTRAINT PK\_APPOINTMENT PRIMARY KEY (PATIENT\_NO,NAME,SPECIALIZATION)

);

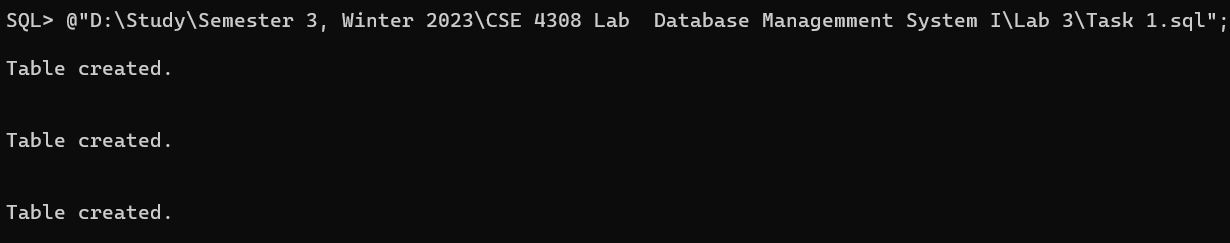
* 1. **Analysis and Explanation**

This task was easy to complete since we already learned how to create tables in the last lab also we learned about primary key at the previous lab.

* 1. **Difficulties**

I did not face any difficulties when doing this task and no mentionable issues were encountered.

* 1. **Output**

****

**Task 2**

1. Write SQL statements to perform the following alteration operations:
   1. Add a new attribute ‘APPOINTMENT\_DATE’ (DATE type) in APPOINTMENT table.
   2. Modify the PRIMARY KEY of APPOINTMENT table and add APPOINTMENT\_DATE too with the previous ones.
   3. Rename the attribute PATIENT\_NO, NAME from APPOINTMENT table to P\_NO and D\_NAME respectively.
   4. Rename the table APPOINTMENT to APPOINTMENT\_INFO.
   5. Add two foreign key constraints FK\_APPOINTMENT\_DOCTOR and FK\_APPOINTMENT\_PATIENT that identifies D\_NAME,SPECIALIZATION and P\_NO as foreign keys.
   6. **Solution**

ALTER TABLE APPOINTMENT ADD APPOINTMENT\_DATE DATE NOT NULL;

ALTER TABLE APPOINTMENT DROP CONSTRAINT PK\_APPOINTMENT;

ALTER TABLE APPOINTMENT ADD CONSTRAINT PK\_APPOINTMENT PRIMARY KEY (PATIENT\_NO, NAME, SPECIALIZATION,APPOINTMENT\_DATE);

ALTER TABLE APPOINTMENT RENAME COLUMN PATIENT\_NO TO P\_NO;

ALTER TABLE APPOINTMENT RENAME COLUMN NAME TO D\_NAME;

ALTER TABLE APPOINTMENT RENAME TO APPOINTMENT\_INFO;

ALTER TABLE APPOINTMENT\_INFO ADD CONSTRAINT FK\_APPOINTMENT\_DOCTOR FOREIGN KEY (D\_NAME, SPECIALIZATION) REFERENCES DOCTOR(NAME, SPECIALIZATION);

ALTER TABLE APPOINTMENT\_INFO ADD CONSTRAINT FK\_APPOINTMENT\_PATIENT FOREIGN KEY (P\_NO) REFERENCES PATIENT(PATIENT\_NO);

