

מיני פרויקט בקורס בסיסי נתונים:

מערכת תרומות הדם בארגון מד"א

מגישות:

שובל חדד 206846297

רעות צפלביץ 213114499

## תוכן עניינים:

### שלב א:

3.....	תיאור האירגון
4.....	תיאור הנתונים הנשמרים
5 .....	טבלת ERD ו
6.....	טבלאות DSD ו
7-9.....	תוכן הקובץ createTable ו
10.....	פעולת describe ו
11.....	קובץ dropTable ו
11.....	קובץ selectAll ו
12-15.....	קובץ insertTable ו
16 .....	גיבוי ושחזור הנתונים

### שלב ב:

שאלות ללא פרמטרים:

17-20.....	שאלות SELECT ו
21-23.....	שאלות DELETE ו
24-27.....	שאלות UPDATE ו
28-35.....	שאלות עם פרמטרים
36-38.....	אילוצים
39.....	שינויים

## שלב א - תיאור המערכת

### תיאור הארגון:

בפרויקט נבנה בסיס נתונים לתרומות דם לארגון מד"א. מגן דוד אדום (מד"א) הוא הארגון הלאומי של ישראל לסיוע רפואי והצלה במצבי חירום. הארגון פועל בתחומים שונים כמו סיוע רפואי חירום, סיוע פסיכוסוציאלי, תרומת דם, סיוע ראשוני, ועוד. מגן דוד אדום גם משמש כגוף להשתתפות בפעולות הומניטריות ברחבי העולם, כולל בזמן מבצעים צבאיים ואסונות טבע. הארגון מתבסס על צוות מתנדבים מוכשר ומקצועי, שמאומן לטיפול במגוון רחב של תרחישים חירום. מד"א מסתמך על תרומות דם לספק חומרים חיוניים לטיפול בפצועים ובחולים במצבי חירום. בפרויקט נתמקד בתחום תרומות הדם. מד"א עורך התרומות דם בנקודות התרמה שונות. חלקן הן ניידות אשר פרושות לאורך הארץ, וחלקן הן התרומות דם בשיתוף פעולה עם ארגונים שונים בכל הארץ אשר דואגים לתורמים פוטנציאליים. הפונקציונאליות העיקריות שנרצה לבצע ע"י המערכת הן למשל: בדיקת נתונים של תורם, כמו סוג הדם שלו ומתי הוא תרם לאחרונה, בשביל לבדוק אם הוא רשאי לתרום שוב, בנוסף, נוכל לבדוק נתוני אספקת דם לבתי החולים שונים, ונוכל לדעת כמה מנות תקינות מכל סוג דם יש במלאי בנק בדם.

## תיאור הנתונים הנשמרים:

### ישויות:

1. אדם (ישות אב)

2. תורם (ישות בן)

3. תרומת הדם

4. מקום ההתרמה

5. פרמדיק (ישות בן, האדם שלוקח את הדם מהתורם)

6. נמען (המוסד המקבל את הדם)

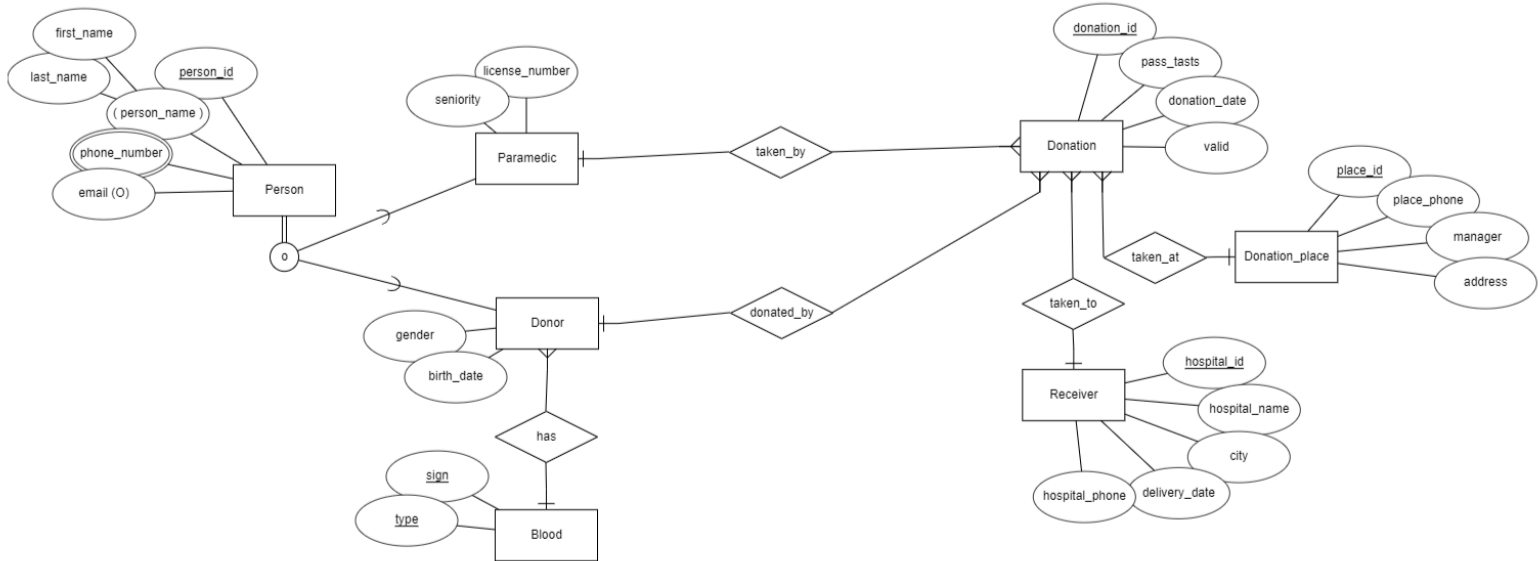
7. דם

היישות המרכזית היא יישות של תרומה, שיש לה מפתחות זרים של רוב הישויות האחרות. יש לנו ישות של אדם עם שדות בסיסיים כמו מספר ת"ז, מספר פלאפון וכו'. ממנה יורשים הישויות תורם ופרמדיק, תורם הוא אדם שממנו נלקח הדם והפרמדיק הוא האדם שלקח את הדם מן התורם. כיוון שבישות של אדם עשינו את התכונה מספר טלפון ככזו שיכולה לקבל מספר ערכים, ניצור גם עבור תכונה זו טבלה.

כמו כן יש ישות של מקום ההתרמה בו נערכה תרומת הדם וישות של ביה"ח שמקבל את מנת הדם לבסוף.

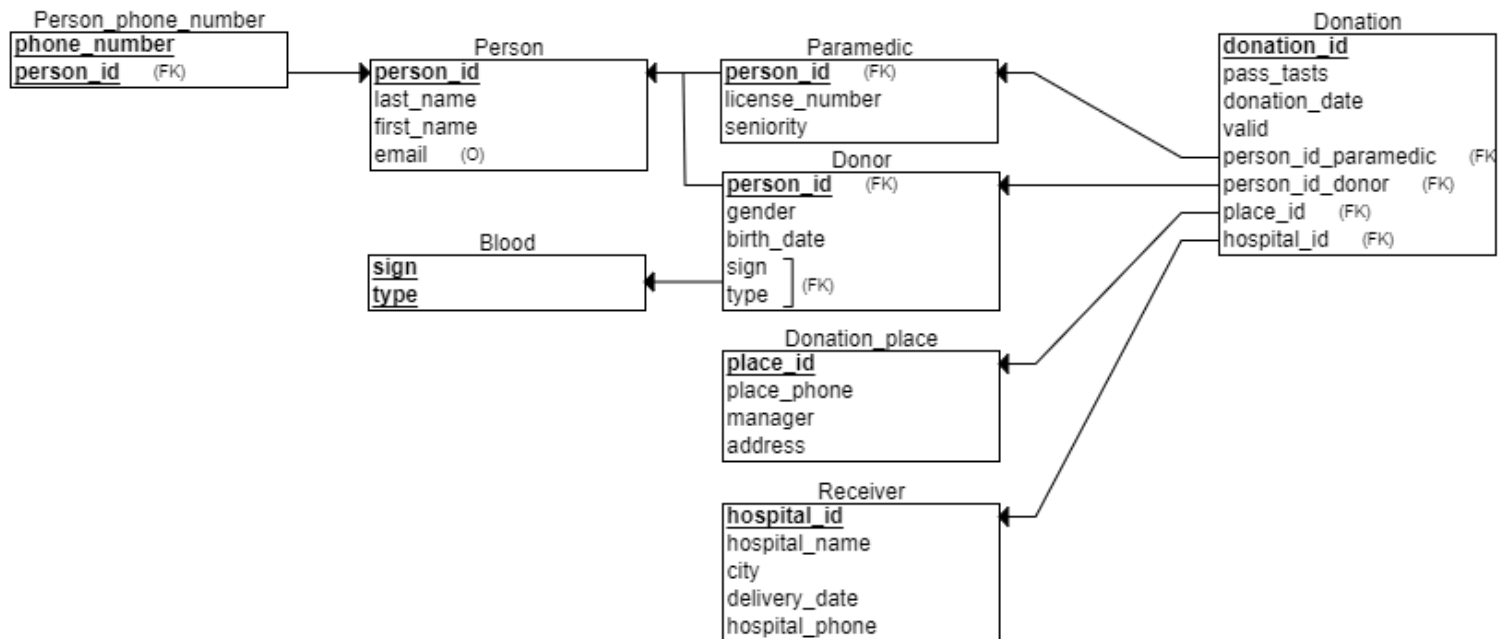
## טבלת ERD -

(בתוכנת ERDPlus)

