Group B

מיני פרויקט בבסיסי נתונים

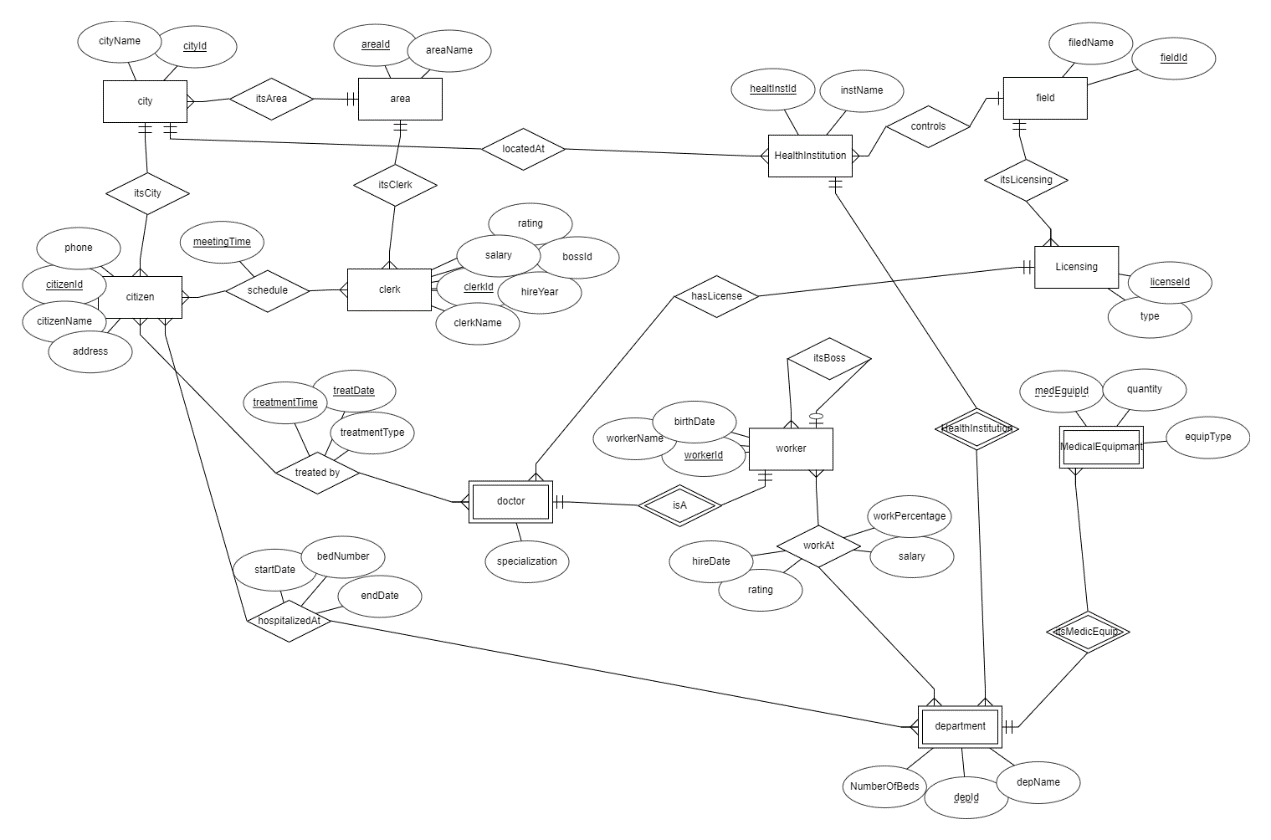
דו"ח שלב 1

עמנואל קואנקה 3940095, שמואל אוריה אשכנזי 312194244

תיאור אגף הגריאטריה

אגף הגריאטריה במשרד הבריאות אחראי על תחום הגריאטריה והמחלות הממושכות במוסדות אשפוז גריאטריים, בבתי החולים הכלליים ובקהילה. אוכלוסיית היעד לפעולות האגף: בני ה- 65 ומעלה וכן חולים במחלות ממושכות עם מגבלות תפקודיות קשות. בין תפקידי האגף: תכנון, קביעות מדיניות, פיתוח שירותים ופיתוח כח אדם, קביעת סטנדרטים ונהלים מקצועיים, רישוי, פיקוח, בקרה ופיתוח אמצעי אכיפה.