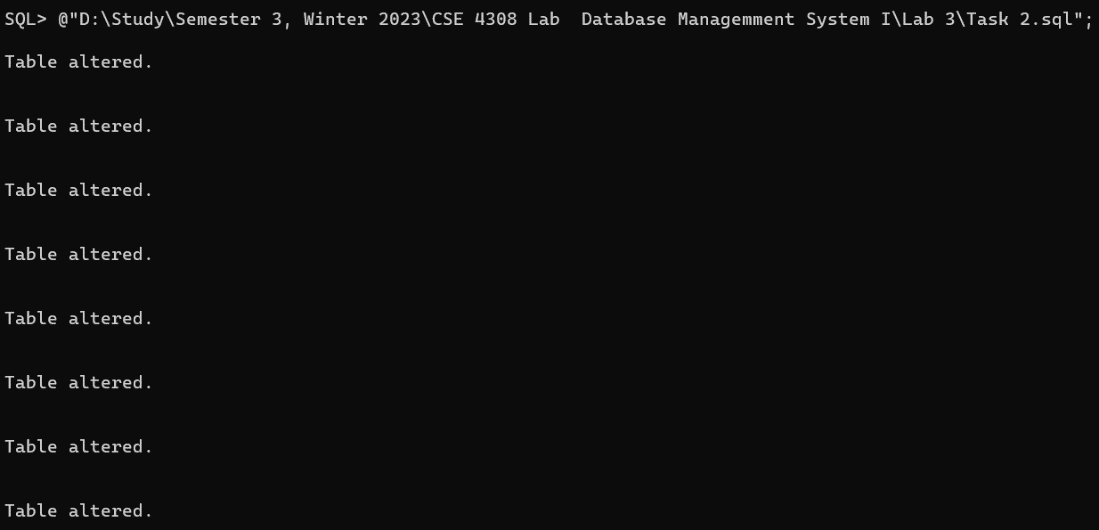
* 1. **Analysis and Explanation**

The task was easy to be solved following the manual of Lab-2 and Lab-3. Also, the class lectures helped me a lot to understand the basic of foreign key which is actually a referral of the primary key in another table.

* 1. **Difficulties**

During the first run I faced some difficulties as due to the alteration of table name and column names the name was not as like as it was. So, then I changed the name according to the alteration and my command run successfully.

* 1. **Output**

****

**Task 3**

3. Insert at least 3 records in each table following the example.

* 1. **Solution**

INSERT INTO DOCTOR VALUES('MR. X','CS',2000);

INSERT INTO DOCTOR VALUES('MR. Y','GS',2200);

INSERT INTO DOCTOR VALUES('MR. Z','IM',1000);

INSERT INTO DOCTOR VALUES('MR. P','ER',1800);

INSERT INTO DOCTOR VALUES('MR. Q','PQ',1500);

INSERT INTO PATIENT VALUES('P-101','A','DHK');

INSERT INTO PATIENT VALUES('P-102','B','KHL');

INSERT INTO PATIENT VALUES('P-103','C','RAJ');

INSERT INTO PATIENT VALUES('P-104','D','BAR');

INSERT INTO APPOINTMENT\_INFO VALUES('P-101','MR. X','CS','25-AUG-2023');

INSERT INTO APPOINTMENT\_INFO VALUES('P-102','MR. Y','GS','26-AUG-2023');

INSERT INTO APPOINTMENT\_INFO VALUES('P-103','MR. Z','IM','26-AUG-2023');

INSERT INTO APPOINTMENT\_INFO VALUES('P-104','MR. Q','PQ','27-AUG-2023');