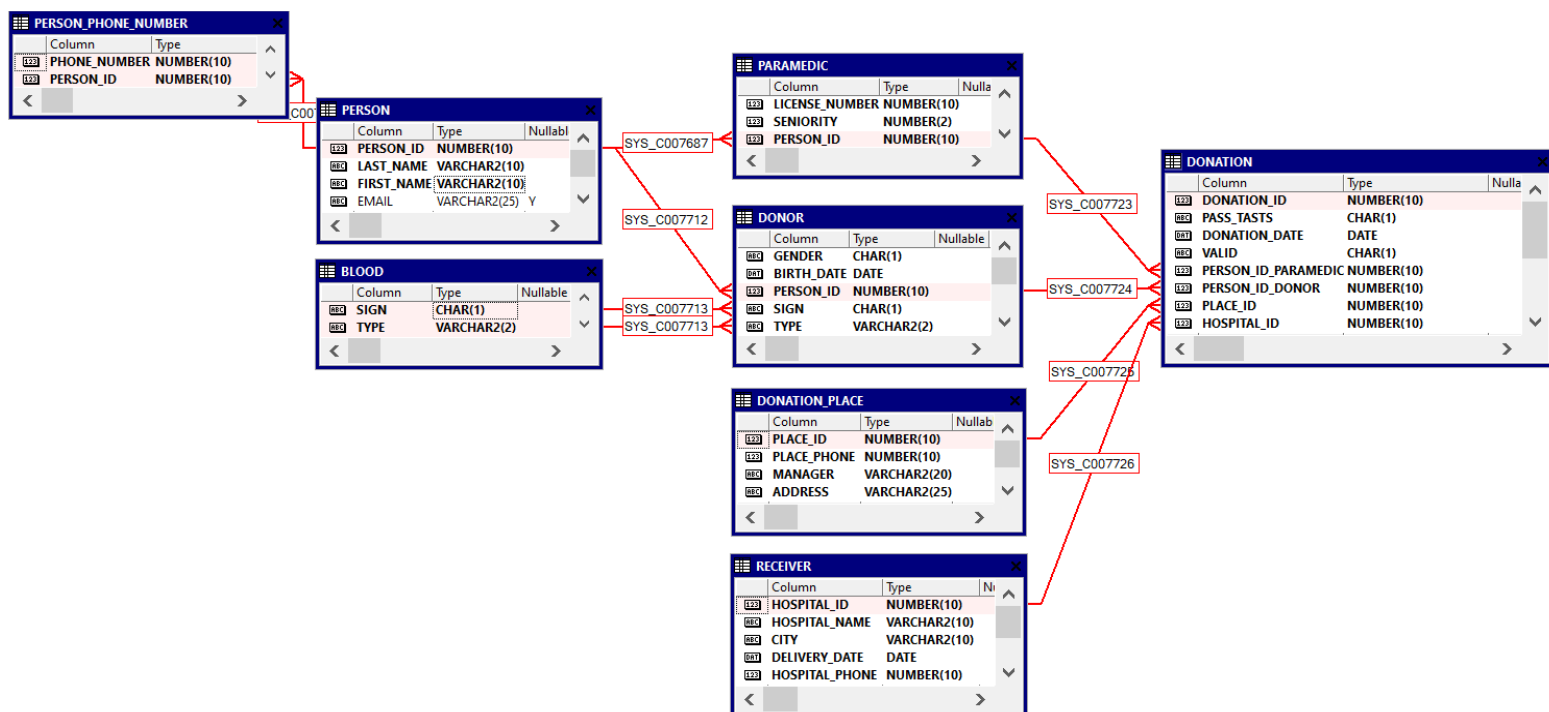


## טבלת DSD -

(בתוכנת ERDPlus)



(בתוכנת Oracle)



## קובץ createTable -

(את הקובץ createTables.sql ובו נמצא הסקריפט ליצירת הטבלאות העלנו גם לגיט)

```
CREATE TABLE Person
(
    person_id NUMERIC(10) NOT NULL,
    last_name VARCHAR(10) NOT NULL,
    first_name VARCHAR(10) NOT NULL,
    email VARCHAR(25),
    PRIMARY KEY (person_id)
);

CREATE TABLE Paramedic
(
    license_number NUMERIC(10) NOT NULL,
    seniority NUMERIC(2) NOT NULL,
    person_id NUMERIC(10) NOT NULL,
    PRIMARY KEY (person_id),
    FOREIGN KEY (person_id) REFERENCES Person(person_id)
);

CREATE TABLE Blood
(
    sign CHAR(1) NOT NULL,
    type VARCHAR(2) NOT NULL,
    PRIMARY KEY (sign, type)
);
```

```
CREATE TABLE Receiver
```

```
(  
    hospital_id NUMERIC(10) NOT NULL,  
    hospital_name VARCHAR(10) NOT NULL,  
    city VARCHAR(10) NOT NULL,  
    delivery_date DATE NOT NULL,  
    hospital_phone NUMERIC(10) NOT NULL,  
    PRIMARY KEY (hospital_id)  
);
```

```
CREATE TABLE Donation_place
```

```
(  
    place_id NUMERIC(10) NOT NULL,  
    place_phone NUMERIC(10) NOT NULL,  
    manager VARCHAR(20) NOT NULL,  
    address VARCHAR(25) NOT NULL,  
    PRIMARY KEY (place_id)  
);
```

```
CREATE TABLE Person_phone_number
```

```
(  
    phone_number NUMERIC(10) NOT NULL,  
    person_id NUMERIC(10) NOT NULL,  
    PRIMARY KEY (phone_number, person_id),  
    FOREIGN KEY (person_id) REFERENCES Person(person_id)  
);
```

```
CREATE TABLE Donor
```



```

(
    gender CHAR(1) NOT NULL,
    birth_date DATE NOT NULL,
    person_id NUMERIC(10) NOT NULL,
    sign CHAR(1) NOT NULL,
    type VARCHAR(2) NOT NULL,
    PRIMARY KEY (person_id),
    FOREIGN KEY (person_id) REFERENCES Person(person_id),
    FOREIGN KEY (sign, type) REFERENCES Blood(sign, type)
);

CREATE TABLE Donation
(
    donation_id NUMERIC(10) NOT NULL,
    pass_tests CHAR(1) NOT NULL,
    donation_date DATE NOT NULL,
    valid CHAR(1) NOT NULL,
    person_id_paramedic NUMERIC(10) NOT NULL,
    person_id_donor NUMERIC(10) NOT NULL,
    place_id NUMERIC(10) NOT NULL,
    hospital_id NUMERIC(10) NOT NULL,
    PRIMARY KEY (donation_id),
    FOREIGN KEY (person_id_paramedic) REFERENCES Paramedic(person_id),
    FOREIGN KEY (person_id_donor) REFERENCES Donor(person_id),
    FOREIGN KEY (place_id) REFERENCES Donation_place(place_id),
    FOREIGN KEY (hospital_id) REFERENCES Receiver(hospital_id)
);

```

## פעולת describe -

ניתן לראות שהטבלאות נוצרו ע"י שימוש בפעולה describe:

Tables	
>	APEX\$_ACL
>	APEX\$_WS_FILES
>	APEX\$_WS_HISTORY
>	APEX\$_WS_LINKS
>	APEX\$_WS_NOTES
>	APEX\$_WS_ROWS
>	APEX\$_WS_TAGS
>	APEX\$_WS_WEBPG_SECTION_HISTORY
>	APEX\$_WS_WEBPG_SECTIONS
>	DEMO_CUSTOMERS
>	DEMO_ORDER_ITEMS
>	DEMO_ORDERS
>	DEMO_PRODUCT_INFO
>	DEMO_STATES
>	DEMO_USERS
>	DEPT
>	EMP
>	PERSON
>	PARAMEDIC
>	BLOOD
>	RECEIVER
>	DONATION_PLACE
>	PERSON_PHONE_NUMBER
>	DONOR
>	DONATION

Columns of BLOOD					
	Name	Type	Nullable	Default	Comments
1	SIGN	CHAR(1)			
2	TYPE	CHAR(2)			

Columns of PARAMEDIC					
	Name	Type	Nullable	Default	Comments
1	LICENSE_NUMBER	NUMBER(10)			
2	PERSON_ID	NUMBER(10)			

Columns of RECEIVER					
	Name	Type	Nullable	Default	Comments
1	HOSPITAL_ID	NUMBER(10)			
2	HOSPITAL_NAME	VARCHAR2(10)			
3	CITY	VARCHAR2(10)			
4	DELIVERY_DATE	DATE			
5	HOSPITAL_PHONE	NUMBER(10)			

Columns of PERSON					
	Name	Type	Nullable	Default	Comments
1	PERSON_ID	NUMBER(10)			
2	LAST_NAME	VARCHAR2(10)			
3	FIRST_NAME	VARCHAR2(10)			
4	EMAIL	VARCHAR2(25)	Y		

Columns of PERSON_PHONE_NUMBER					
	Name	Type	Nullable	Default	Comments
1	PHONE_NUMBER	NUMBER(10)			
2	PERSON_ID	NUMBER(10)			

Columns of DONOR					
	Name	Type	Nullable	Default	Comments
1	GENDER	CHAR(1)			
2	BIRTH_DATE	DATE			
3	PERSON_ID	NUMBER(10)			
4	SIGN	CHAR(1)			
5	TYPE	CHAR(2)			

Columns of DONATION_PLACE					
	Name	Type	Nullable	Default	Comments
1	PLACE_ID	NUMBER(10)			
2	PLACE_PHONE	NUMBER(10)			
3	MANAGER	VARCHAR2(20)			
4	ADDRESS	VARCHAR2(25)			

Columns of DONATION					
	Name	Type	Nullable	Default	Comments
1	DONATION_ID	NUMBER(10)			
2	PASS_TESTS	CHAR(1)			
3	DONATION_DATE	DATE			
4	VALID	CHAR(1)			
5	LICENSE_NUMBER	NUMBER(10)			
6	PERSON_ID_PARAMEDIC	NUMBER(10)			
7	PERSON_ID_DONOR	NUMBER(10)			
8	PLACE_ID	NUMBER(10)			
9	HOSPITAL_ID	NUMBER(10)			

## קובץ dropTable -

כשכתבנו את הקובץ dropTables.sql סידרנו את סדר הטבלאות כך שקודם ימחקו הטבלאות שיש בהם מפתחות זרים לטבלאות אחרות ורק לאחר מכן הטבלאות שלא מסתמכות על טבלאות אחרות, כדי שתתאפשר המחיקה. (את הקובץ העלנו גם לגיט)

```
DROP TABLE Donation;

DROP TABLE Donor;

DROP TABLE Person_phone_number;

DROP TABLE Paramedic;

DROP TABLE Blood;

DROP TABLE Receiver;

DROP TABLE Donation_place;

DROP TABLE Person;
```

## קובץ selectAll -

קובץ selectAll.sql מאפשר לנו לראות את כל הטבלאות שנוצרו עם הנתונים לאחר הכנסתם. (את הקובץ העלנו גם לגיט)

```
select * from Donation;

select * from Donor;

select * from Person_phone_number;

select * from Paramedic;

select * from Blood;

select * from Receiver;

select * from Donation_place;

select * from Person;
```

## קובץ insertTable -

לאחר שהכנסנו את כל הנתונים בשלושה שיטות השונות, יצרנו את הקובץ insertTables.sql (את הקובץ העלנו לגיט), בקובץ זה כינסנו את כל הפקודות הכנסה של הפרויקט. את הפקודות ההכנסה של הפרויקט כינסנו לאחר שהכנסנו לכל הטבלאות נתונים ועשינו זאת כך:

1. לטבלאות Blood, Receiver ו-Person הכנסנו נתונים באמצעות הרצת קוד בפייתון (insertData.py שהעלנו לגיט) ושימוש בספריית Faker ע"מ להגדיל נתונים רנדומליים. הנתונים של הטבלאות Blood, Receiver ו-Person הוכנסו לתוך קובצי txt ולאחר מכן הוכנסו לטבלאות המתאימות בפרויקט.