# דיאגרמת ERD



תיאור הישויות

**(הערה: קו תחתון מסמל מפתח ראשי של הטבלה)**

**Area (אזור):** מתארת פרטי אזור.

מכילה : **שם אזור (areaName) ומזהה (areaId)**

**City (אזור):** מתארת פרטי עיר.

מכילה : **שם עיר (cityName) ומזהה (cityId)**

**Citizen (אזרח):** מתארת פרטי אזרח.

מכילה : **שם (citizenName) טלפון(phone), כתובת(address) ומזהה (citizenId)**

**Clerk (פקיד):** מתארת פרטי פקיד.

מכילה : **שם פקיד (clerkName), דירוג(rating), משכורת(salary), ת.ז מנהל(bossId), שנת העסקה(hireYear), ומזהה (clerkId)**

**Field (תחום-אגף):** מתארת פרטי אגף במשרד הבריאות.

מכילה : **שם תחום (fieldName) ומזהה (fieldId)**

**Licensing (רישוי):** מתארת פרטי רישוי.

מכילה : **סוג (type) ומזהה (licensingId)**

**Health Institution (מוסד בריאות):** מתארת פרטי מוסד בריאות.

מכילה : **שם (instName) ומזהה (healthInstId)**

**Department (מחלקה):** מתארת פרטי מחלקה במוסד בריאות. **זוהי ישות חלשה לעומת 'מוסד בריאות'.**

מכילה : **מספר מיטות(numOfBeds), שם מחלקה (depName) ומזהה (depId)**

**Medical Equipment (ציוד רפואי):** מתארת פרטי ציוד רפואי של מחלקה. **זוהי ישות חלשה לעומת 'מחלקה'.**

מכילה : **סוג(type), כמות(quantity) ומזהה (medEquipId)**

**Worker (עובד):** מתארת פרטי עובד.

מכילה : **תאריך לידה(birthDate), שם(workerName) ומזהה (workerId)**

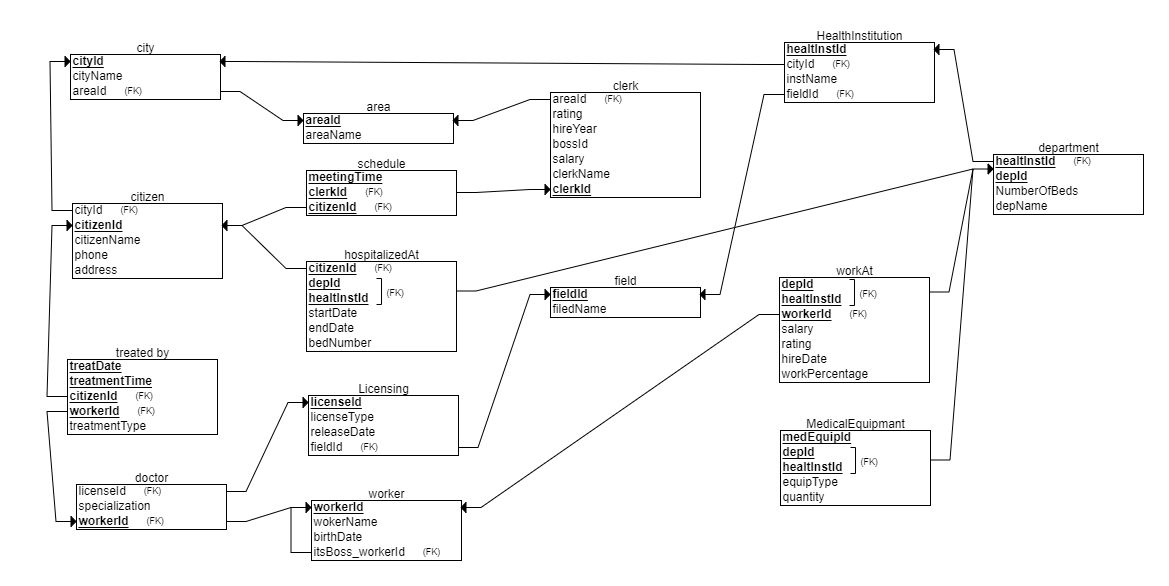
**Doctor (רופא):** מתארת פרטי רופא. **ישות זו יורשת מ'עובד'.**

