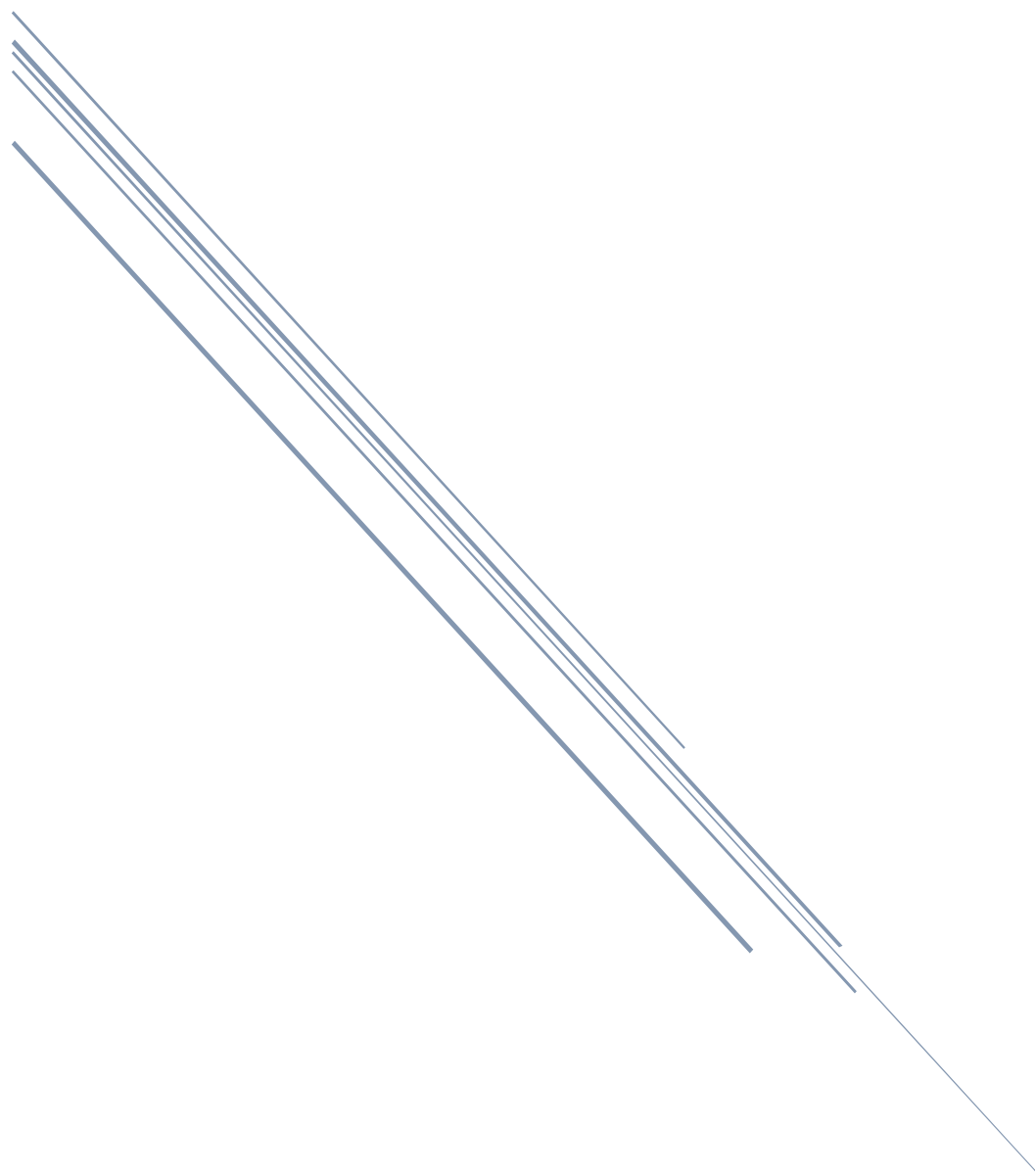


# מיני פרויקט בבסיסי נתונים

מערכת: צבא ההגנה לישראל | יחידה: אגף האפסנאות

גיטהב: <https://github.com/udolev/Mini-Project-in-DB>



אוריאל דולב 215676560  
שילה סופיר 328308002

## תוכן עניינים

1	תוכן עניינים
2	שלב 1
2	1.1 תיאור המערכת
3	1.2 ERD
3	1.3 דיאגרמת DSD
4	1.3.1 החלטות עיצוביות
4	1.3.1 תיאור הישויות
5	1.3.2 תיאור הטבלאות
5	1.3.3 נרמול ל 3NF
6	1.3.4 תיאור הקשרים
7	1.4 יצירת הטבלאות (SQL)
8	1.5 מחיקת הטבלאות (SQL)
8	1.6 SELECT ALL (SQL)
9	1.7 יצירת הנתונים
9	דרך א – mockaroo (דרך עיקרית):
14	דרך ב – פייתון:
15	דרך ג - data generator:
17	1.8 גיבוי ושחזור הנתונים

## שלב 1

### 1.1 תיאור המערכת

בפרוייקט אנחנו נממש את ניהול בסיס הנתונים של מערך האפסנאות בצהל.