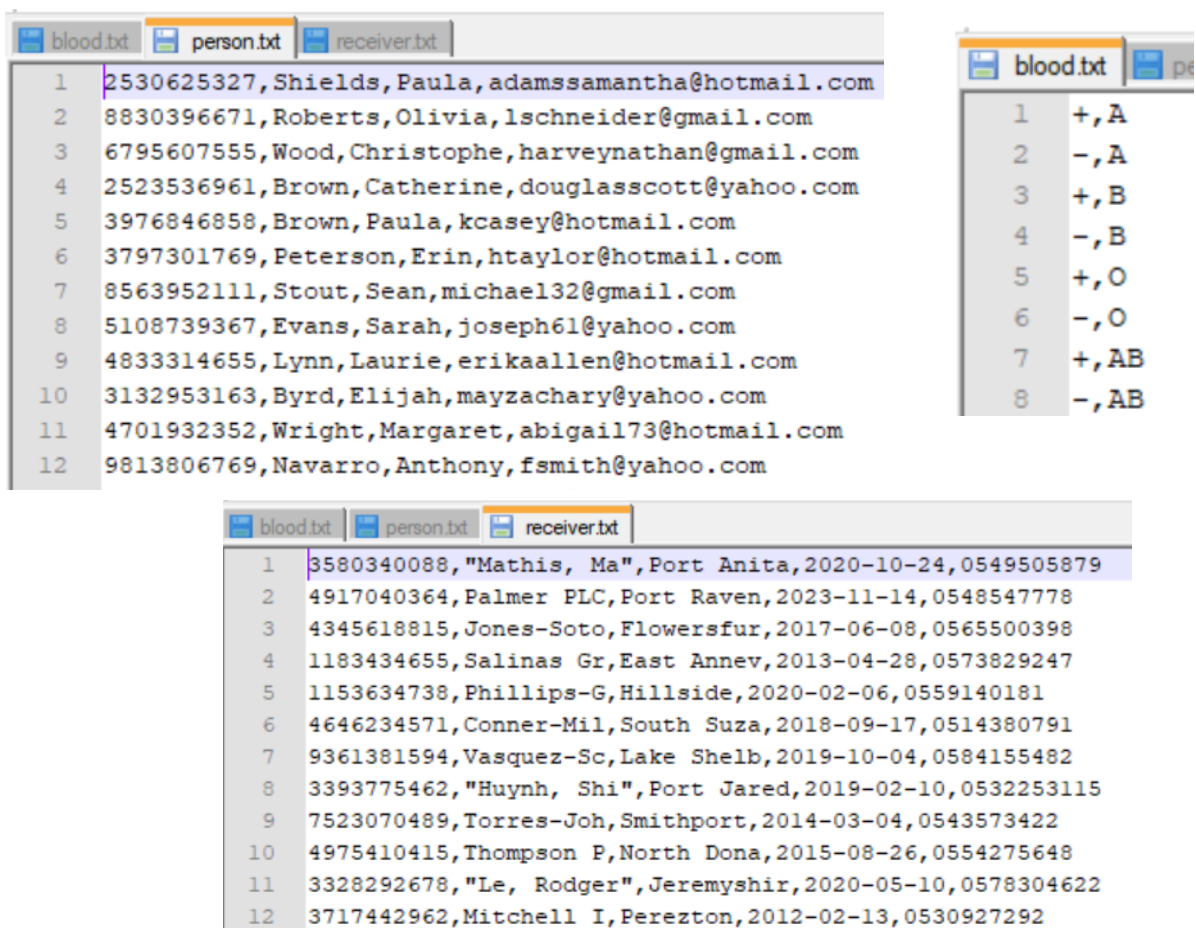
צילום מסך של חלק מהקוד-

```
# Generate 400 rows of random data for Receiver table
receiver_data = [
    {
        'hospital_id': unique_hospital_ids[i],
        'hospital_name': fake.company()[:max_length1],
        'city': fake.city()[:max_length1],
        'delivery_date': fake.date_between(start_date='-13y', end_date='today'),
        'hospital_phone': unique_phone_ids[i]
    }
    for i in range(400)
]

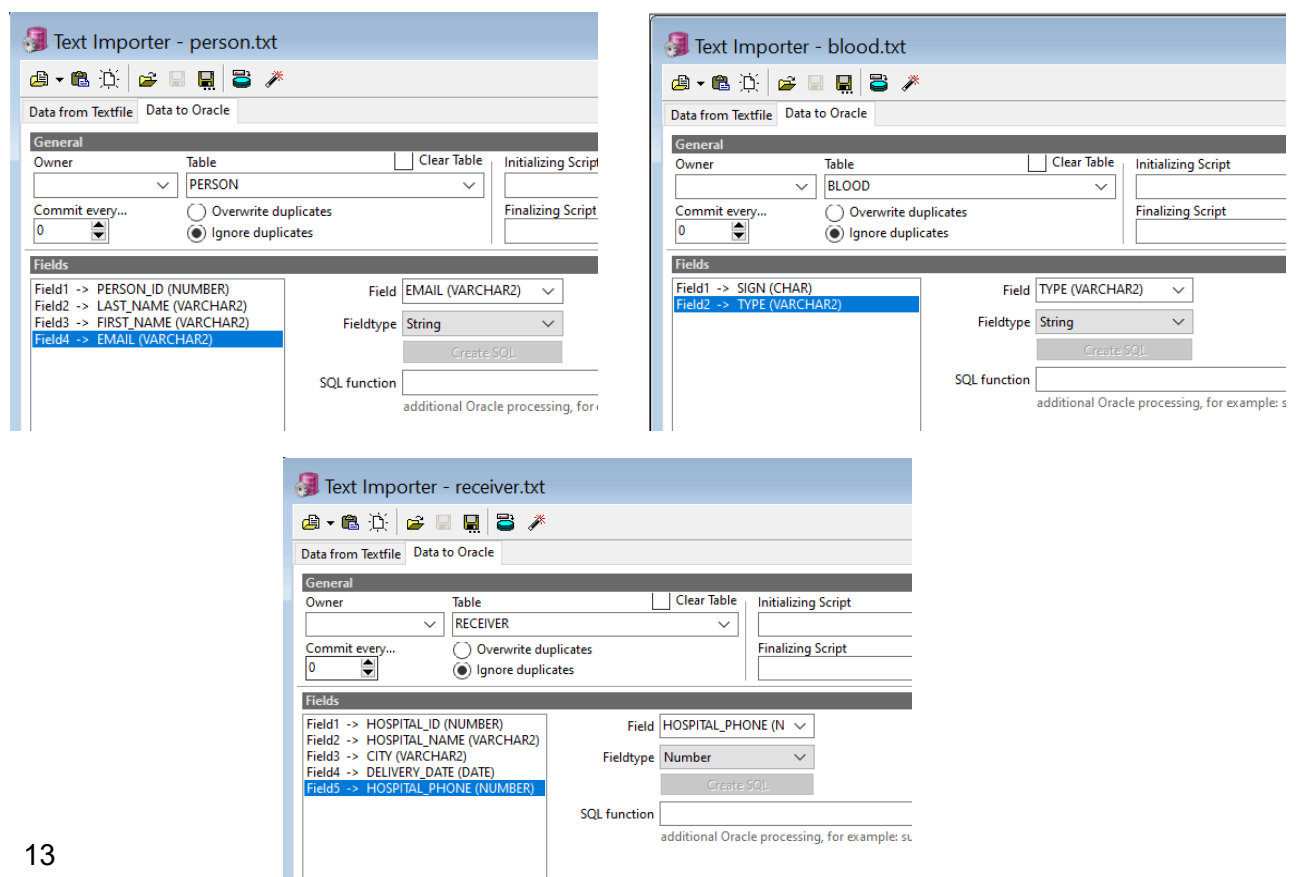
# Convert data to DataFrame
blood_df = pd.DataFrame(blood_data)
person_df = pd.DataFrame(person_data)
receiver_df = pd.DataFrame(receiver_data)

# Write DataFrames to TXT files for Blood and Person
blood_df.to_csv( path_or_buf: 'blood.txt', index=False, sep=',', header=False)
person_df.to_csv( path_or_buf: 'person.txt', index=False, sep=',', header=False)
receiver_df.to_csv( path_or_buf: 'receiver.txt', index=False, sep=',', header=False)
```

## צילום מסך של חלק מקבצי txt שהתקבלו-



## צילום מסך של חלק מתהליך הכנסת הקבצים לטבלאות באורקל-



2. בעזרת Mockaroo יצרנו קבצי SQL עבור הטבלה Donation\_Place ושמרנו והכנסנו גם את הנתונים של הטבלה הזו לפרויקט.  
(צילום מסך מאתר Mockaroo)

The screenshot shows the Mockaroo interface for creating a table named **DONATION\_PLACE**. The table has 400 rows. The fields are configured as follows:

Field Name	Type	Options
place_id	Row Number	blank: 0 %
phone_num	Phone	format: #####-####, blank: 0 %
manager	Full Name	blank: 0 %
address	Street Address	blank: 0 %

Buttons: + ADD ANOTHER FIELD, GENERATE FIELDS USING AI...

Footer: # Rows: 400, Format: SQL, Table Name: DONATION\_PLACE, ☐ include CREATE TABLE

3. לטבלאות Donation, Donor, Person\_phone\_number ו- Paramedic הכנסנו נתונים בעזרת ה Data generator.

