

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
6.....	טבלאות DSD
7-9.....	תוכן הקובץ createTable
10.....	פעולת describe
11.....	קובץ dropTable
11.....	קובץ selectAll
12-15.....	קובץ insertTable
16	גיבוי ושחזור הנתונים

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:

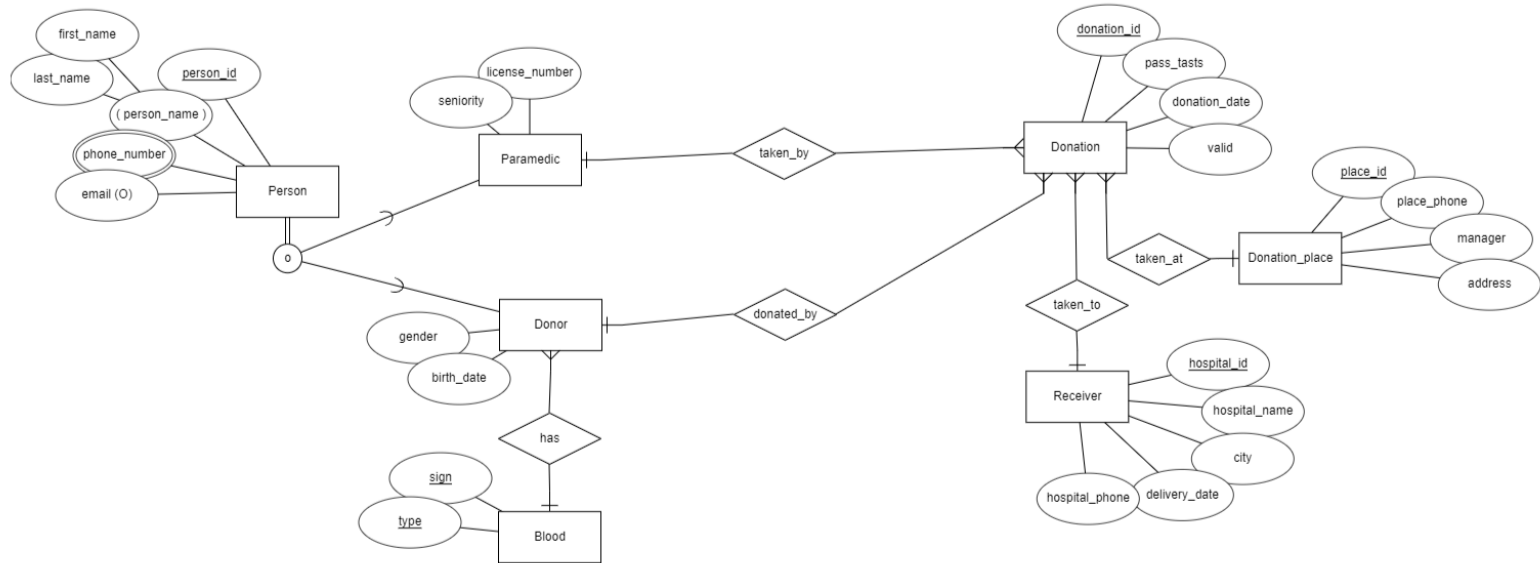
ישויות:

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)



קובץ createTable

(את הקובץ createTables.sql 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)
);

```

פעולה describe

ניתן לראות שהטבלאות נוצרו ע"י שימוש בפעולה describe:

Tables

- APEXS_ACL
- APEXS_WS_FILES
- APEXS_WS_HISTORY
- APEXS_WS_LINKS
- APEXS_WS_NOTES
- APEXS_WS_ROWS
- APEXS_WS_TAGS
- APEXS_WS_WEBPG_SECTION_HISTORY
- APEXS_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.

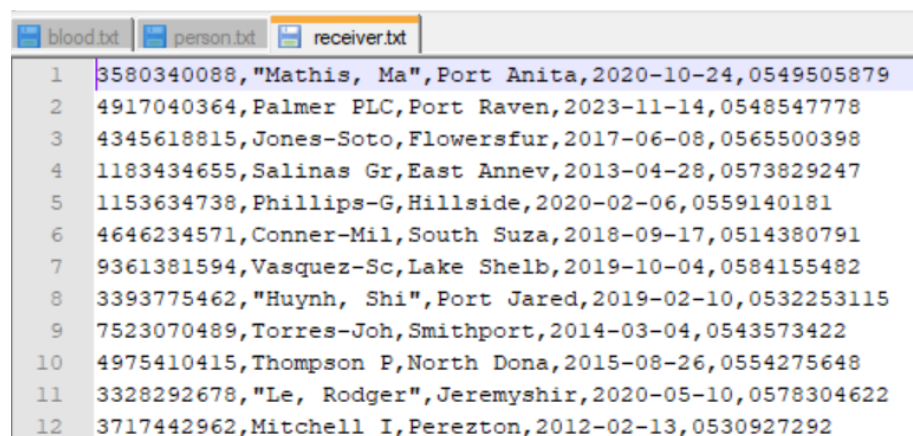
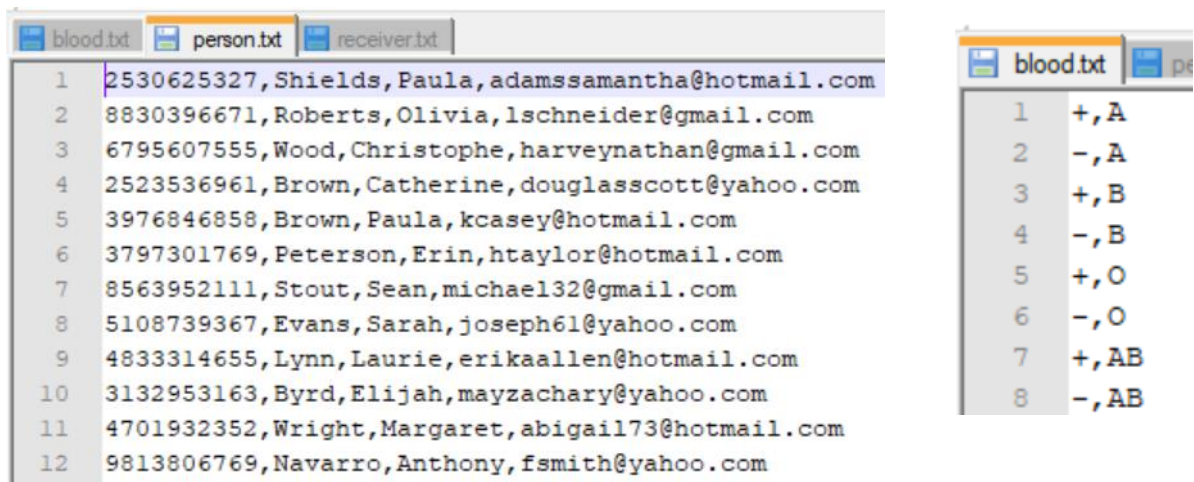
צילום מסך של חלק מהקוד-