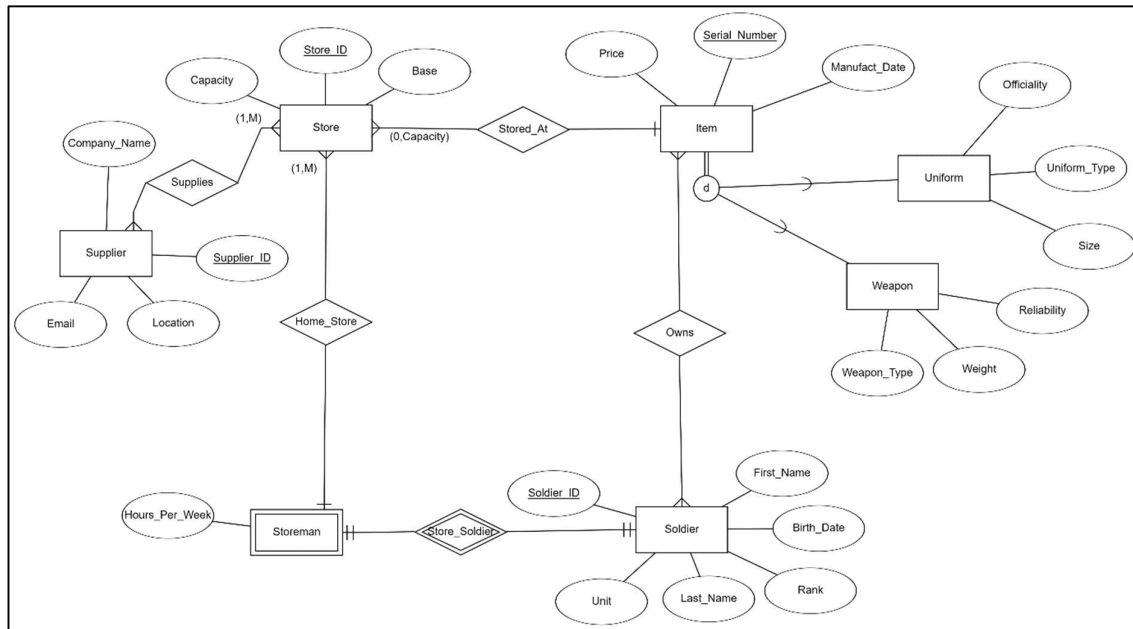
תפיקו העיקרי של מערך האפסנאות הוא להקצות לכל חייל את הציוד שהוא צריך ועבור פריטים שלא בשימוש למצוא להם מחסן לשמור אותם. לצורך כך קיימים מספר מחסנים בצהל כאשר יכולים להיות כמה מחסנים בכל בסיס או אף מחסן. כל מחסן יהיה ייחודי לפי מספר סידורי מיוחד שיש לו. לכל מחסן יהיה ספק אחד או יותר שיספק לו את הציוד.

כל פריט יחולק לשתי קטגוריות. פריט מסוג "מדים" – חולצות, מכנסיים, נעליים... ופריט מסוג "נשק" – M16, רימונים, פגזים, מקלעים... לכל פריט יהיה מספר סידורי ייחודי לצור זיהוי ועוד תכונות נלוות כגון מידה אם זה פריט לבוש או משקל אם זה סוג נשק. כך שנוכל למיין את הפריטים לפי מחסנים לדוגמא אם נרצה להקצות מחסן מסויים רק לפרטי לבוש נוכל לבחור רק פריטים מסוג "מדים". מספר הפריטים בכל מחסן יוגבל על פי תכולת הפריטים המקסימלית של המחסן.

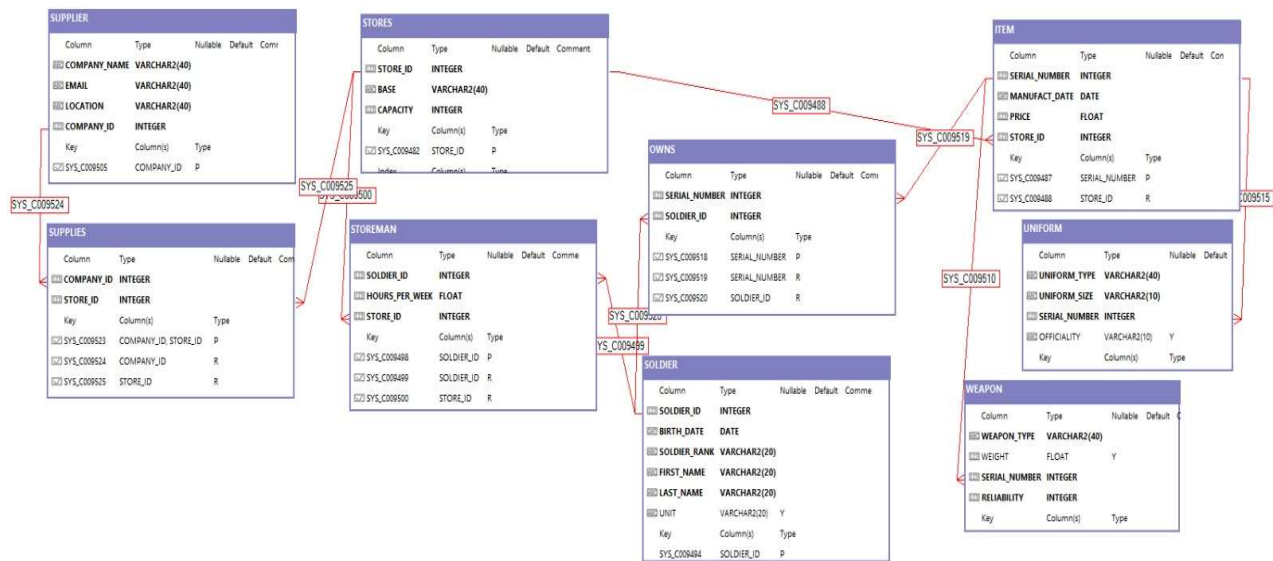
לכל חייל יהיו מספר פריטים השייכים לו. כך שפריט ששייך לחייל כלומר נמצא בשימוש לא יופיע באף מחסן.

עובד מחסן הוא חייל מסוג עובד מחסן שיכיל תכונה נוספת של מספר שעות עבודה בשבוע. כל עובד יוכל לעבוד במחסן אחד בדיוק אך כמובן שלכל מחסן יוכלו להיות יותר מעובד אחד.

## ERD 1.2



## 1.3 דיאגרמת DSD



### 1.3.1 החלטות עיצוביות

ראשית בנינו את הישויות מחסן, ספק, פריט וחייל שהן הישויות העומדות בבסיס כל המערכת. כי למעשה המערכת שלנו מנסה לנהל את הקשר בין הפריטים שיש לצהל למקומות בהם הם מאוכסנים ולחיילים להם הם שייכים.

לאחר מכן החלטנו שאת היישות פריט (Item) נפצל לשני יישויות (בעזרת ירושה). Uniform Weapon. החלטה זאת נובעת מהשוני המהותי בין סוגי הפריטים ששמורים במחסנים. יש סוג אחד שהוא נשקים ויש לו תכונות משלו ויש סוג שונה שהוא המדים שלו יש תכונות משלו כמו למשל מידה (Size) תכונה שלא תתאים לנשק.