The screenshot shows the Data Generator - donor.gd interface. The table **DONOR** is selected, and the number of records is set to 400. The fields are configured as follows:

Name	Type	Size	Data
GENDER	CHAR	1	List('F', 'M')
BIRTH_DATE	DATE		Random(1.1.1960, 1.5.2006)
PERSON_ID	NUMBER	10	List(select person_id from PERSON)
SIGN	CHAR	1	List(select sign from BLOOD)
TYPE	CHAR	2	List(select type from BLOOD)
*			

**Data Generator - paramedic.gd**

PARAMEDIC

Owner: SHOVAL Table: PARAMEDIC Number of records: 400

Name	Type	Size	Data
LICENSE_NUMBER	NUMBER	10	[1111111111]
SENIORITY	NUMBER	2	Random(1, 70)
PERSON_ID	NUMBER	10	List(select person_id from PERSON)
*			

**Data Generator - person\_phone\_number.gd**

PERSON\_PHONE\_NUMBER

Owner: SHOVAL Table: PERSON\_PHONE\_NUMBER Number of records: 600

Name	Type	Size	Data
PHONE_NUMBER	NUMBER	10	'05'+[1111111111]
PERSON_ID	NUMBER	10	List(select person_id from PERSON)
*			

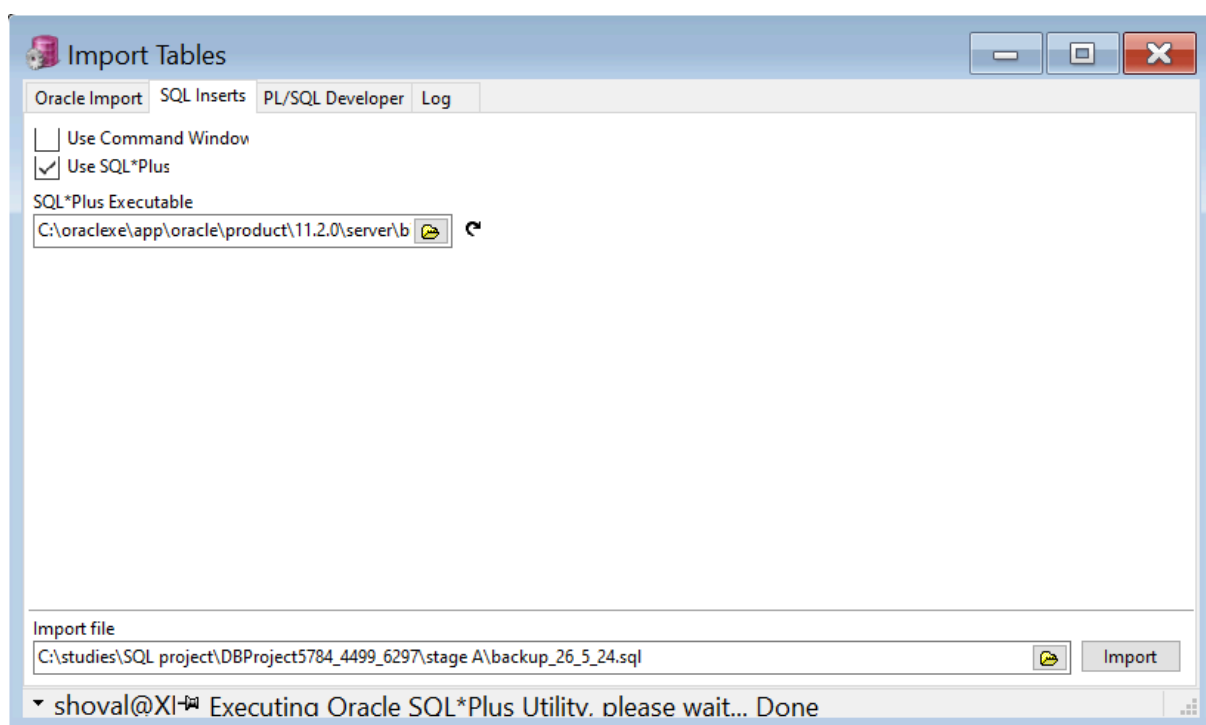
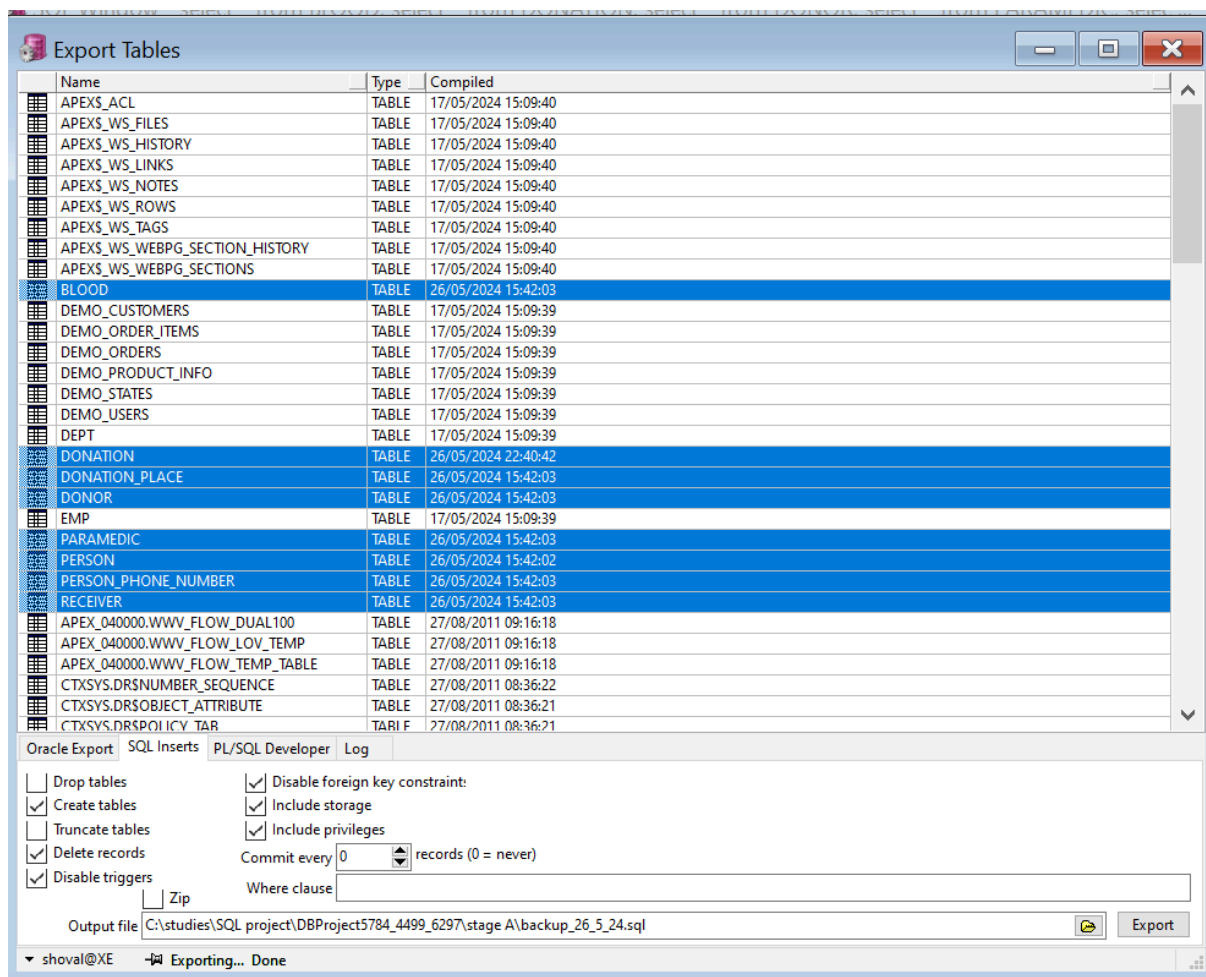
**Data Generator - donation.gd**

DONATION

Owner: SHOVAL Table: DONATION Number of records: 400

Name	Type	Size	Data
DONATION_ID	NUMBER	10	[1111111111]
PASS_TESTS	CHAR	1	List('Y', 'N')
DONATION_DATE	DATE		Random(1.1.2000, 1.5.2024)
VALID	CHAR	1	List('Y', 'N')
PERSON_ID_PARAMED	NUMBER	10	List(select person_id from PARAMEDIC)
PERSON_ID_DONOR	NUMBER	10	List(select person_id from DONOR)
PLACE_ID	NUMBER	10	List(select place_id from DONATION_PLACE)
HOSPITAL_ID	NUMBER	10	List(select hospital_id from RECEIVER)
*			

## גיבוי ושחזור הנתונים:





## שלב ב - שאלות ואילוצים

### שאלות ללא פרמטרים:

(ניתן לראות את השאלות בקובץ Queries.sql)

### • שאלות SELECT:

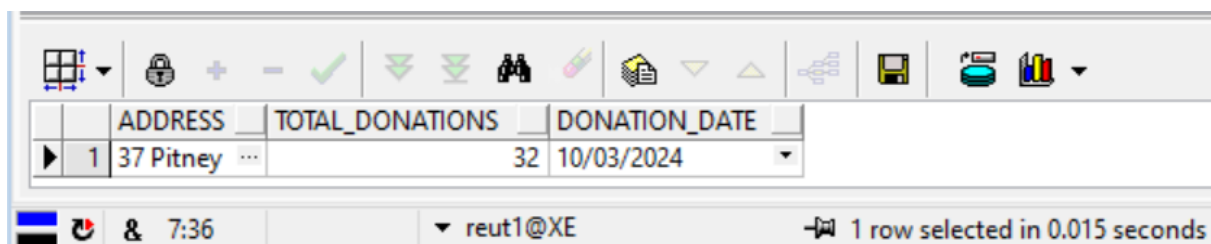
1. רוצים לדעת איפה נערכה ההתרמה הגדולה ביותר ביום יחיד בשנה האחרונה

וכמה תרומות דם נתרמו באותו יום בשביל הענקת פרס מטעם משרד הבריאות.

הקוד-

```
SELECT dp.address, daily_donations.donation_count AS total_donations,
       daily_donations.donation_date
FROM (
    SELECT d.place_id,
           d.donation_date,
           COUNT(*) AS donation_count
    FROM Donation d
    WHERE d.donation_date BETWEEN ADD_MONTHS(SYSDATE, -12) AND SYSDATE
    GROUP BY d.place_id, d.donation_date
    ORDER BY donation_count DESC
) daily_donations
JOIN Donation_place dp ON dp.place_id = daily_donations.place_id
WHERE ROWNUM = 1;
```

התוצאה -



	ADDRESS	TOTAL_DONATIONS	DONATION_DATE
1	37 Pitney ...	32	10/03/2024

1 row selected in 0.015 seconds

## 2. מתרחשת התרמה ב 'Lily Ave 63', מספר תחנה: 69545, ורוצים לשלוח

הודעות אודות ההתרמה לאנשים שתרמו שם בעבר, אך שלא תרמו דם בשלושה חודשים האחרונים, כדי שיבואו לתרום.