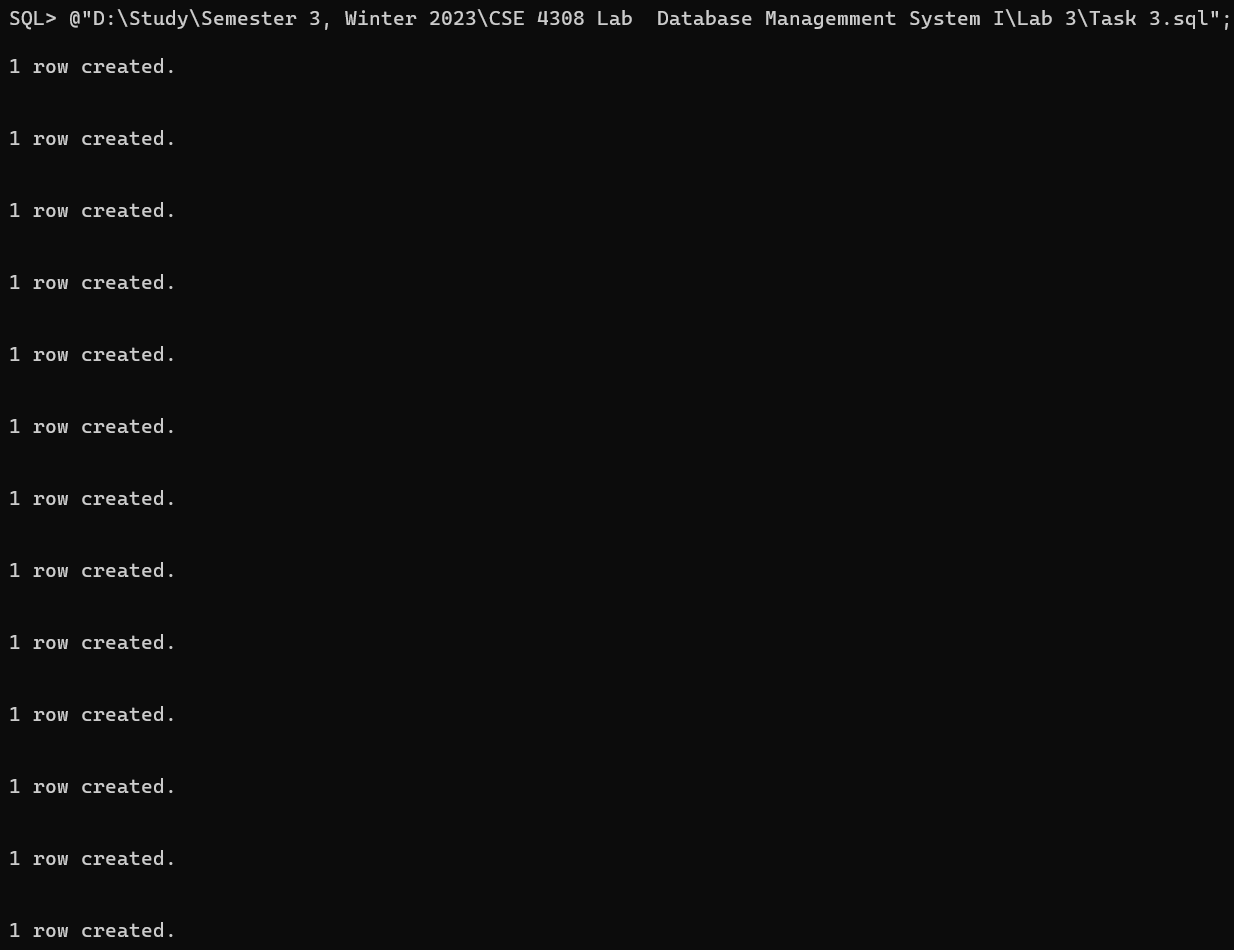
* 1. **Analysis and Explanation**

I inserted some records into the tables I created in task 1. This task was also easy to complete.

* 1. **Difficulties**

I did not face any difficulties when doing this task.

* 1. **Output**

****

**Task 4**

4. Write SQL statements to answer the following queries:

(a) Find all the Doctors’ names whose fees are less than 1500.

(b) Find all the Patients’ names who live in ‘KHL’ city.

(c) Show the result of Cartesian Product between PATIENT and APPOINTMENT\_INFO table.

(d) Show the result of Natural Join between PATIENT and APPOINTMENT\_INFO table.

(e) Find all the Patient’s names and their address who have an appointment today.

(f) Find all the Doctor-related information who have patients from ’DHK’.

(g) Find all Patient-related information who has an appointment with a doctor of ’GS’ specialization or a doctor whose fee is greater than 1500.