לאחר מכן החלטנו שכל חפץ ששייך למחסן מסויים יכול להיות גם כרגע שייך לחייל ספציפי. אד כדי לדעת לאיפה החייל יצטרך להחזיר בסוף את הפריט ייצרנו קשר Owns ששומר עבור כל פריט ששייך לחייל מסויים לאיזה חייל הוא שייך.

לבסוף הוספנו יישות שתייצג לנו את החיילים שעובדים במחסנים כדי שיהיה נוח לנהל את כוח העבודה במחסנים. כל "עובד מחסן" הוא חייל ולכן עובד מחסן זו יישות חלשה שהמפתח שלה נלקח מ"חייל".

### 1.3.1 תיאור הישויות

**ספק (Supplier)** – מתאפיין במספר החברה הייחודי לכל חברה, שם החברה, מייל ומיקום של משרדי החברה.

**מחסן (Store)** – מחסן מתאפיין במספר מחסן (מפתח ראשי), תכולה מקסימלית במחסן ומספר הבסיס בו הוא נמצא.

**חייל (Soldier)** – לכל חייל יש את התכונות הבאות: שם פרטי, שם משפחה, תאריך לידה, דרגה, יחידה בה הוא משרת ומספר החייל (מפתח ראשי).

**Storman (אפסנאי)** – לאפסנאי יש את התכונות הבאות: מספר הזהות שלו כחייל כמפתח זר וגם כמפתח ראשי, מספר המחסן בו הוא עובד ומספר שעות העבודה השבועיות שלו.

**פריט (Item)** – כל פריט מתאפיין בתכונות הכלליות שהן מחיר, תאריך ייצור ומספר סידורי (מפתח ראשי).

**מדים (Uniform)** – פריט מסוג מדים מכיל את המידה של הפריט, את סוג הפריט (חולצה\מכנסיים\נעליים וכו') ואת רשמיות המדים (מדי א' מדי ב').

**נשק (Weapon)** – פריט מסוג נשק מכיל את סוג הנשק, משקל הנשק, אמינות של הנשק (מס' מ' עד 10 כאשר 1 מסמל אמינות נמוכה ו10 מסמל אמינות גבוהה).

### 1.3.2 תיאור הטבלאות

**מקרא:** Foreign key, Primer key, Attribute

ספק (מספר זיהוי חברה, שם חברה, מייל, מיקום)

מחסן (מספר מחסן, בסיס, תכולה מקסימלית)

מסופק על ידי (מספר מחסן, מספר חברה)

עובד מחסן (מספר מחסן, מספר חייל, שעות עבודה שבועיות)

חייל (תז חייל, שם פרטית שם משפחה, תאריך לידה, דרגה, שם יחידה)

Own (מספר סידורי, תז חייל)

פריט (מספר סידורי, מחיר, תאריך ייצור, מספר מחסן)

מדים (מספר סידורי, סוג מדים, מידה, רשמיות)

נשק (מספר סידורי, סוג נשק, משקל, אמינות)

### 1.3.3 נרמול ל-3NF

כל הסכמות מנורמלות ל-3NF מכיוון שרק הגדרנו אילו תכונות יהיו מפתחות ולא הוספנו תלויות מעבר. לכן לא יכול להיווצר מצב של סכמה שאינה עומדת ב-3NF.

### 1.3.4 תיאור הקשרים

#### ספק - מחסן: Many to Many

ספק – כל ספק יכול לספק למספר מחסנים או לאפס.

מחסן – כל מחסן מחובר למספר ספקים או לאפס.

#### מחסן – פריט: One to Many

מחסן - מכיל מספר פריטים או אפס.

פריט – כל פריט נמצא מקסימום במחסן אחד.

#### פריט – נשק/מדים: inheritance

כל פריט הוא או מסוג נשק או מסוג מדים.

#### פריט – חייל: Many to One

פריט – כל פריט מחובר למקסימום חייל אחד (או שהוא לא מחובר בכלל כלומר נשאר במחסן).

חייל – לכל חייל יש מספר פריטים או אף פריט.

#### עובד מחסן – חייל: ישות חלש, One to One

חייל – כל חייל יכול להיות או חייל רגיל או עובד מחסן.

עובד מחסן – כל עובד מחסן הוא חייל ולכן הוא מקבל מחייל את המפתח שלו (id).

#### עובד מחסן – מחסן: One to Many

עובד מחסן – כל עובד מחסן עובד בדיוק במחסן אחד (חץ עגול).

מחסן – בכל מחסן יכולים לעבוד כמה עובדים.

## 1.4 יצירת הטבלאות (SQL)

```

CREATE TABLE Stores (
    Store_ID INT NOT NULL,
    Base VARCHAR2(40) NOT NULL,
    Capacity INT NOT NULL,
    PRIMARY KEY (Store_ID)
);

CREATE TABLE Item (
    Serial_Number INT NOT NULL,
    Manufact_Date DATE NOT NULL,
    Price FLOAT NOT NULL,
    Store_ID INT NOT NULL,
    PRIMARY KEY (Serial_Number),
    FOREIGN KEY (Store_ID) REFERENCES Stores(Store_ID)
);

CREATE TABLE Soldier (
    Soldier_ID INT NOT NULL,
    Birth_Date DATE NOT NULL,
    Soldier_Rank VARCHAR2(20) NOT NULL,
    First_Name VARCHAR2(20) NOT NULL,
    Last_Name VARCHAR2(20) NOT NULL,
    Unit VARCHAR2(20),
    PRIMARY KEY (Soldier_ID)
);

CREATE TABLE Storeman (
    Soldier_ID INT NOT NULL,
    Hours_Per_Week FLOAT NOT NULL,
    Store_ID INT NOT NULL,
    PRIMARY KEY (Soldier_ID),
    FOREIGN KEY (Soldier_ID) REFERENCES Soldier(Soldier_ID),
    FOREIGN KEY (Store_ID) REFERENCES Stores(Store_ID)
);

CREATE TABLE Supplier
(
    Company_Name VARCHAR2(40) NOT NULL,
    Email VARCHAR2(40) NOT NULL,
    Location VARCHAR2(40) NOT NULL,
    Company_ID INT NOT NULL,
    PRIMARY KEY (Company_ID)
);

CREATE TABLE Weapon (
    Weapon_Type VARCHAR2(40) NOT NULL,
    Weight FLOAT,
    Serial_Number INT NOT NULL,
    Reliability INT NOT NULL,
    PRIMARY KEY (Serial_Number),
    FOREIGN KEY (Serial_Number) REFERENCES Item(Serial_Number)
);

CREATE TABLE Uniform (
    Uniform_Type VARCHAR2(40) NOT NULL,
    Uniform_Size VARCHAR2(10) NOT NULL,
    Serial_Number INT NOT NULL,