מכילה : **התמחות (specialization) ומזהה (doctorId)**

תיאור הקשרים

1. **כל עיר שייכת לאזור אחד בדיוק (קשר 1:n)**
2. **כל אזרח שייך לעיר אחת בדיוק (קשר 1:n)**
3. **כל פקיד שייך לאזור אחד בדיוק (קשר 1:n)**
4. **לוח זמנים (schedule) מקשר בין פקיד לאזרח (קשר רבים לרבים) מכיל גם זמן פגישה כחלק מהמפתח.**
5. **לאזרח יש קשר של טיפול עם רופא (קשר רבים לרבים). הקשר מכיל גם סוג טיפול, תאריך ושעה. תאריך ושעה הם חלק מהמפתח של הקשר.**
6. **לאזרח יש קשר של אשפוז עם מחלקה. שכולל תאריכי התחלה וסיום של אשפוז ומספר מיטה.**
7. **לעובד יש קשר רקורסיבי עם עצמו ('המנהל שלו') (קשר 1:n)**
8. **רופא יורש מ'עובד' ויש לו קשר 1:n עם רישוי (לרופא רשיון בודד).**
9. **לרופא יש קשר עם מחלקה שכולל גם אחוז משרה, תאריך העסקה,דירוג ומשכורת. (קשר 1:n)**
10. **כל מחלקה שייכת למוסד בריאות אחד בדיוק (קשר 1:n)**
11. **ציוד רפואי שייך בדיוק מחלקה אחת (קשר 1:n)**
12. **רישוי מסוים ניתן על ידי תחום-אגף רפואי אחד בדיוק (קשר 1:n)**
13. **מוסד בריאות ממוקם בעיר אחת בדיוק (קשר 1:n)**
14. **מוסד בריאות נשלט על ידי תחום-אגף רפואי אחד לכל היותר (קשר 1:n)**

# דיאגרמת DSD



# קוד SQL של יצירת הטבלאות (מסודר לפי התלויות)

CREATE TABLE area (

areaId NUMERIC(3) primary key,

areaName VARCHAR(50) NOT NULL,

);

CREATE TABLE city (

cityId NUMERIC(3) primary key,

cityName VARCHAR(50) NOT NULL,

areaId NUMERIC(3) NOT NULL,

FOREIGN KEY (areaId) REFERENCES area(areaId)

);

CREATE TABLE citizen (

citizenId NUMERIC(9) primary key,

citizenName VARCHAR(50) NOT NULL,

phone VARCHAR(15) NOT NULL,

address VARCHAR(50) NOT NULL,

cityId NUMERIC(3) NOT NULL,

FOREIGN KEY (cityId) REFERENCES city(cityId)

);

CREATE TABLE clerk (

clerkId NUMERIC(9) primary key,

rating NUMERIC(2) NOT NULL,

hireYear NUMERIC(4) NOT NULL,

bossId NUMERIC(9) NOT NULL,

salary FLOAT NOT NULL,

clerkName VARCHAR(50) NOT NULL,

areaId NUMERIC(3) NOT NULL,

FOREIGN KEY (areaId) REFERENCES area(areaId)

);

CREATE TABLE schedule (

meetingTime DATE NOT NULL,

clerkId NUMERIC(9) NOT NULL,

citizenId NUMERIC(9) NOT NULL,

PRIMARY KEY (meetingTime, clerkId, citizenId),

FOREIGN KEY (clerkId) REFERENCES clerk(clerkId),

FOREIGN KEY (citizenId) REFERENCES citizen(citizenId),

);

CREATE TABLE field (

fieldId NUMERIC(3) primary key,

filedName VARCHAR(50) NOT NULL,

);

CREATE TABLE Licensing (

licenseType VARCHAR(50) NOT NULL,

licenseId NUMERIC(3) primary key,

releaseDate DATE NOT NULL,

fieldId NUMERIC(3) NOT NULL,

FOREIGN KEY (fieldId) REFERENCES field(fieldId)