```
# 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)
]

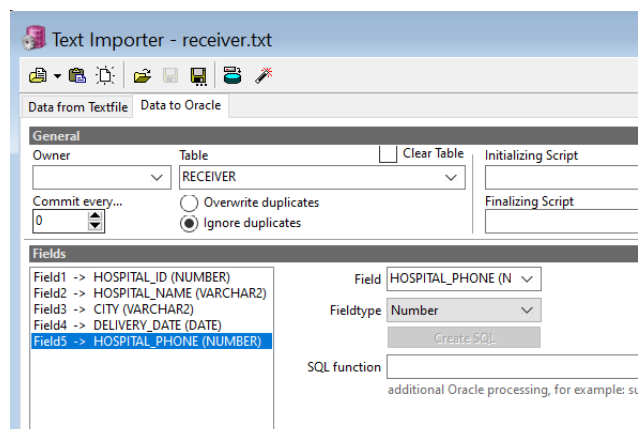
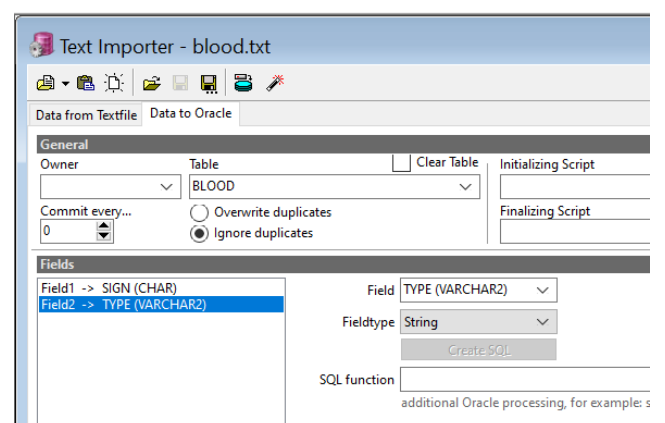
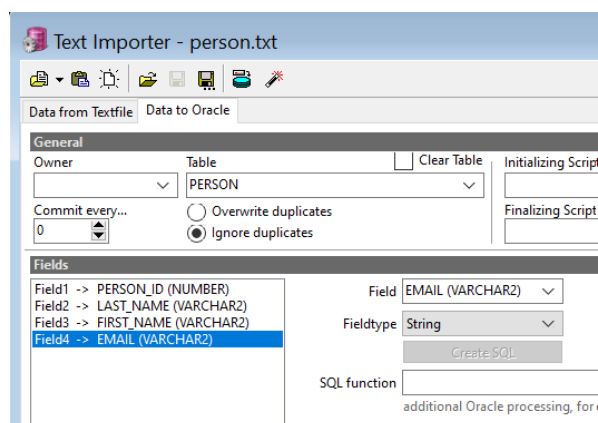
# Convert data to DataFrame
blood_df = pd.DataFrame(blood_data)
person_df = pd.DataFrame(person_data)
receiver_df = pd.DataFrame(receiver_data)

# 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 configuration interface. It lists four fields: place_id (Row Number), phone_num (Phone), manager (Full Name), and address (Street Address). Each field has a 'blank' percentage set to 0% and a 'Σ' icon. At the bottom, the configuration is set for 400 rows, SQL format, and the table name is DONATION_PLACE. There is an option to 'include CREATE TABLE' which is currently unchecked.

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

+ ADD ANOTHER FIELD GENERATE FIELDS USING AI...

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. It is configured for the 'DONOR' table with 400 records. The fields and their configurations are 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	[111111111]
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'+[111111111]
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	[111111111]
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 ושיחזור הנתונים -