```



```

    Officiality VARCHAR2(1) NOT NULL Check(Officiality = 'A' OR Officiality =
    'B'),
    PRIMARY KEY (Serial_Number),
    FOREIGN KEY (Serial_Number) REFERENCES Item(Serial_Number)
);

CREATE TABLE Owns (
    Serial_Number INT NOT NULL,
    Soldier_ID INT NOT NULL,
    PRIMARY KEY (Serial_Number, Soldier_ID),
    FOREIGN KEY (Serial_Number) REFERENCES Item(Serial_Number),
    FOREIGN KEY (Soldier_ID) REFERENCES Soldier(Soldier_ID)
);

CREATE TABLE Supplies (
    Company_ID INT NOT NULL,
    Store_ID INT NOT NULL,
    PRIMARY KEY (Company_ID, Store_ID),
    FOREIGN KEY (Company_ID) REFERENCES Supplier(Company_ID),
    FOREIGN KEY (Store_ID) REFERENCES Stores(Store_ID)
);

```

## 1.5 מחיקת הטבלאות (SQL)

```

DROP TABLE Owns CASCADE CONSTRAINTS;
DROP TABLE Supplier CASCADE CONSTRAINTS;
DROP TABLE Stores CASCADE CONSTRAINTS;
DROP TABLE Supplies CASCADE CONSTRAINTS;
DROP TABLE Storeman CASCADE CONSTRAINTS;
DROP TABLE Item CASCADE CONSTRAINTS;
DROP TABLE Weapon CASCADE CONSTRAINTS;
DROP TABLE Uniform CASCADE CONSTRAINTS;
DROP TABLE Soldier CASCADE CONSTRAINTS;

```

הערה: הוספנו את הפעולה cascade כדי להמנע עם בעייה בעת מחיקת טבלה המקושרת במפתח זר למפתח אחר. הפעולה מוחקת את המופע המקושר בטבלה האחרת.

## 1.6 SELECT ALL (SQL)

```

SELECT * FROM Stores;
SELECT * FROM Item;
SELECT * FROM Soldier;
SELECT * FROM Storeman;
SELECT * FROM Supplier;
SELECT * FROM Weapon;
SELECT * FROM Uniform;
SELECT * FROM Owns;
SELECT * FROM Supplies;

```

## 1.7 יצירת הנתונים

### דרך א – mockaroo (דרך עיקרית):<sup>1</sup>

על מנת ליצור את הנתונים, הגדרנו את הסכמות שלנו בmockaroo, הגדרנו ערכים אפשריים לכל אחד מהשדות ולבסוף קיבלנו קובץ sql שעושה insert עבור כל סכמה. עבור הסכמות המכילות את הערכים עליהם מצביעים מפתחות זרים בסכמות אחרות, ייצרנו בהתחלה קובץ CSV אותו הכנסנו לבסיס הנתונים בmockaroo ואז יכולנו לבחור שהשדה יהיה מפתח זר לסכמה שנמצאת בבסיס הנתונים. לבסוף, הרצנו את הקבצים והכנסנו את הנתונים לבסיס הנתונים שלנו.

#### הטבלה Stores

#### קובץ ההגדרות:

The screenshot shows the mockaroo configuration for a table named 'Stores'. It lists three fields with their types and options:

- Store\_ID**: Type: Number, min: 1, max: 999999999, decimals: 0, blank: 0%.
- Base**: Type: Custom List, options: Tze'elim, KiryatMalakhi, TelHashomer, BeitEl, Nevatim, RamatDavid, etc., random selection, blank: 0%.
- Capacity**: Type: Number, min: 0, max: 5000, decimals: 0, blank: 0%.

At the bottom, the configuration shows: # Rows: 400, Format: SQL, Table Name: Stores, and an option to include CREATE TABLE.

הגדרנו את Store\_ID כמספר סדרתי מאחר שהוא מפתח והוא צריך להיות ייחודי. בנוסף, נתנו לו רשימה של מחרוזות המייצגות בסיסים (Custom List) בהם המחסן יכול להיות והגדרנו שהתכולה של המחסן יכולה להיות מספר שלם עד 5000.

#### לאחר ההוספה:

<sup>1</sup> נראה את קובץ הנתונים שיצרנו עבור כל טבלה ואת בסיס הנתונים לאחר ההכנסה (והרצת SelectAll). נסביר את ההגדרות בשתי הדוגמאות הראשונות ולאחר מכן לשאר הסכמות נוסף צילומי מסך של ההגדרות ובסיס הנתונים לאחר ההכנסה של הנתונים.

STORE_ID	BASE	CAPACITY
1	7998045433 Efraim	423
2	5260113197 Yadin	808
3	5381021633 Katif	4167
4	8719506268 RamatDavid	2069
5	3220487253 Palmachim	4264
6	8994723781 Nevatim	1168
7	8015551792 Julis	2318
8	4309160419 Dotan	4401
9	4473030814 TelHashomer	3277
10	7323150268 Efraim	4535
11	4305773626 Dotan	1037
12	4094829436 Katif	557
13	4041541856 HaKirya	4748
14	4254840602 Yadin	541
15	8741317356 Zikim	4187
16	3029221199 TelHashomer	1054
17	7272891544 RamatDavid	2179

## הטבלה Supplies

### קובץ ההגדרות:

### Supplies

Field Name	Type	Options
Company_ID	Dataset Column	Suppliers Company_ID sequential blank: 0 % Σ ×
Store_ID	Dataset Column	Stores Store_ID sequential blank: 0 % Σ ×

+ ADD ANOTHER FIELD
GENERATE FIELDS USING AI...

