Database Project
By Modib Qadir

QUERIES:

1) List all the doctors RICHARD MILLER is consulting.

```
sqlite> select D.D_NAME
   ...> from DOCTORS D
   ...> join P_ASSIGNMENT PA ON D.D_ID = PA.D_ID
   ...> join PATIENTS P ON PA.P_ID = P.P_ID
   ...> where P.P_NAME = 'RICHARD MILLER';
"WILLIAM BROWN"
"PATRICIA JOHNSON"
```

```
SELECT D.D_NAME
FROM DOCTORS D
JOIN P_ASSIGNMENT PA ON D.D_ID = PA.D_ID
JOIN PATIENTS P ON PA.P_ID = P.P_ID
WHERE P.P_NAME = 'RICHARD MILLER';
```

Result:

"WILLIAM BROWN"
"PATRICIA JOHNSON"

2) Find all the test results of cancer patients. (Note: There may be different type of cancer)

```
sqlite> SELECT P.P_NAME, P.P_ID, P.P_DISEASE, T.T_NAME, T.T_RESULT
   ...> FROM TESTS T
   ...> JOIN PATIENTS P ON T.P_ID = SUBSTR(P.P_ID, -4)
   ...> WHERE P.P_DISEASE LIKE '%cancer%';
"MARK WHITE",000001008,"Prostate Cancer",Test29,Negative
"HELEN HARRIS",000001029,Cancer,Test48,Positive
"SANDRA MARTIN",000001030,"Bladder Cancer",Test4,Positive
```

```
SELECT P.P_NAME, P.P_ID, P.P_DISEASE, T.T_NAME, T.T_RESULT
FROM TESTS T
JOIN PATIENTS P ON T.P_ID = SUBSTR(P.P_ID, -4)
WHERE P.P_DISEASE LIKE '%cancer%';
```

THE (SUBSTR(P.P_ID,-4) is there because in the Patients table I noticed that the id started with 4 zeros so they didn't match. So with SUBSTR(P.P_ID, -4) it pretty much returns the last 4.

Result:

"MARK WHITE",000001008,"Prostate Cancer",Test29,Negative
"HELEN HARRIS",000001029,Cancer,Test48,Positive
"SANDRA MARTIN",000001030,"Bladder Cancer",Test4,Positive

3) List all the instruments produced by a manufacturer whose name starts with "S".

```
sqlite> SELECT *
   ...> FROM INSTRUMENTS
   ...> WHERE I_MANUFACTURER LIKE 'S%';
9,Instrument9,"Stryker Corporation"
sqlite>
```

```
SELECT *
FROM INSTRUMENTS
WHERE I_MANUFACTURER LIKE 'S%';
```

Result:

9,Instrument9,"Stryker Corporation"

4) Find the most experienced doctor in the hospital.

```
sqlite> SELECT D_NAME, D_YEARS_OF_EXPERIENCE
   ...> FROM DOCTORS
   ...> ORDER BY CAST(D_YEARS_OF_EXPERIENCE AS INTEGER) DESC
   ...> LIMIT 1;
"ELIZABETH BROWN",31
```

SELECT D_NAME, D_YEARS_OF_EXPERIENCE
FROM DOCTORS
ORDER BY CAST(D_YEARS_OF_EXPERIENCE AS INTEGER) DESC
LIMIT 1;

Result:

"ELIZABETH BROWN",31

5) List all the patients of doctor JAMES SMITH who live in the same street and same city as him.

```
sqlite> SELECT P.P_NAME, P.P_STREET, P.P_CITY
   ...> FROM PATIENTS P
   ...> JOIN P_ASSIGNMENT PA ON P.P_ID = PA.P_ID
   ...> JOIN DOCTORS D ON PA.D_ID = D.D_ID
   ...> WHERE D.D_NAME = 'JAMES SMITH' AND P.P_STREET = D.D_STREET AND P.P_CITY = D.D_CITY;
"BETTY WHITE","Sterling Place",Reading
```

SELECT P.P_NAME, P.P_STREET, P.P_CITY
FROM PATIENTS P
JOIN P_ASSIGNMENT PA ON P.P_ID = PA.P_ID
JOIN DOCTORS D ON PA.D_ID = D.D_ID
WHERE D.D_NAME = 'JAMES SMITH' AND P.P_STREET = D.D_STREET AND
P.P_CITY = D.D_CITY;

Result:

"BETTY WHITE","Sterling Place",Reading

6) Find the nurses who assist at least two doctors. Display nurse name and the number of doctors he/she is assisting

```
sqlite> SELECT N.N_NAME, COUNT(DISTINCT NA.D_ID) AS Doctors_Assisted
   ...> FROM NURSES N
   ...> JOIN N_ASSISTS NA ON N.N_ID = NA.N_ID
   ...> GROUP BY N.N_NAME
   ...> HAVING COUNT(DISTINCT NA.D_ID) >= 2;
"CAROL GARCIA",2
"JENNIFER DAVIS",3
"MARGARET MOORE",2
"MARIA MILLER",2
"MICHELLE CLARK",3
"SANDRA MARTIN",2
"SHARON ROBINSON",2
"SOLINDA WILLIAMS",3
```

```
SELECT N.N_NAME, COUNT(DISTINCT NA.D_ID) AS Doctors_Assisted
FROM NURSES N
JOIN N_ASSISTS NA ON N.N_ID = NA.N_ID
GROUP BY N.N_NAME
HAVING COUNT(DISTINCT NA.D_ID) >= 2;
```