);

CREATE TABLE HealthInstitution (

healtInstId NUMERIC(3) primary key,

instName VARCHAR(50) NOT NULL,

cityId NUMERIC(3) NOT NULL,

fieldId NUMERIC(3) NOT NULL,

FOREIGN KEY (cityId) REFERENCES city(cityId),

FOREIGN KEY (fieldId) REFERENCES field(fieldId)

);

CREATE TABLE department (

depId NUMERIC NOT NULL,

NumberOfBeds NUMERIC(4) NOT NULL,

depName VARCHAR(50) NOT NULL,

healtInstId NUMERIC(3) NOT NULL,

PRIMARY KEY (depId, healtInstId),

FOREIGN KEY (healtInstId) REFERENCES HealthInstitution(healtInstId)

);

CREATE TABLE MedicalEquipmant (

equipType VARCHAR(50) NOT NULL,

medEquipId VARCHAR(3) NOT NULL,

quantity NUMERIC(5) NOT NULL,

depId NUMERIC NOT NULL,

healtInstId NUMERIC(3) NOT NULL,

PRIMARY KEY (medEquipId, depId, healtInstId),

FOREIGN KEY (depId, healtInstId) REFERENCES department(depId, healtInstId)

);

CREATE TABLE hospitalizedAt (

startDate DATE NOT NULL,

endDate DATE NOT NULL,

bedNumber NUMERIC(4) NOT NULL,

citizenId NUMERIC(9) NOT NULL,

depId NUMERIC NOT NULL,

healtInstId NUMERIC(3) NOT NULL,

PRIMARY KEY (citizenId, depId, healtInstId),

FOREIGN KEY (citizenId) REFERENCES citizen(citizenId),

FOREIGN KEY (depId, healtInstId) REFERENCES department(depId, healtInstId)

);

CREATE TABLE worker (

workerId NUMERIC(9) primary key,

wokerName VARCHAR(50) NOT NULL,

birthDate DATE NOT NULL,

itsBoss\_workerId NUMERIC(9) NOT NULL,

FOREIGN KEY (itsBoss\_workerId) REFERENCES worker(workerId)

);

CREATE TABLE doctor (

specialization VARCHAR(50) NOT NULL,

licenseId NUMERIC(3) NOT NULL,

workerId NUMERIC(9) primary key,

FOREIGN KEY (licenseId) REFERENCES Licensing(licenseId),

FOREIGN KEY (workerId) REFERENCES worker(workerId)

);

CREATE TABLE workAt (

salary FLOAT NOT NULL,

rating NUMERIC(2) NOT NULL,

hireDate DATE NOT NULL,

workPercentage NUMERIC(2) NOT NULL,

depId NUMERIC NOT NULL,

healtInstId NUMERIC(3) NOT NULL,

workerId NUMERIC(9) NOT NULL,

PRIMARY KEY (depId, healtInstId, workerId),

FOREIGN KEY (depId, healtInstId) REFERENCES department(depId, healtInstId),

FOREIGN KEY (workerId) REFERENCES worker(workerId)

);

CREATE TABLE treated\_by (

treatDate DATE NOT NULL,

treatmentType VARCHAR(50) NOT NULL,

treatmentTime VARCHAR(50) NOT NULL,

citizenId NUMERIC(9) NOT NULL,

workerId NUMERIC(9) NOT NULL,

PRIMARY KEY (treatDate, treatmentTime, citizenId, workerId),

FOREIGN KEY (citizenId) REFERENCES citizen(citizenId),

FOREIGN KEY (workerId) REFERENCES doctor(workerId)