# Rows: 400
Format: SQL
Table Name: Supplies
☐ include CREATE TABLE

ל-Supplies יש שתי תכונות שהן מפתחות זרים: Store\_ID ו-Company\_ID. את Store\_ID יצרנו ב-mockaroo והכנסנו לבסיס הנתונים שלו ולכן יכולנו להגדיר את התכונה כמפתח זר, ואילו את Company\_ID מהטבלה Suppliers שיצרנו בעזרת קובץ הפייתון היינו צריכים לייבא ל-mockaroo ואז יכולנו להגדיר את התכונה כמפתח זר.

### לאחר ההוספה:

COMPANY_ID	STORE_ID
1	559146 7998045433
2	757260 5260113197
3	738303 5381021633
4	366104 8719506268
5	143701 3220487253
6	256875 8994723781
7	732488 8015551792
8	291669 4309160419
9	581216 4473030814
10	277568 7323150268
11	650634 4305773626
12	467493 4094829436
13	760359 4041541856
14	953422 4254840602
15	717258 8741317356
16	275177 3029221199
17	859667 7272891544
18	444940 3558617930

הטבלה Item

## קובץ ההגדרות:

**Item**

Field Name	Type	Options
Serial_Number	Number	min: 1 max: 999999999 decimals: 0 blank: 0 % Σ ×
Manufact_Date	Datetime	01/01/1990 to 12/31/2023 format: yyyy-mm-dd blank: 0 % Σ ×
Price	Number	min: 1000 max: 200000 decimals: 2 blank: 0 % Σ ×
Store_ID	Dataset Column	Stores Store_ID random blank: 0 % Σ ×

+ ADD ANOTHER FIELD    GENERATE FIELDS USING AI...

# Rows: 400    Format: CSV    Line Ending: Windows (CRLF)    Include: ☒ header ☐ BOM

## לאחר ההוספה:

	SERIAL_NUMBER	MANUFACT_DATE	PRICE	STORE_ID
1	7053260187	08/11/2017	138380.49	2051033117
2	3680210664	30/06/1997	110537.22	3199368328
3	5836903518	04/08/2007	76494.52	1697357799
4	8451575436	10/12/2017	116264.28	5499267131
5	9726764684	30/11/1996	197955.3	198493237
6	1834828258	08/10/2001	196973.01	4993512060
7	3330706469	06/09/2021	40299.62	3466456849
8	5756394082	19/01/2011	83506.78	3006393424
9	38179425	26/03/2023	10228.39	7194272436
10	390672970	25/07/2023	194493.57	9691746383
11	9676581907	04/11/2007	62612.84	6954833971
12	3873774387	10/06/1997	16237.58	112043827
13	5642982211	15/05/1991	20700.99	5025045083
14	692110927	16/05/2019	171674.36	6996515713
15	1582602210	30/08/1994	161882.65	9322452254
16	4389875876	29/09/1997	62208.95	9662472323
17	2263419287	14/06/2016	65216.24	7873678207
18	7467542023	23/11/2005	10925.42	4798387437

1 of 400    0.01    shilo@XE AS SYSDBA    [19:45:27] 400 rows selected in 1.012 seconds

הטבלה Uniform

## קובץ ההגדרות:

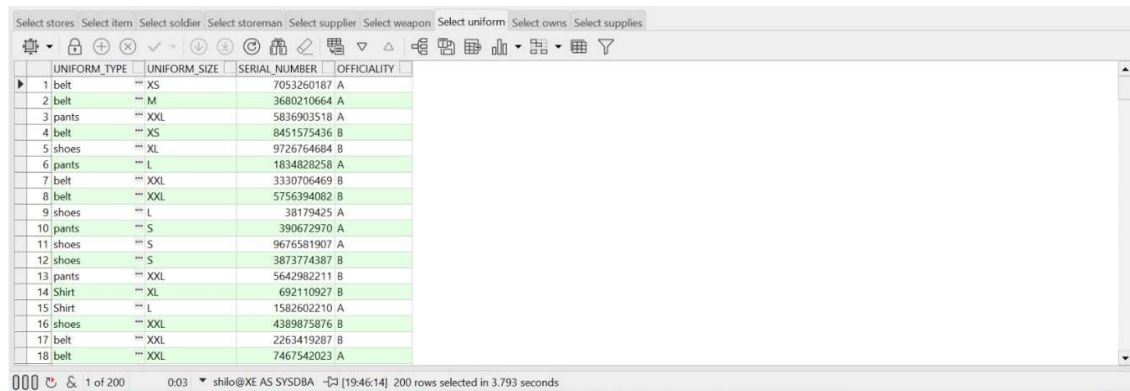
**Uniform**

Field Name	Type	Options
Uniform_Type	Custom List	Shirt, pants, belt, shoes random blank: 0 % Σ ×
Uniform_Size	Custom List	XS,S,M,L,XL,XXL random blank: 0 % Σ ×
Serial_Number	Dataset Column	Item Serial_Number sequential blank: 0 % Σ ×
Officiality	Custom List	A,B random blank: 0 % Σ ×

+ ADD ANOTHER FIELD    GENERATE FIELDS USING AI...

בס"ד

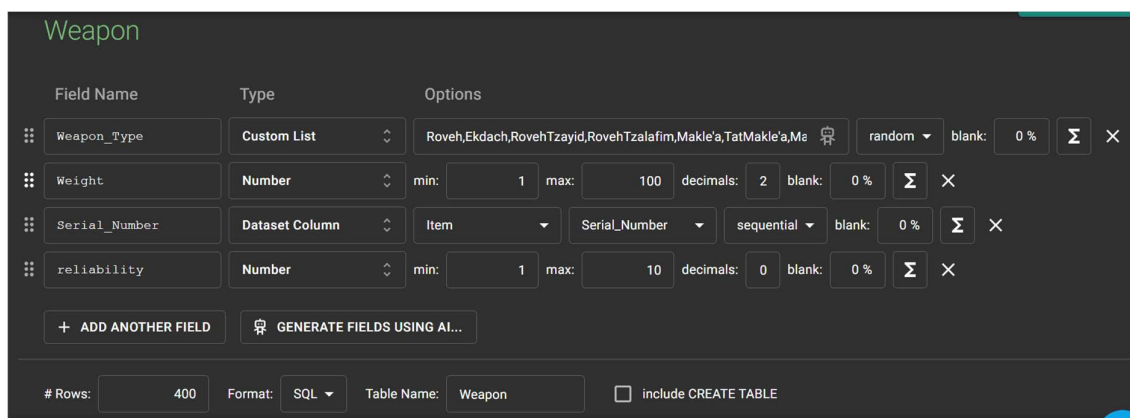
לאחר ההוספה:



	UNIFORM_TYPE	UNIFORM_SIZE	SERIAL_NUMBER	OFFICIALITY
1	belt	XS	7053260187	A
2	belt	M	3680210664	A
3	pants	XXL	5836903518	A
4	belt	XS	8451575436	B
5	shoes	XL	9726764684	B
6	pants	L	1834828258	A
7	belt	XXL	3330706469	B
8	belt	XXL	5756394082	B
9	shoes	L	38179425	A
10	pants	S	390672970	A
11	shoes	S	9676581907	A
12	shoes	S	3873774387	B
13	pants	XXL	5642982211	B
14	Shirt	XL	692110927	B
15	Shirt	L	1582602210	A
16	shoes	XXL	4389875876	B
17	belt	XXL	2263419287	B
18	belt	XXL	7467542023	A

הטבלה Weapon

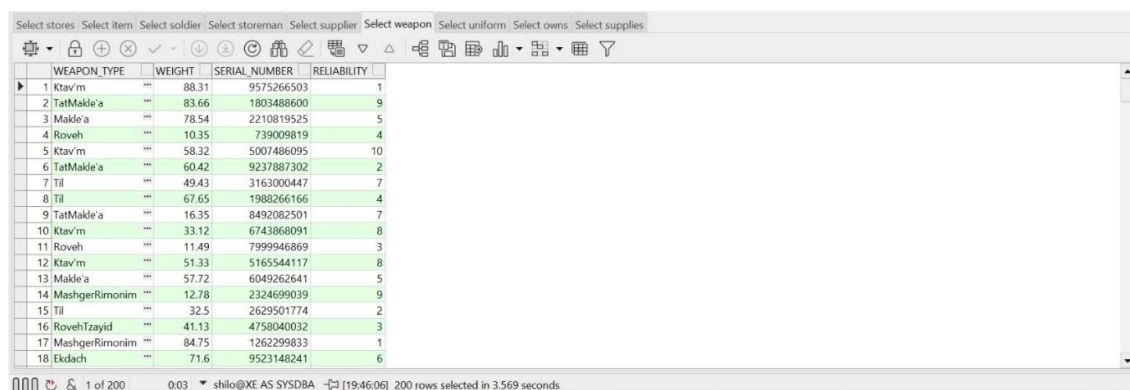
קובץ ההגדרות:



Field Name	Type	Options
Weapon_Type	Custom List	Roveh,Ekdach,RovehTzayid,RovehTzafim,Makle'a,TatMakle'a,Me random blank: 0 %
Weight	Number	min: 1 max: 100 decimals: 2 blank: 0 %
Serial_Number	Dataset Column	Item Serial_Number sequential blank: 0 %
reliability	Number	min: 1 max: 10 decimals: 0 blank: 0 %

# Rows: 400 Format: SQL Table Name: Weapon include CREATE TABLE

לאחר ההוספה:



	WEAPON_TYPE	WEIGHT	SERIAL_NUMBER	RELIABILITY
1	Ktav'm	88.31	9575266503	1
2	TatMakle'a	83.66	1803488600	9
3	Makle'a	78.54	2210819525	5
4	Roveh	10.35	739009819	4
5	Ktav'm	58.32	5007486095	10
6	TatMakle'a	60.42	9237887302	2
7	Til	49.43	3163000447	7
8	Til	67.65	1988266166	4
9	TatMakle'a	16.35	8492082501	7
10	Ktav'm	33.12	6743868091	8
11	Roveh	11.49	7999946869	3
12	Ktav'm	51.33	5165544117	8
13	Makle'a	57.72	6049262641	5
14	MashgerRimonim	12.78	2324699039	9
15	Til	32.5	2629501774	2
16	RovehTzayid	41.13	4758040032	3
17	MashgerRimonim	84.75	1262299833	1
18	Ekdach	71.6	9523148241	6

## הטבלה Soldier

### קובץ ההגדרות:

**Soldier**

Field Name	Type	Options
Soldier_ID	Number	min: 1 max: 999999999 decimals: 0 blank: 0 %
Birth_Date	Datetime	01/01/1950 to 01/01/2005 format: yyyy-mm-dd blank: 0 %
Soldier_Rank	Custom List	RavTurai,Turai,Samal,SamalRishon,RavSamal,SamalRishonMishneh,RavSamalMishneh,Segen,SegenMi random blank: 0 %
First_Name	Custom List	Aviv,Yael,Eitan,Shira,Yair,Noa,Amir,Maya,Tal,Lior,Gal,Noga,Ronen,Tamar,Itai,Or,Dalia,Ziv,Roni,Talia,Yotam random blank: 0 %
Last_Name	Custom List	Cohen,Levi,Mizrahi,Peretz,Biton,Levy,Amar,Malka,Dayan,Ben-David,Mor,Azoulay,Mizrahi,Biton,Avraham random blank: 0 %
Unit	Custom List	Golani,Givati,Paratroopers,Kfir,Nahal,Maglan,Shayetet13,SayeretMatkal,Egoz,Duvdevan,Yahalom,Oketz random blank: 0 %

+ ADD ANOTHER FIELD GENERATE FIELDS USING AI...