הקוד -

```
SELECT P.pid AS person_id, P.f_name AS first_name, P.l_name AS last_name,
       LISTAGG (PPN.phone_number, ', ' ) WITHIN GROUP (ORDER BY
PPN.phone_number) AS phone_numbers
FROM DONATION DN
JOIN DONOR D ON DN.donor_id = D.donor_id
JOIN PERSON P ON D.donor_id = P.pid
JOIN DONATION_PLACE DP ON DN.place_id = DP.place_id
JOIN PERSON_PHONE_NUMBER PPN ON P.pid = PPN.pid
WHERE DP.address = '63 Lily Ave'
AND DP.place_id = 69545
AND D.donor_id NOT IN (
    SELECT D1.donor_id
    FROM DONATION DN1
    JOIN DONOR D1 ON DN1.donor_id = D1.donor_id
    WHERE MONTHS_BETWEEN (SYSDATE, DN1.donation_date) <= 3
)
GROUP BY P.pid, P.f_name, P.l_name;
```

התוצאה -

		PERSON_ID	FIRST_NAME	LAST_NAME	PHONE_NUMBERS
▶	1	004214981	Tracy	Gertner	053-209-8007, 053-209-8007
	2	005988217	Rosanne	Chilton	051-411-2514, 051-411-2514, 051-411-2514, 051-411-2514
	3	007507474	Jena	Horton	050-365-0841, 050-365-0841, 053-664-8631, 053-664-8631
	4	014472246	Jim	Murdock	054-046-8463, 054-046-8463, 054-046-8463, 054-046-8463
	5	025800738	Albertina	Creek	052-921-6102

161 rows selected in 0.061 seconds

3. עבור אדם שבא לתרום, נרצה לבדוק האם לא עברו 3 חודשים מהתרומה האחרונה שלו והאם יש יותר מ-4 תרומות שלו שלא עברו את הבדיקות ולכן הוא לא יכול לתרום (לפי תקנות משרד הבריאות).

הקוד -

```
SELECT P.f_name AS first_name,P.l_name AS last_name,
       D.type AS blood_type,D.sign AS rh_sign,
       (SELECT MAX(DN.donation_date)
        FROM DONATION DN
        WHERE DN.donor_id = P.pid) AS last_donation_date,
       CASE
           WHEN MONTHS_BETWEEN(SYSDATE, (SELECT MAX(DN.donation_date)
                                           FROM DONATION DN
                                           WHERE DN.donor_id = P.pid)) >= 3
           AND (SELECT COUNT(*)
                FROM DONATION DN
                WHERE DN.donor_id = P.pid
                AND DN.pass_tests = 'N') <= 4
       THEN 'Yes'
       ELSE 'No'
       END AS can_donate
FROM PERSON P
JOIN DONOR D ON P.pid = D.donor_id
WHERE P.pid = '238348956';
```

התוצאה -

	FIRST_NAME	LAST_NAME	BLOOD_TYPE	RH_SIGN	LAST_DONATION_DATE	CAN_DONATE
1	Howard	Adams	A	-	19/08/2023	No

1 row selected in 0.031 seconds

4. יש מחסור חמור בכל הארץ בדם מסוג 'O-', רוצים לקבל רשימה של כל האנשים עם סוג הדם הנ"ל, בקבוצות לפי מיקום תרומות קודמות שלהם כדי לשלוח להם בקשה לבוא לתרום, ולהבין איפה הכי כדאי לקיים התרמה לסוג דם הזה.

הקוד -

```
SELECT DP.address AS donation_place,

       LISTAGG(P.f_name || ' ' || P.l_name || ' (' || PPN.phone_number || ')',
', ') WITHIN GROUP (ORDER BY P.l_name) AS donors

FROM DONATION DN

JOIN DONOR D ON DN.donor_id = D.donor_id

JOIN PERSON P ON D.donor_id = P.pid

JOIN PERSON_PHONE_NUMBER PPN ON P.pid = PPN.pid

JOIN DONATION_PLACE DP ON DN.place_id = DP.place_id

WHERE D.sign LIKE '%+%' AND D.type LIKE '%O%'

GROUP BY DP.address;
```

התוצאה -

		DONATION_PLACE	DONORS	
▶	1	1 Kyra Road	Geena Tillis (050-004-3135)	...
	2	10 Albuquerque Ave	Quentin Addy (053-213-5158), Quentin Addy (055-765-3132)	...
	3	11 Breslin Drive	Leslie Akins (054-898-5129)	...
	4	11 Stockard Road	Ethan Roundtree (055-254-2641)	...
	5	13rd Street	Boyd Seviqny (058-862-5971)	...

reut1@XE 34 rows selected in 0.155 seconds

## • שאלות DELETE:

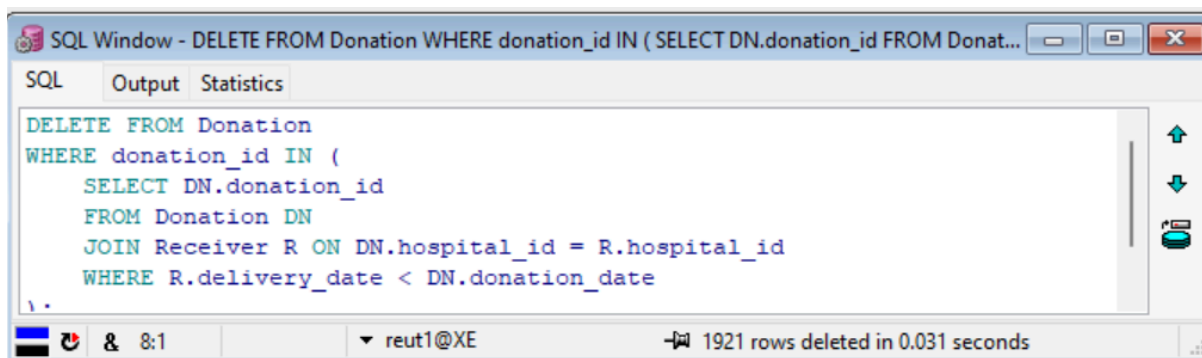
1. רוצים למחוק מהרשומות את כל מנות הדם שרשום שנמסרו לפני שבכלל נלקחו בעקבות תקלה.

הקוד -

```
DELETE FROM Donation
WHERE donation_id IN (
    SELECT DN.donation_id
    FROM Donation DN
    JOIN Receiver R ON DN.hospital_id = R.hospital_id
    WHERE R.delivery_date < DN.donation_date
);
```

כיוון שהיו בהרבה רשומות תקלה, הוספנו נתונים חדשים וחזרנו על פעולת המחיקה.

התוצאה -



ניתן לראות שפעולת המחיקה בוצעה.

2. רוצים למחוק מהרשומות את כל התורמים שגילם גדול מ-64 ולא תרמו בעשר השנים האחרונות ואת כל רשומות התרומות שלהם.

את פעולת המחיקה הזו חילקנו ל 2 פעולות מחיקה שונות.

המחיקה ראשונה היא מחיקת כל רשומות התרומות של תורם מעל גיל 64 שלא תרם ב 10 שנים האחרונות.

הקוד של מחיקה הראשונה -

```
DELETE FROM DONATION
WHERE donor_id IN (
    SELECT D.donor_id
    FROM DONOR D
    JOIN PERSON P ON D.donor_id = P.pid
    WHERE D.birth_date <= ADD_MONTHS(SYSDATE, -64*12)
    AND NOT EXISTS (
        SELECT 1
        FROM DONATION DN
        WHERE DN.donor_id = D.donor_id
        AND DN.donation_date >= ADD_MONTHS(SYSDATE, -120)
    )
);
```

מחיקה שנייה היא מחיקת התורם עצמו שמעל גיל 64 ושלא תרם דם ב 10 שנים האחרונות.

הקוד של המחיקה השנייה -

```
DELETE FROM DONOR
WHERE donor_id IN (
    SELECT D.donor_id
    FROM DONOR D
    JOIN PERSON P ON D.donor_id = P.pid
```

```

WHERE D.birth_date <= ADD_MONTHS(SYSDATE, -64*12)

AND NOT EXISTS (

SELECT 1

FROM DONATION DN

WHERE DN.donor_id = D.donor_id

AND DN.donation_date >= ADD_MONTHS(SYSDATE, -120)

)

);

```

טבלת התורמים מעל גיל 64 לפני המחיקה -

	PERSON_ID	FIRST_NAME	LAST_NAME	BIRTH_DATE
1	349734903	Chaka	Belushi	06/10/1959
2	853599144	Liv	Black	02/12/1959
3	165323833	Leslie	Broadbent	25/08/1953
4	608234327	Ian	Callow	09/12/1950
5	609830431	Lenny	Cronin	20/04/1954

15 rows selected in 0.031 seconds

טבלת התורמים מעל גיל 64 לאחר המחיקה -

	PERSON_ID	FIRST_NAME	LAST_NAME	BIRTH_DATE
1	349734903	Chaka	Belushi	06/10/1959
2	853599144	Liv	Black	02/12/1959
3	165323833	Leslie	Broadbent	25/08/1953

3 rows selected in 0.032 seconds

סה"כ היו 15 תורמים שגילם גדול מ 64 ומתוכם רק 3 המשיכו לתרום ב 10 שנים האחרונות.

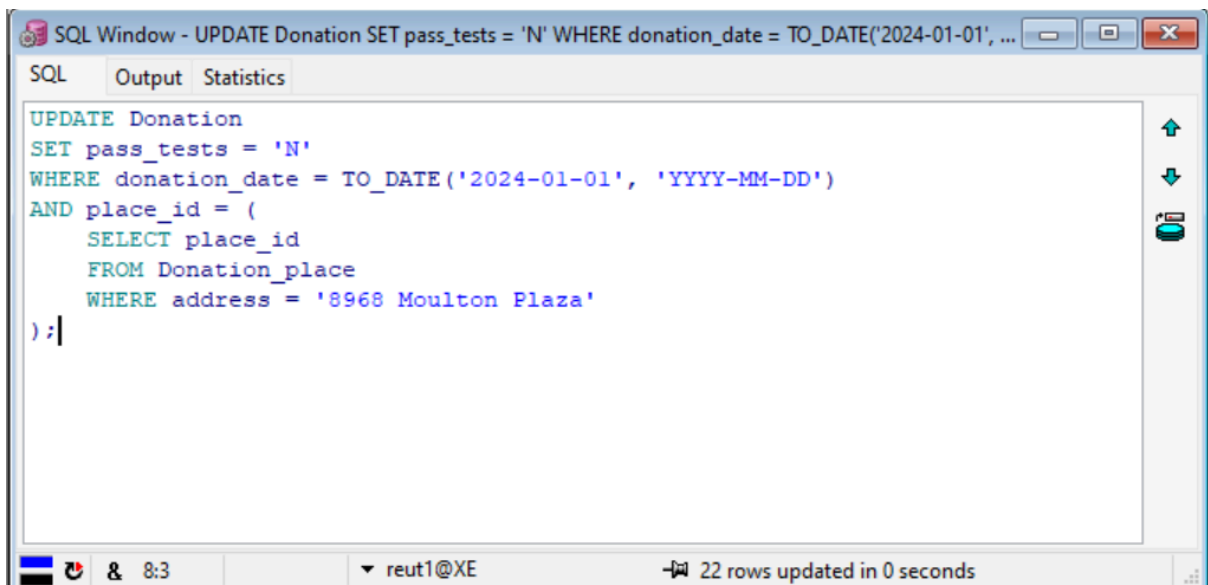
## • שאלות UPDATE:

1. רוצים לסמן שהדם של כל האנשים שתרמו בתחנה מספר 69545 בתאריך 10.03.2024 לא עבר את הבדיקות כיוון שהייתה תקלה במקרר התרומות.