Result:

```
"CAROL GARCIA",2
"JENNIFER DAVIS",3
"MARGARET MOORE",2
"MARIA MILLER",2
"MICHELLE CLARK",3
"SANDRA MARTIN",2
"SHARON ROBINSON",2
"SOLINDA WILLIAMS",3
```

7) List the doctors and the number of nurses they have in the descending order of their number.

```
sqlite> SELECT D.D_NAME, COUNT(DISTINCT NA.N_ID) AS Nurses_Count
   ...> FROM DOCTORS D
   ...> JOIN N_ASSISTS NA ON D.D_ID = NA.D_ID
   ...> GROUP BY D.D_NAME
   ...> ORDER BY Nurses_Count DESC;
"PATRICIA JOHNSON",5
"ELIZABETH BROWN",4
"WILLIAM BROWN",3
"ROBERT WILLIAMS",3
"MICHAEL JONES",3
"JOHN JOHNSON",3
"JAMES SMITH",3
"MARY SMITH",2
"LINDA WILLIAMS",2
"BARBARA JONES",2
```

```
SELECT D.D_NAME, COUNT(DISTINCT NA.N_ID) AS Nurses_Count
FROM DOCTORS D
JOIN N_ASSISTS NA ON D.D_ID = NA.D_ID
GROUP BY D.D_NAME
ORDER BY Nurses_Count DESC;
```

Result:

```
"PATRICIA JOHNSON",5
"ELIZABETH BROWN",4
"WILLIAM BROWN",3
"ROBERT WILLIAMS",3
"MICHAEL JONES",3
"JOHN JOHNSON",3
"JAMES SMITH",3
"MARY SMITH",2
"LINDA WILLIAMS",2
"BARBARA JONES",2
```

8) Find all the nurses who are not assigned to any doctors.

```
sqlite> SELECT N_NAME
   ...> FROM NURSES
   ...> WHERE N_ID NOT IN (SELECT DISTINCT N_ID FROM N_ASSISTS);
"MARY SMITH"
```

SELECT N_NAME
FROM NURSES
WHERE N_ID NOT IN (SELECT DISTINCT N_ID FROM N_ASSISTS);

Result:

"MARY SMITH"

9) Increment years of experience of all the female doctors by 5.

```
sqlite> UPDATE DOCTORS
   ...> SET D_YEARS_OF_EXPERIENCE = CAST(CAST(D_YEARS_OF_EXPERIENCE AS INTEGER) + 5 AS TEXT)
   ...> WHERE D_GENDER = 'f';
sqlite> SELECT D_NAME, D_YEARS_OF_EXPERIENCE
   ...> FROM DOCTORS
   ...> WHERE D_GENDER = 'f';
"MARY SMITH",15
"PATRICIA JOHNSON",10
"LINDA WILLIAMS",33
"BARBARA JONES",24
"ELIZABETH BROWN",36
```

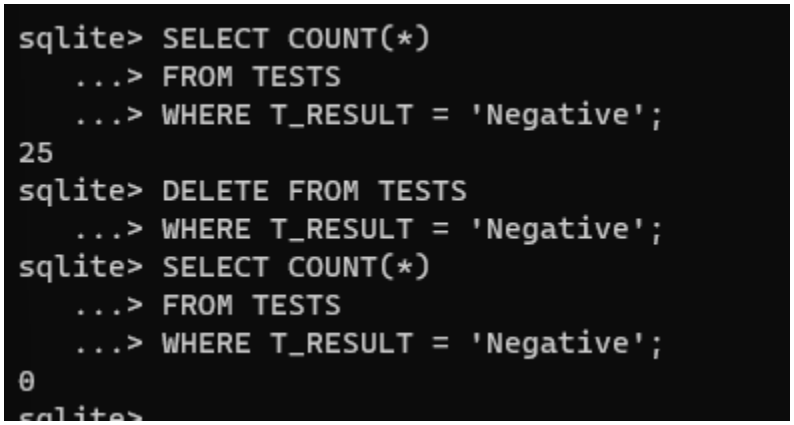
(Increments years of experience)
UPDATE DOCTORS
SET D_YEARS_OF_EXPERIENCE = D_YEARS_OF_EXPERIENCE + 5
WHERE D_GENDER = 'Female';

(shows result)
SELECT D_NAME, D_YEARS_OF_EXPERIENCE
FROM DOCTORS
WHERE D_GENDER = 'f';

Result:

"PATRICIA JOHNSON",10
"LINDA WILLIAMS",33
"BARBARA JONES",24
"ELIZABETH BROWN",36

10) Delete all the tests whose result is negative.



(Deletes the Negative tests)
DELETE FROM TESTS
WHERE T_RESULT = 'Negative';

(Checks how many tests are Negative)
SELECT COUNT(*)
FROM TESTS
WHERE T_RESULT = 'Negative';

Result:

(Before Deletion)

25

(After Deletion)

0