# Rows: 600 Format: CSV Line Ending: Windows (CRLF) Include: ☒ header ☐ BOM

### לאחר ההוספה:

SQL Output Statistics

Select stores Select item Select soldier Select storeman Select supplier Select weapon Select uniform Select owns Select supplies

	SOLDIER_ID	BIRTH_DATE	SOLDIER_RANK	FIRST_NAME	LAST_NAME	UNIT
1	807378623	09/07/1973	RavSamalMishneh	Gal	Levi	Carmeli
2	349984301	23/10/1952	Rotem	Alon	Mizrahi	Givati
3	328124739	09/11/1967	RavSamal	Alon	Malka	IronDomeBattery
4	867539907	17/05/1983	RavSamal	Ronen	Mor	Oketz
5	918922079	19/07/1975	RoshHamatehHaklali	Lia	Dayan	8200
6	655001398	12/07/1983	RavAluf	Ziv	Ben-Avraham	Kidon
7	939435895	13/01/1972	SegenMishneh	Galit	Mizrahi	Bardelas
8	597520467	02/09/1982	RavAlufBait	Shira	Amar	DavidsSlingBattery
9	876816056	10/03/1962	RavTurai	Dana	Levi	Golani
10	896592730	13/02/1984	SegenMishneh	Erez	Azoulay	Givati
11	270641945	03/09/1963	Aluf	Matan	Ben-Haim	DavidsSlingBattery
12	894326216	31/01/1978	RavSamalMishneh	Ziv	Dayan	DavidsSlingBattery
13	330583188	28/03/1990	RavAluf	Elad	David	Kfir
14	412660931	07/03/1955	Turai	Noga	Mizrahi	Oketz
15	863879911	01/01/1951	RavTurai	Yonatan	Ben-Avraham	Alpinist
16	78008934	15/02/1967	RavAlufBait	Oren	Azoulay	Yahalom
17	276412817	09/01/1959	RavAluf	Dana	Azulay	Kfir
18	707520332	19/07/1998	SamalRishon	Eitan	Mor	Carmeli

1 of 600 0:02 shilo@XE AS SYSDBA [19:45:41] 600 rows selected in 2.016 seconds

## דרך ב – פייתון:

### הטבלה Supplier

בעזרת פייתון ייצרנו נתונים עבור הישות Supplier. ראשית הגדרנו קבוצת מיילים, מיקומים ושמות של חברות ולאחר מכן ייצרנו קובץ sql שמכניס לבסיס הנתונים 400 ספקים. (חתכנו לטובת הדו"ח את הרשימה הארוכה של שמות החברות).

### הסקריפט:

```
import random
import string

# Function to generate a random company name
def random_company_name():
    return random.choice(["Apple", "Microsoft", "Amazon", "Google",
"Facebook", "Tesla", "Walmart", "NVIDIA", ...])

# Function to generate a random email
def random_email(company_name):
    domains = ["example.com", "mail.com", "company.com", "business.com"]
    return f"{company_name.lower()}@{random.choice(domains)}"

# Function to generate a random location
def random_location():
    locations = ["New York", "Los Angeles", "Chicago", "Houston", "Phoenix",
"Philadelphia", "San Antonio", "San Diego", "Dallas", "San
Jose"]
    return random.choice(locations)

# File to write the SQL insert statements
with open('insert_suppliers.sql', 'w') as file:

    for i in range(1, 401): # Generate 400 suppliers
        company_name = random_company_name()
        email = random_email(company_name)
        location = random_location()
        company_id = i
        insert_statement = (
            f"INSERT INTO Supplier (Company_Name, Email, Location, Company_ID)
"
            f"VALUES ('{company_name}', '{email}', '{location}',
{company_id});\n"
        )

        # Write the insert statement to the file
```

```
file.write(insert_statement)

file.write('COMMIT;\n') # Commit transaction

print('insert_suppliers.sql file created successfully.')
```

## לאחר ההוספה:

	COMPANY_NAME	EMAIL	LOCATION	COMPANY_ID
1	Apple	tpeddersen0@last.fm	Williston	559146
2	Microsoft	teort1@prlog.org	Douz	757260
3	Amazon	fhaynes2@trellan.com	Pocito	738303
4	Google	ddenholm3@360.cn	Zaqatala	366104
5	Facebook	rmaglibbon4@nifty.com	Teberda	143701
6	Tesla	mlhommeau5@mail.ru	Maunatlala	256875
7	BerkshireHathaway	lhuccaby6@blogtalkradio.com	Langres	732488
8	JohnsonJohnson	fhacket7@oaic.gov.au	Cibunar	291669
9	Walmart	rcmins8@ow.ly	Jiyukou	581216
10	JPMorganChase	drenfield9@google.ca	Klyuchi	277568
11	ProcterGamble	gtomasza@hhs.gov	Oslo	650634
12	NVIDIA	smattssonb@ovh.net	Ranot	467493
13	Visa	hisonc@si.edu	Nyakhachava	760359
14	Mastercard	imainwaringd@cornell.edu	Boavista	953422
15	UnitedHealthGroup	hmettse@123-reg.co.uk	Hanyin Chengguanzhen	717258
16	Disney	gtresise@paypal.com	Sakai	275177
17	Nike	pbutreg@cloudflare.com	Sai&™erlong Xiang	859667
18	PayPal	kshooterh@cmu.edu	Tsuen Wan	444940

## דרך ג - data generator:

לאחר שבמוקרו ובפייתון ייצרנו את פקודות הinsert עבור רוב הטבלאות, איחדנו את כל הפקודות לקובץ אחד וייבאנו אותו ל־psql. כעת נותר לנו ייצור הנתונים של הטבלאות האחרונות, Storeman ו־Owns. אותם החלטנו לעשות ב־data generator.

## הטבלה Storeman

## קובץ ההגדרות:

Name	Type	Size	Data
SOLDIER_ID	NUMBER		List(select soldier_id from soldier)
HOURS_PER_WEEK	FLOAT	22	Random(0, 60)
STORE_ID	NUMBER		List(select store_id from store)
*			

הגדרנו לו שייקח מפתחות זרים מהטבלאות soldier ו־store וכן הגדרנו שמספר השעות יהיה בין 0 ל־60.



## לאחר ההוספה:

	SELLER_ID	HOURS_PER_WEEK	STORE_ID
1	807378623	43	7998045433
2	349984301	51	5260113197
3	328124739	23	5381021633
4	867539907	27	8719506268
5	918922079	37	3220487253
6	655001398	27	8994723781
7	939435895	38	8015551792
8	597520467	35	4309160419
9	876816056	8	4473030814
10	896592730	3	7323150268
11	270641945	34	4305773626
12	894326216	19	4094829436
13	330583188	21	4041541856
14	412660931	12	4254840602
15	863879911	60	8741317356
16	78008934	33	3029221199
17	276412817	10	7272891544
18	707520332	13	3558617930

## הטבלה Owns

## קובץ ההגדרות:

Name	Type	Size	Data	Master
SERIAL_NUMBER	NUMBER		List(select SERIAL_NUMBER from Item)	...
SOLDIER_ID	NUMBER		List(select SOLDIER_ID from SOLDIER)	...
*				...