הקוד -

```
UPDATE Donation
SET pass_tests = 'Y', valid = 'N'
WHERE donation_date = TO_DATE('2024-01-01', 'YYYY-MM-DD')
AND place_id = (
    SELECT place_id
    FROM Donation_place
    WHERE address = '8968 Moulton Plaza'
);
```

התוצאה -



ניתן לראות שפעולת העדכון בוצעה.



2. תחנת '82nd Street' רוצה לתת העלאה בשכר לפרמדיקים שערכו אצלה תרומות בשנה האחרונה בשביל לעודד פרמדיקים חדשים לבוא לעבוד אצלם. כל פרמדיק שהוותק שלו גדול משנתיים והשתתף ביותר מ-3 תרומות בתחנה יקבל העלאה של 400 ש"ח בשכר, מי שהשתתף ביותר מ-5 יקבל העלאה של 600 ש"ח.

הקוד -

```
UPDATE PARAMEDIC

SET SALARY = SALARY +

CASE

    WHEN PARAMEDIC_ID IN (

        SELECT D.PARAMEDIC_ID

        FROM DONATION D

        JOIN DONATION_PLACE DP ON D.PLACE_ID = DP.PLACE_ID

        JOIN PARAMEDIC P ON D.PARAMEDIC_ID = P.PARAMEDIC_ID

        WHERE DP.ADDRESS = '82nd Street'

        AND P.SENIORITY > 2

        GROUP BY D.PARAMEDIC_ID

        HAVING COUNT(D.DONATION_ID) > 5

    ) THEN 600

    WHEN PARAMEDIC_ID IN (

        SELECT D.PARAMEDIC_ID

        FROM DONATION D

        JOIN DONATION_PLACE DP ON D.PLACE_ID = DP.PLACE_ID

        JOIN PARAMEDIC P ON D.PARAMEDIC_ID = P.PARAMEDIC_ID

        WHERE DP.ADDRESS = '82nd Street'

        AND P.SENIORITY > 2

        GROUP BY D.PARAMEDIC_ID

        HAVING COUNT(D.DONATION_ID) > 3
```

```

) THEN 400

ELSE 0

END

WHERE PARAMEDIC_ID IN (

SELECT D.PARAMEDIC_ID

FROM DONATION D

JOIN DONATION_PLACE DP ON D.PLACE_ID = DP.PLACE_ID

JOIN PARAMEDIC P ON D.PARAMEDIC_ID = P.PARAMEDIC_ID

WHERE DP.ADDRESS = '82nd Street'

AND P.SENIORITY > 2

GROUP BY D.PARAMEDIC_ID

HAVING COUNT(D.DONATION_ID) > 3

);

```

### תוצאת הרצת הקוד -

6:28	reut1@XE	86 rows updated in 0.766 seconds
------	----------	----------------------------------

ניתן לראות שפעולת העדכון בוצעה.

טבלת הפרמדיקים שהתרימו יותר מ 3 פעמים בתחנה לפני העדכון -

	PARAMEDIC_ID	SENIORITY	NUM_DONATIONS	SALARY
1	479117663	45	4	6800
2	490388060	40	4	12600
3	432174519	24	5	6800
4	922303690	32	5	5700
5	616117735	17	5	10500
6	452597433	33	8	9400
7	347949461	14	4	5700
8	680172578	19	4	7200

טבלת אותם פרמדיקים לאחר העדכון -

	PARAMEDIC_ID	SENIORITY	NUM_DONATIONS	SALARY
1	254115667	43	4	15100
2	479117663	45	4	7200
3	118056743	21	5	7600
4	432174519	24	5	7200
5	184623309	11	4	13000

86 rows selected in 0.062 seconds

ניתן לראות ששכר הפרמדיק עם ת"ז 479117663 גדלה מ-6,800 ל-7,200 כיוון שהוא ביצע יותר מ-3 תרומות בתחנה.

## שאלות עם פרמטרים:

(ניתן לראות את השאלות בקובץ ParamsQueries.sql)

1. כשמתרחשת התרמה רוצים לשלוח הודעות אודות ההתרמה לאנשים שתרמו שם בעבר, אך שלא תרמו דם בשלושה חודשים האחרונים, כדי שיבואו לתרום. לכן, השאלתה מקבלת כתובת של תחנת התרומה, מספר התחנה ותאריך שבו נקבעה ההתרמה ומחזירה את כל הפרטים לאנשים שצריך לשלוח להם הודעות.

הקוד -

```
SELECT P.pid AS person_id,
       P.f_name AS first_name,
       P.l_name AS last_name,
       LISTAGG (PPN.phone_number, ', ') WITHIN GROUP (ORDER BY
PPN.phone_number) AS phone_numbers
FROM DONATION DN
JOIN DONOR D ON DN.donor_id = D.donor_id
JOIN PERSON P ON D.donor_id = P.pid
JOIN DONATION_PLACE DP ON DN.place_id = DP.place_id
JOIN PERSON_PHONE_NUMBER PPN ON P.pid = PPN.pid
WHERE DP.address = &<name="Address" required="true" type="string">
      AND DP.place_id = &<name="Place id" required="true" type="string">
      AND D.donor_id NOT IN (
          SELECT D1.donor_id
          FROM DONATION DN1
          JOIN DONOR D1 ON DN1.donor_id = D1.donor_id
          WHERE MONTHS_BETWEEN(&<name="Donation date" required="true"
type="date">, DN1.donation_date) <= 3
      )
GROUP BY P.pid, P.f_name, P.l_name;
```

פרמטרים -

Variables

Name	Value
Address	63 Lily Ave
Place id	69545
Donation date	03/06/2024

OK Cancel Clear

התוצאות -

		PERSON_ID	FIRST_NAME	LAST_NAME	PHONE_NUMBERS
▶	1	004214981	Tracy	Gertner	053-209-8007, 053-209-8007
	2	005988217	Rosanne	Chilton	051-411-2514, 051-411-2514, 051-411-2514, 051-411-2514
	3	007507474	Jena	Horton	050-365-0841, 050-365-0841, 053-664-8631, 053-664-8631
	4	014472246	Jim	Murdock	054-046-8463, 054-046-8463, 054-046-8463, 054-046-8463
	5	025800738	Albertina	Creek	052-921-6102

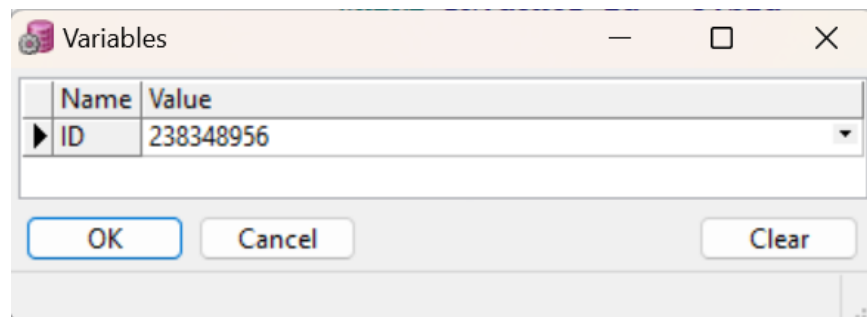
155 rows selected in 0.266 seconds

2. עבור אדם שבא לתרום, נרצה לבדוק האם לא עברו 3 חודשים מהתרומה האחרונה שלו והאם יש יותר מ-4 תרומות שלו שלא עברו את הבדיקות ולכן הוא לא יכול לתרום (לפי תקנות משרד הבריאות).

הקוד -

```
SELECT P.f_name AS first_name,P.l_name AS last_name,
       D.type AS blood_type,D.sign AS rh_sign,
       (SELECT MAX(DN.donation_date)
        FROM DONATION DN
        WHERE DN.donor_id = P.pid) AS last_donation_date,
CASE
    WHEN MONTHS_BETWEEN(SYSDATE, (SELECT MAX(DN.donation_date)
                                     FROM DONATION DN
                                     WHERE DN.donor_id = P.pid)) >= 3
    AND (SELECT COUNT(*)
          FROM DONATION DN
          WHERE DN.donor_id = P.pid
          AND DN.pass_tests = 'N') <= 4
    THEN 'Yes'
    ELSE 'No'
END AS can_donate
FROM PERSON P
JOIN DONOR D ON P.pid = D.donor_id
WHERE P.pid = &<name="ID" type="string" required="true">;
```

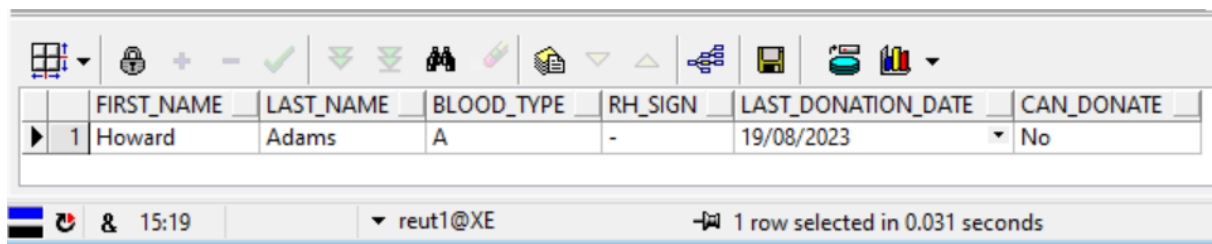
פרמטרים -



Name	Value
ID	238348956

OK Cancel Clear

התוצאות -



	FIRST_NAME	LAST_NAME	BLOOD_TYPE	RH_SIGN	LAST_DONATION_DATE	CAN_DONATE
1	Howard	Adams	A	-	19/08/2023	No

15:19 reut1@XE 1 row selected in 0.031 seconds

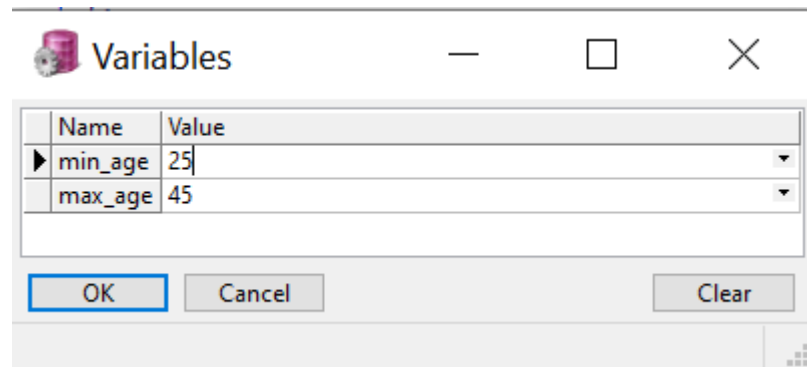
3. בית החולים מבקש מבנק הדם שמות ומספרי טלפון של תורמים אשר בעלי סוג דם "O-", בטווח גילאים מסויים (פרמטרים) ושתרמו לפחות תרומה אחת שעברה את הבדיקות בשנה האחרונה.