* 1. **Solution**

SELECT NAME FROM DOCTOR WHERE FEE<1500;

SELECT NAME FROM PATIENT WHERE ADDRESS='KHL';

SELECT \* FROM PATIENT, APPOINTMENT\_INFO WHERE PATIENT.PATIENT\_NO = APPOINTMENT\_INFO.P\_NO;

SELECT \* FROM PATIENT NATURAL JOIN APPOINTMENT\_INFO WHERE PATIENT.PATIENT\_NO = APPOINTMENT\_INFO.P\_NO;

SELECT NAME, ADDRESS FROM PATIENT, APPOINTMENT\_INFO WHERE PATIENT.PATIENT\_NO = APPOINTMENT\_INFO.P\_NO AND APPOINTMENT\_INFO.APPOINTMENT\_DATE= '26-AUG-2023';

SELECT DOCTOR.NAME,DOCTOR.SPECIALIZATION,DOCTOR.FEE FROM PATIENT, DOCTOR, APPOINTMENT\_INFO WHERE PATIENT.PATIENT\_NO = APPOINTMENT\_INFO.P\_NO AND

APPOINTMENT\_INFO.D\_NAME = DOCTOR.NAME AND APPOINTMENT\_INFO.SPECIALIZATION= DOCTOR.SPECIALIZATION AND PATIENT.ADDRESS='DHK';

SELECT PATIENT.NAME,PATIENT.PATIENT\_NO,PATIENT.ADDRESS FROM PATIENT, DOCTOR, APPOINTMENT\_INFO WHERE PATIENT.PATIENT\_NO = APPOINTMENT\_INFO.P\_NO AND

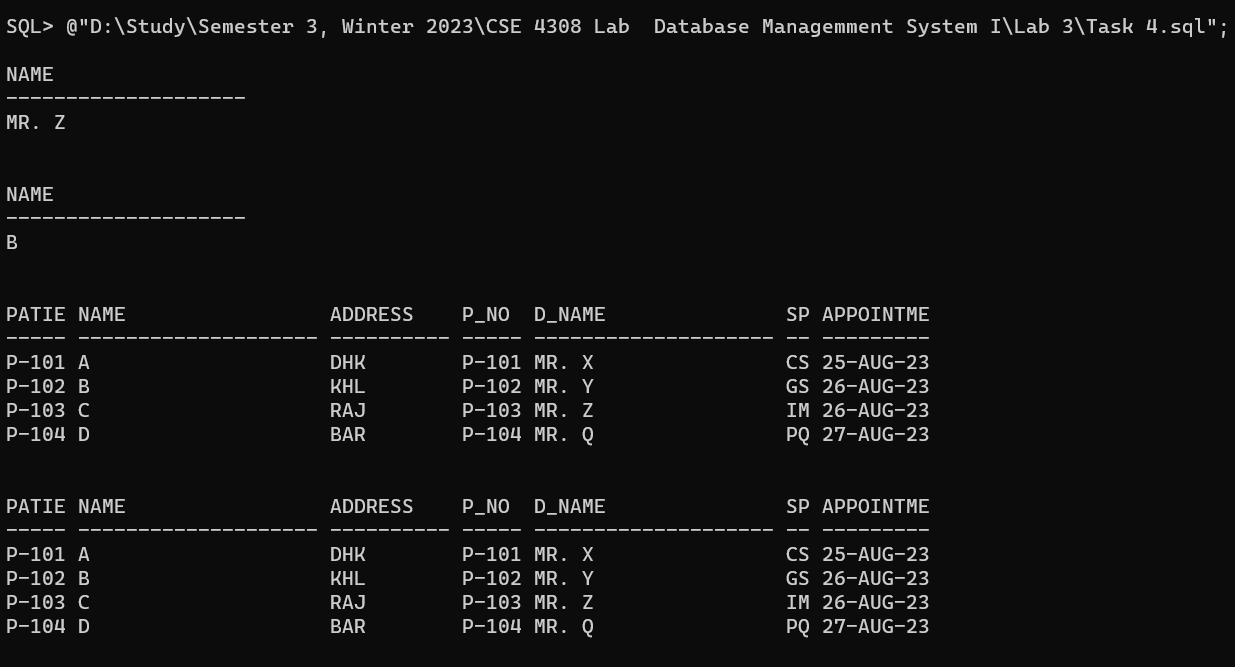
APPOINTMENT\_INFO.D\_NAME = DOCTOR.NAME AND APPOINTMENT\_INFO.SPECIALIZATION= DOCTOR.SPECIALIZATION AND DOCTOR.SPECIALIZATION='GS' AND DOCTOR.FEE>1500;