הגדרנו לו שייקח מפתחות זרים מהטבלאות storeItem עבור התכונות המתאימות.

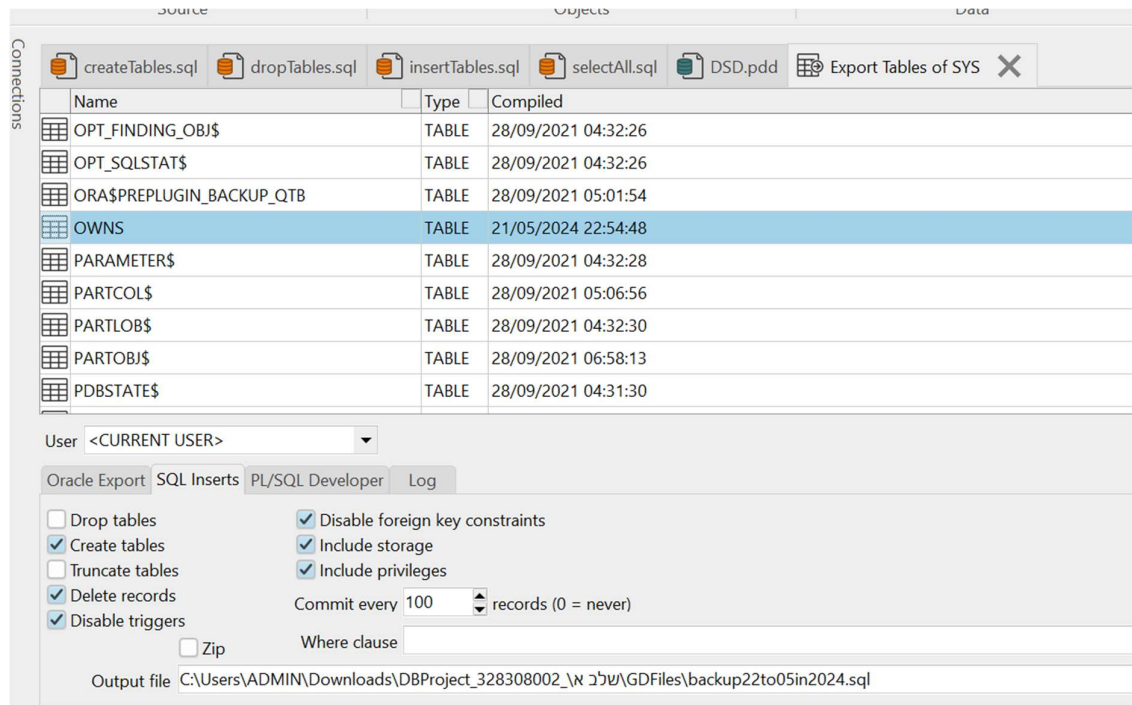
## לאחר ההוספה:

	SERIAL_NUMBER	SOLDIER_ID	STORE_ID
1	7053260187	268280838	
2	3680210664	527623213	
3	5836903518	896601848	
4	8451575436	838807727	
5	9726764684	301683965	
6	1834828258	958408916	
7	3330706469	678454411	
8	5756394082	864915460	
9	38179425	401210905	
10	390672970	14859674	
11	9676581907	787377762	
12	3873774387	61751992	
13	5642982211	346165870	
14	692110927	920019884	
15	1582602210	335043978	
16	4389875876	501339491	
17	2263419287	716887752	
18	7467542023	84777329	

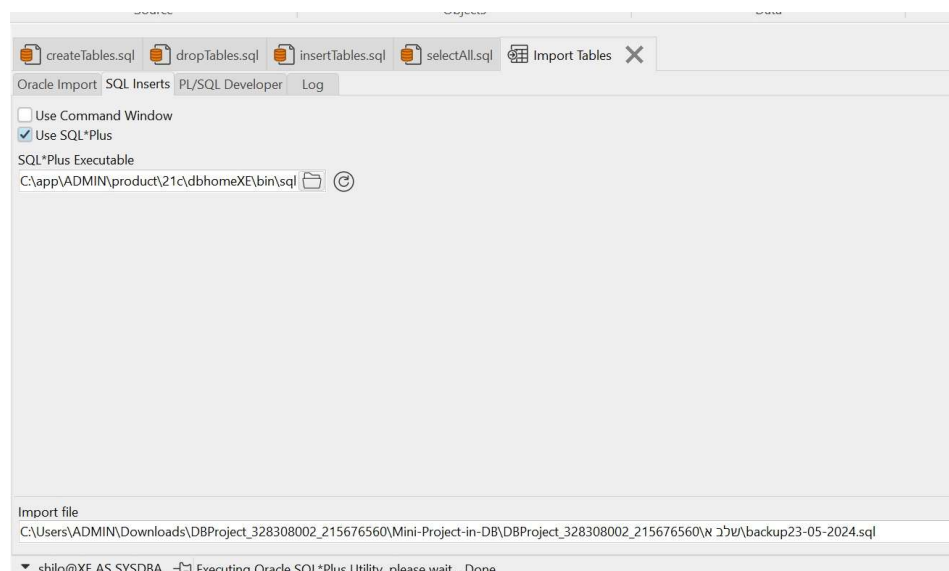
ניתן לראות שהמפתחות הזרים באמת מופיעים בנתונים בטבלאות המתאימות.

## 1.8 גיבוי ושחזור הנתונים

לאחר יצירת הטבלאות והכנסת הנתונים, גיבוי אותם ע"י Export Tables בsqlplus. בחרנו את כל הטבלאות שלנו וסימנו את האפשרות create tables. לאחר מכן בחרנו את המיקום הרצוי אליו נכניס את הגיבוי.



לאחר שקיבלנו את קובץ הsql שמכיל את הגיבוי של הטבלאות שלנו, בדקנו האם הגיבוי אכן עבד. מחקנו את הטבלאות שלנו וניסינו לשחזר את הנתונים ע"י Import Tables:



ואכן הטבלאות שוחזרו:

```
Loading STORES...
100 records committed...
200 records committed...
300 records committed...
400 records loaded
Loading ITEM...
```