הקוד -

```
SELECT  p.f_name,  p.l_name,  pp.phone_number
FROM    person p
        JOIN donor d ON p.pid = d.donor_id
        JOIN person_phone_number pp ON p.pid = pp.pid
WHERE   d.type = 'O' AND d.sign= '-'
        AND ROUND(MONTHS_BETWEEN(SYSDATE, d.birth_date) / 12) BETWEEN
        &min_age AND &max_age
        AND EXISTS (SELECT 1
                     FROM    donation dn
                     WHERE   dn.donor_id = d.donor_id
                             AND dn.donation_date >= TRUNC(SYSDATE) - INTERVAL
                             '1' YEAR
                             AND dn.pass_tests = 'Y');
```



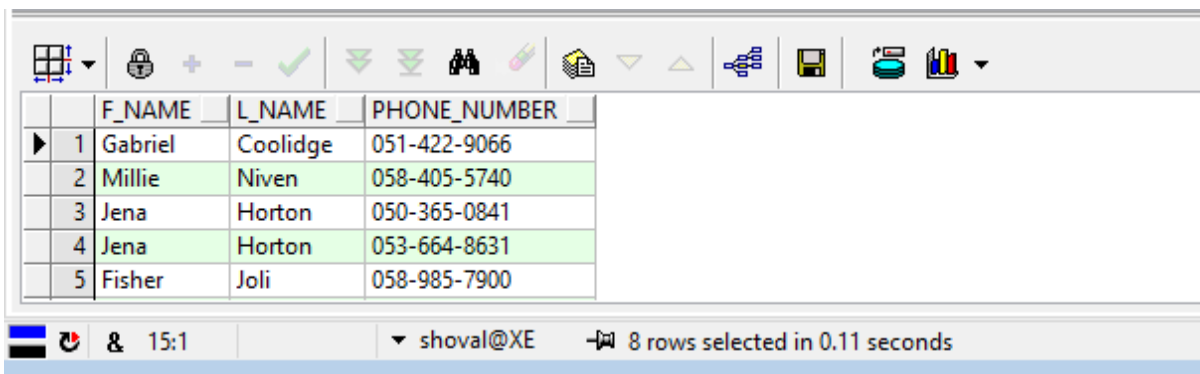
פרמטרים -



The Variables dialog box contains a table with two rows. The first row has 'min\_age' in the Name column and '25' in the Value column. The second row has 'max\_age' in the Name column and '45' in the Value column. Below the table are three buttons: 'OK', 'Cancel', and 'Clear'.

Name	Value
min_age	25
max_age	45

התוצאות -



The Query Results window displays a table with 5 rows of data. The columns are F\_NAME, L\_NAME, and PHONE\_NUMBER. The first row is highlighted in green. The status bar at the bottom indicates that 8 rows were selected in 0.11 seconds.

	F_NAME	L_NAME	PHONE_NUMBER
1	Gabriel	Coolidge	051-422-9066
2	Millie	Niven	058-405-5740
3	Jena	Horton	050-365-0841
4	Jena	Horton	053-664-8631
5	Fisher	Joli	058-985-7900

4. לאחר המלחמה מד"א התעניינו לתגמל פרמדיקים שעבדו בזמן המלחמה ולתגמל בכך שיתנו פרס לפרמדיקים שהתרימו באותו חודש.


הקוד -

```
SELECT p.f_name,
       p.l_name,
       p.pid AS paramedic_id,
       COUNT(d.donation_id) AS donation_count
FROM   person p
       JOIN paramedic pr ON p.pid = pr.paramedic_id
       JOIN donation d ON pr.paramedic_id = d.paramedic_id
WHERE  EXTRACT(MONTH FROM d.donation_date) = EXTRACT(MONTH FROM
TO_DATE(&target_month, 'YYYY-MM'))

      AND EXTRACT(YEAR FROM d.donation_date) = EXTRACT(YEAR FROM
TO_DATE(&target_month, 'YYYY-MM'))

GROUP BY p.f_name, p.l_name, p.pid
HAVING COUNT(d.donation_id) > 0
ORDER BY donation_count DESC;
```

פרמטרים -

 Variables

Name	Value
▶ target_month	'2023-10'

OK Cancel Clear

התוצאות -

		F_NAME	L_NAME	PARAMEDIC_ID	DONATION_COUNT
▶	1	Reese	Cage	984430684	1
	2	Susan	Warwick	915523925	1
	3	Roger	Cronin	766797561	1
	4	Walter	Liotta	716522491	1
	5	Tara	Sewell	898034681	1
	6	Earl	Ingram	837205710	1
	7	Julianna	Silverman	475834164	1

< [ ]

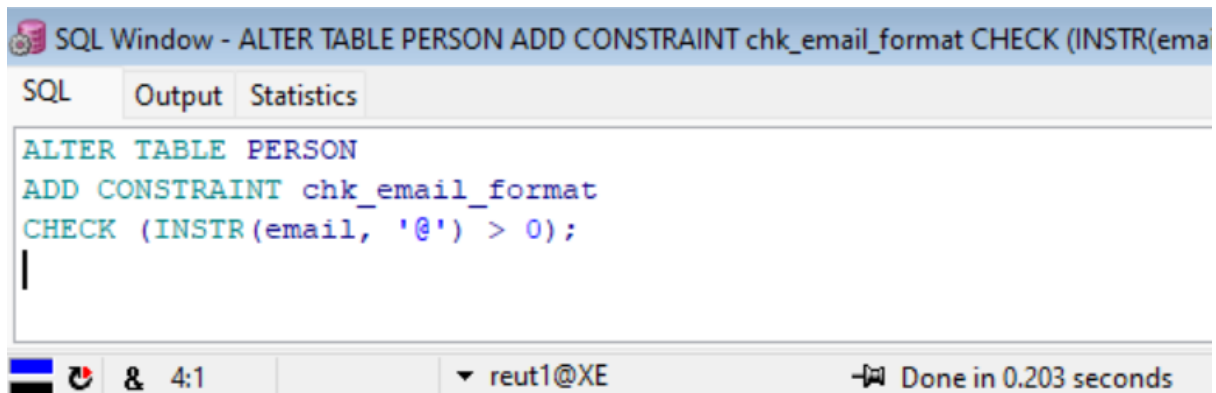
13:1 shoval@XE 17 rows selected in 0.047 seconds (more...)

## אילוצים:

(ניתן לראות את השאילתות בקובץ Constraints.sql)

## 1. אילוץ על טבלת PERSON -

כאשר מכניסים כתובת אימייל לטבלה של PERSON, ייבדק שהיא כוללת '@'.



SQL Window - ALTER TABLE PERSON ADD CONSTRAINT chk\_email\_format CHECK (INSTR(email, '@') > 0);

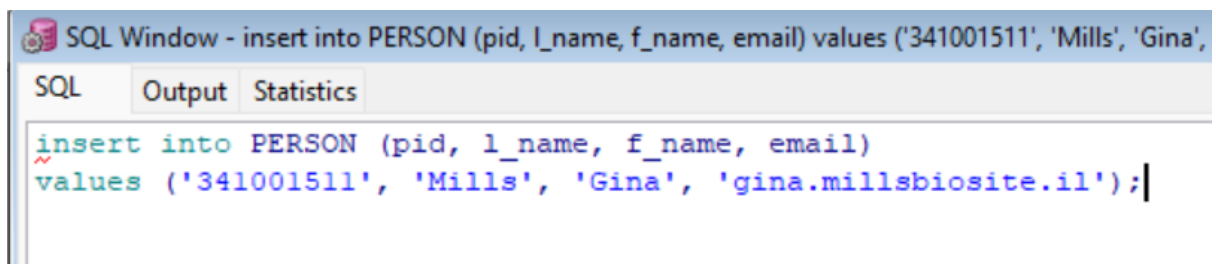
```

ALTER TABLE PERSON
ADD CONSTRAINT chk_email_format
CHECK (INSTR(email, '@') > 0);

```

Done in 0.203 seconds

בדיקה -



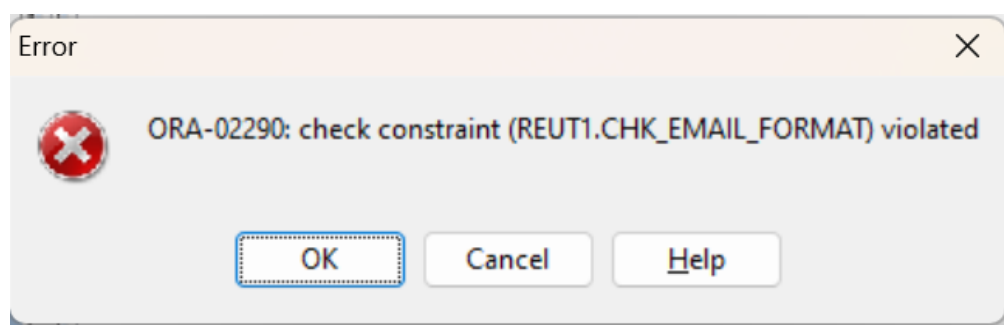
SQL Window - insert into PERSON (pid, l\_name, f\_name, email) values ('341001511', 'Mills', 'Gina', 'gina.millsbiosite.il');

```

insert into PERSON (pid, l_name, f_name, email)
values ('341001511', 'Mills', 'Gina', 'gina.millsbiosite.il');

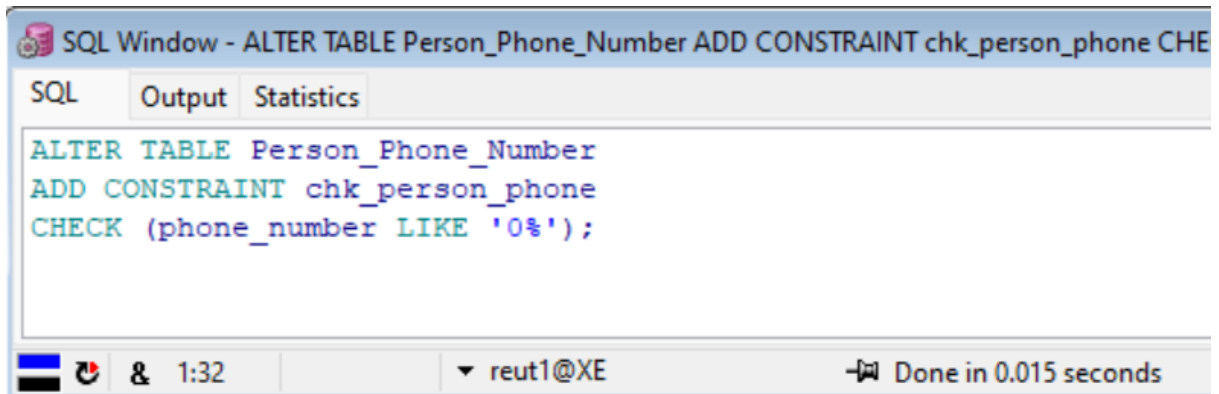
```

התקבלה שגיאה -



## 2. אילויז על טבלת PERSON\_PHONE\_NUMBER -

כאשר מכניסים מספר טלפון לטבלה של PERSON\_PHONE\_NUMBER, ייבדק שהוא מתחיל ב-0.