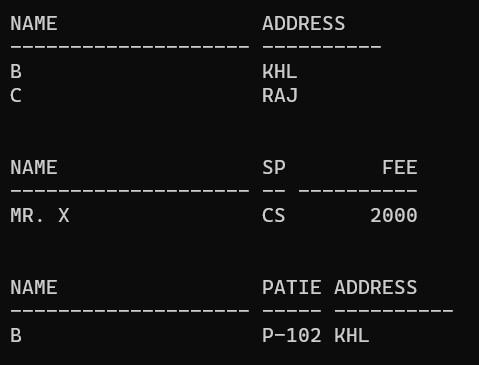
* 1. **Analysis and Explanation**

(a) And (b) are simple queries. (c) shows the cartesian product of cartesian product of Patient table and Appointment table. (d) shows the natural join of Patient table and Appointment table. The difference cartesian join and natural join is that cartesian join matches every element of between table 1 with every element of table 2 whereas natural join is only consisting of the matching elements of tables. In (e), (f), (g) I had to run query to find information from cartesian product of tables.

**Difficulties**

Rather selecting table\_name.column\_name, I selected only column\_name, thus didn’t get expected output but later I resolved that.

* 1. **Output**

****

**Task 5**

5. Write following DML statements:

(a) Update the NAME and ADDRESS of a tuple from ’A’ and ’DHK ’ to ’K’ and ’RAJ’ accordingly.

(b) Update the NAME of table DOCTOR from ’MR. Y’ as ’Ms. Y’.

(c) Delete Patient with PATIENT\_NO P-101.

(d) Delete all the information without deleting the table structure.

* 1. **Solution**

UPDATE PATIENT

SET NAME = 'K', ADDRESS = 'RAJ'

WHERE NAME = 'A' AND ADDRESS = 'DHK';

ALTER TABLE APPOINTMENT\_INFO DROP CONSTRAINT FK\_APPOINTMENT\_DOCTOR;

UPDATE APPOINTMENT\_INFO

SET APPOINTMENT\_INFO.D\_NAME= 'MS. Y' WHERE APPOINTMENT\_INFO.D\_NAME= 'MR. Y';

UPDATE DOCTOR

SET DOCTOR.NAME = 'MS. Y' WHERE DOCTOR.NAME = 'MR. Y';

ALTER TABLE APPOINTMENT\_INFO DROP CONSTRAINT FK\_APPOINTMENT\_PATIENT;

DELETE FROM APPOINTMENT\_INFO WHERE P\_NO = 'P-101';

DELETE FROM PATIENT WHERE PATIENT\_NO = 'P-101';

ALTER TABLE APPOINTMENT\_INFO ADD CONSTRAINT FK\_APPOINTMENT\_PATIENT FOREIGN KEY (P\_NO) REFERENCES PATIENT(PATIENT\_NO);

ALTER TABLE APPOINTMENT\_INFO DROP CONSTRAINT FK\_APPOINTMENT\_PATIENT;

DELETE FROM APPOINTMENT\_INFO;

DELETE FROM PATIENT;

ALTER TABLE APPOINTMENT\_INFO ADD CONSTRAINT FK\_APPOINTMENT\_PATIENT FOREIGN KEY (P\_NO) REFERENCES PATIENT(PATIENT\_NO);

* 1. **Analysis and Explanation**

In this task I had to update tables by performing query. I also Deleted all the information of the Patient table without deleting the table structure.

* 1. **Difficulties**

I was unable to update or delete tables due to the referencing of foreign key of another table. Then by discussing with the teacher I learnt that to update or delete such table, first I have to delete the query from the child table and then from the parent table. So, at first I removed the constraints of appointment table and updated the table, then updating the parent table I again assigned the constraints.

* 1. **Output**

****

**Conclusion**

As shown in the report, I have solved and tested the solutions for all five tasks given in the lab. All the commands used were written in visual studio code which was then saved with .sql extension. The .sql file was then run through the SQL command line to execute all the commands.