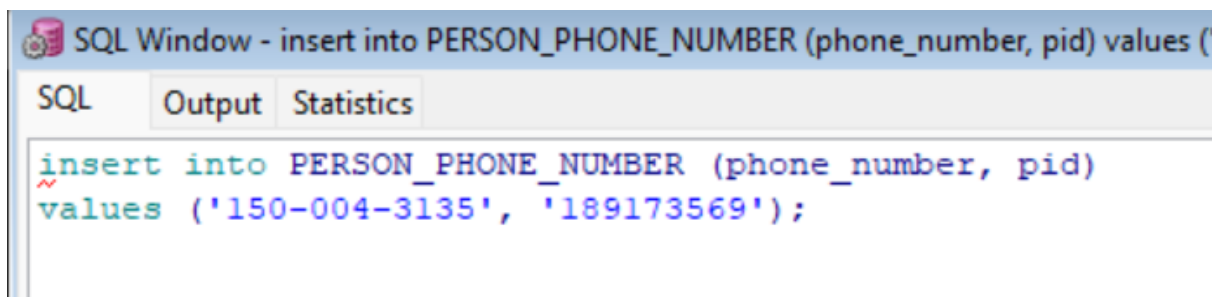
SQL Window - ALTER TABLE Person\_Phone\_Number ADD CONSTRAINT chk\_person\_phone CHE

SQL Output Statistics

```
ALTER TABLE Person_Phone_Number
ADD CONSTRAINT chk_person_phone
CHECK (phone_number LIKE '0%');
```

1:32 reut1@XE Done in 0.015 seconds

בדיקה -

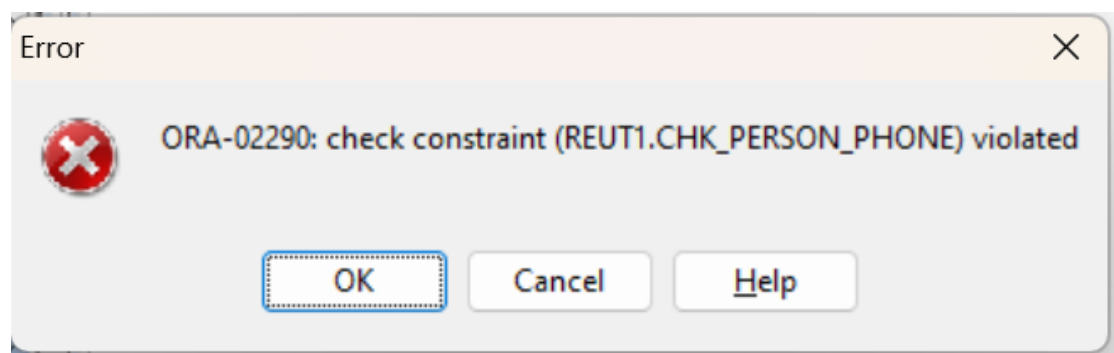


SQL Window - insert into PERSON\_PHONE\_NUMBER (phone\_number, pid) values (

SQL Output Statistics

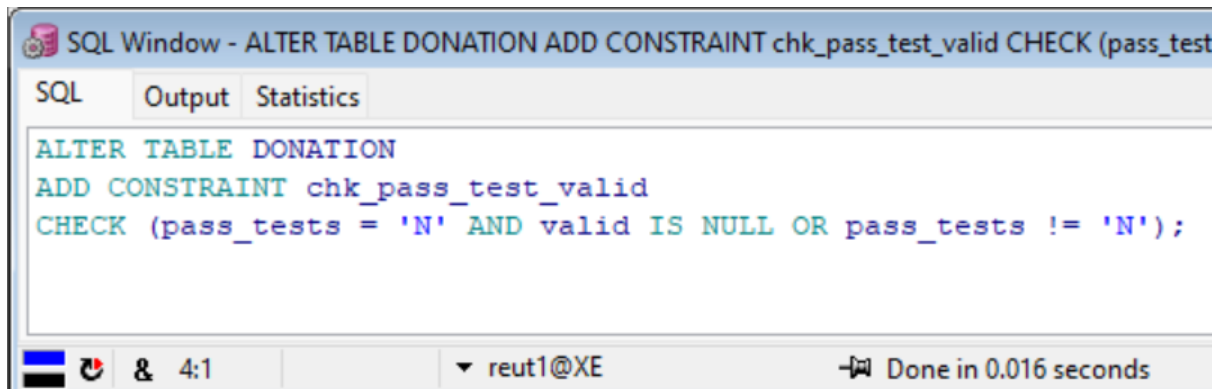
```
insert into PERSON_PHONE_NUMBER (phone_number, pid)
values ('150-004-3135', '189173569');
```

התקבלה שגיאה -



## 3. אילון על טבלת DONATION -

כאשר מכניסים תרומה לטבלה של DONATION, אם היא עוד לא עברה בדיקות, צריך שהשדה valid יהיה שווה ל NULL.

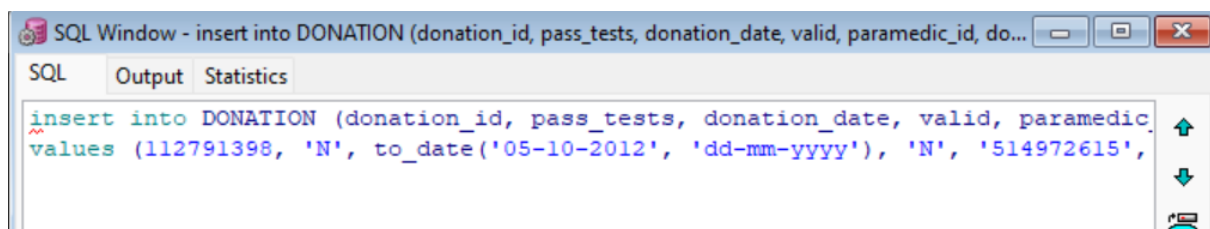


SQL Window - ALTER TABLE DONATION ADD CONSTRAINT chk\_pass\_test\_valid CHECK (pass\_test

```
SQL
ALTER TABLE DONATION
ADD CONSTRAINT chk_pass_test_valid
CHECK (pass_tests = 'N' AND valid IS NULL OR pass_tests != 'N');
```

Done in 0.016 seconds

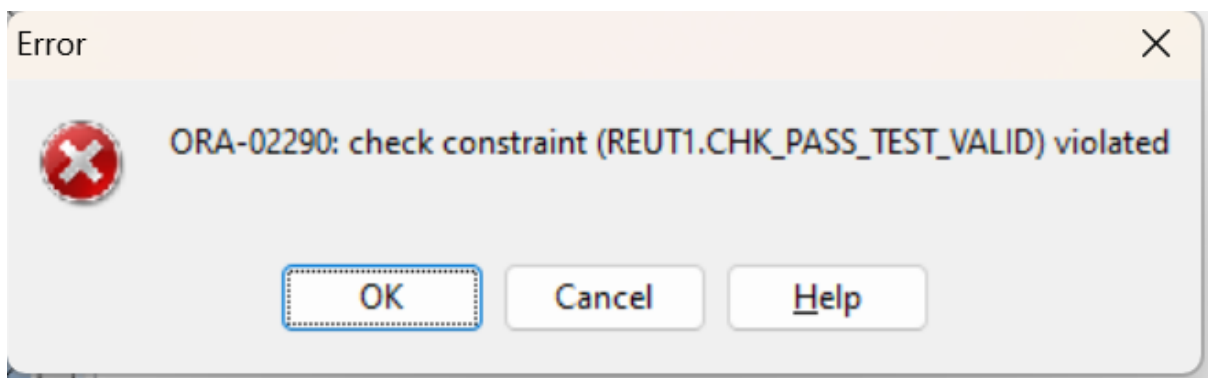
בדיקה -



SQL Window - insert into DONATION (donation\_id, pass\_tests, donation\_date, valid, paramedic\_id, do...

```
SQL
insert into DONATION (donation_id, pass_tests, donation_date, valid, paramedic_id, do...
values (112791398, 'N', to_date('05-10-2012', 'dd-mm-yyyy'), 'N', '514972615',
```

התקבלה שגיאה -



## שינויים:

(ניתן גם לראות בקובץ AlterTables.sql)

חוזן מהשינויים לאילוצים עשינו רק שינוי אחד שהוא הוספה לטבלת ה PARAMEDIC ערך חדש של שכר.

הקוד -

```
ALTER TABLE Paramedic
ADD salary NUMERIC(6) DEFAULT 0;
```

הטבלה לאחר הפקודה -

Type owner

Name

	Name	Virtual	Type	Nullable	Default/Expr.	Storage	Comments
▶	LICENSE_NUM	<input type="checkbox"/>	NUMBER(5) ▼	<input type="checkbox"/>	...		
	SENIORITY	<input type="checkbox"/>	NUMBER(2) ▼	<input checked="" type="checkbox"/>	0	...	
	PARAMEDIC_ID	<input type="checkbox"/>	VARCHAR2(9) ▼	<input type="checkbox"/>	...		
	SALARY	<input type="checkbox"/>	NUMBER(6) ▼	<input checked="" type="checkbox"/>	0	...	
*		<input checked="" type="checkbox"/>	▼	<input checked="" type="checkbox"/>	...	...	