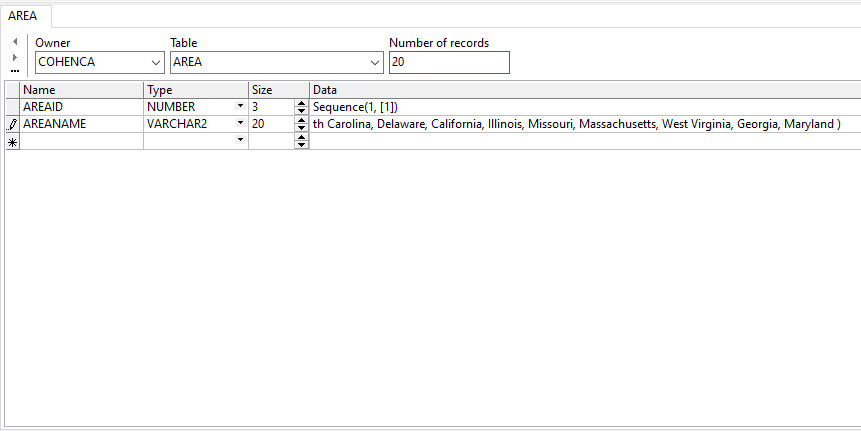
);

# אכלוס טבלאות

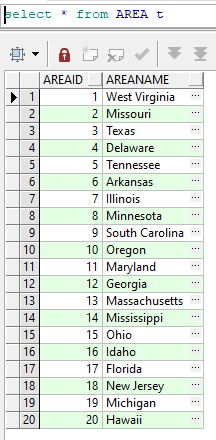
## טבלת Area

אכלסנו באמצעות מחולל הנתונים שב-PLSQL:

### הגדרת המחולל:



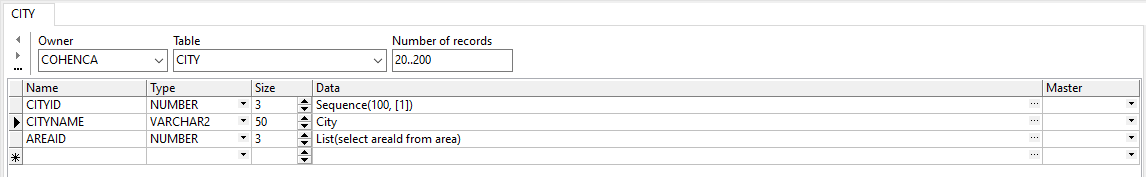
### הטבלה לאחר האכלוס:



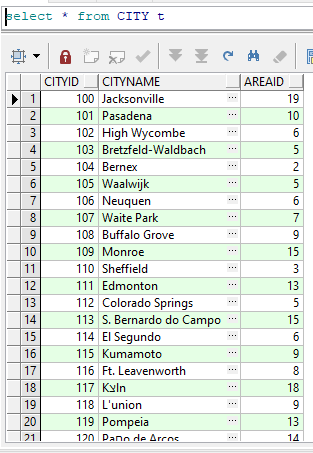
## טבלת City

אוכלסה באמצעות מחולל הנתונים שבPLSQL

### הגדרת המחולל:

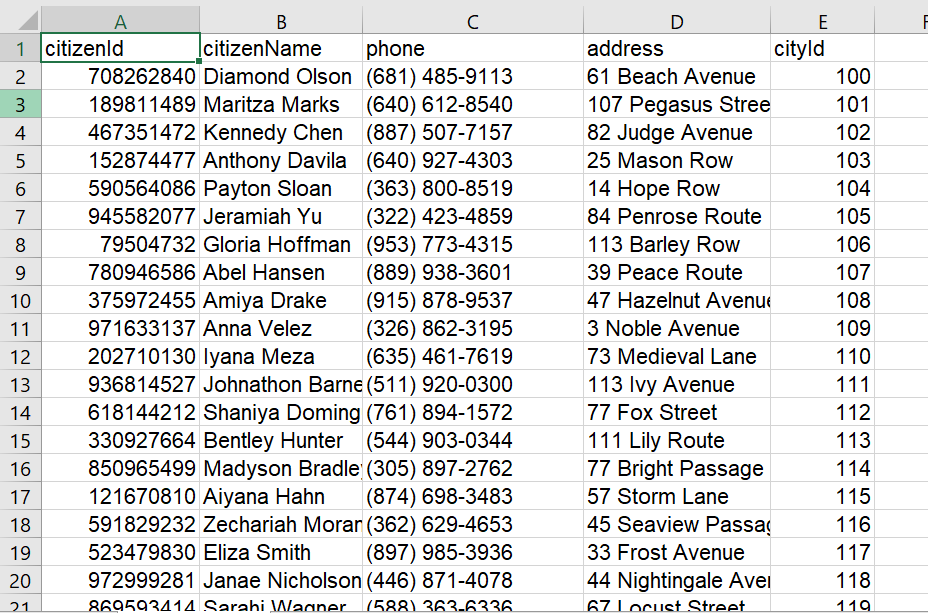


### לאחר האכלוס:

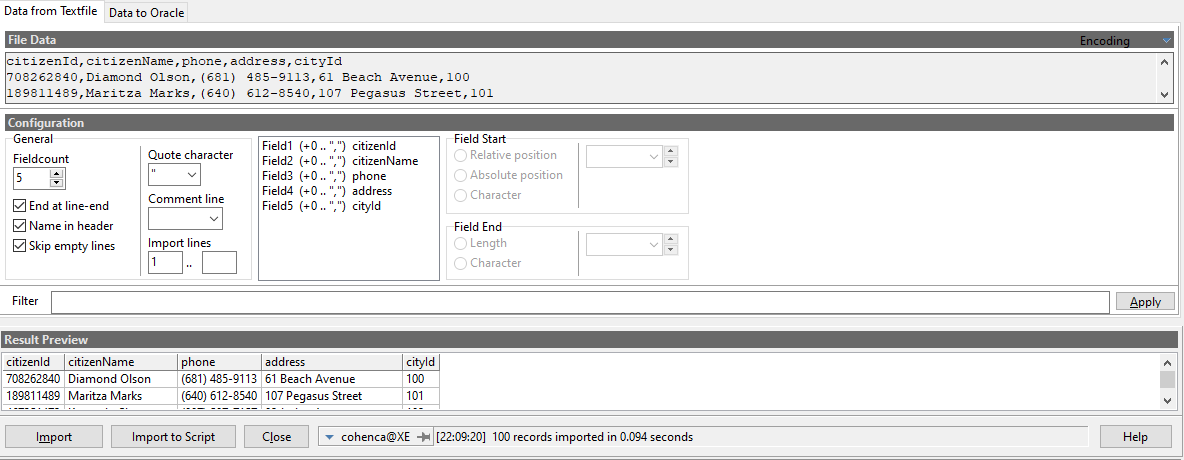


## טבלת citizen:

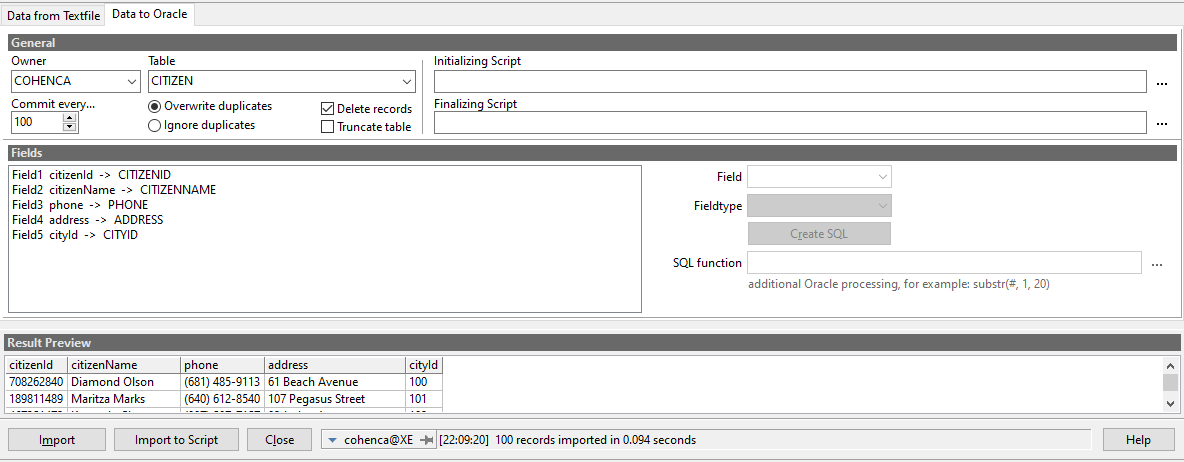
### יצרנו קובץ csv באמצעות excel:



### יבוא הקובץ באמצעות PLSQL:

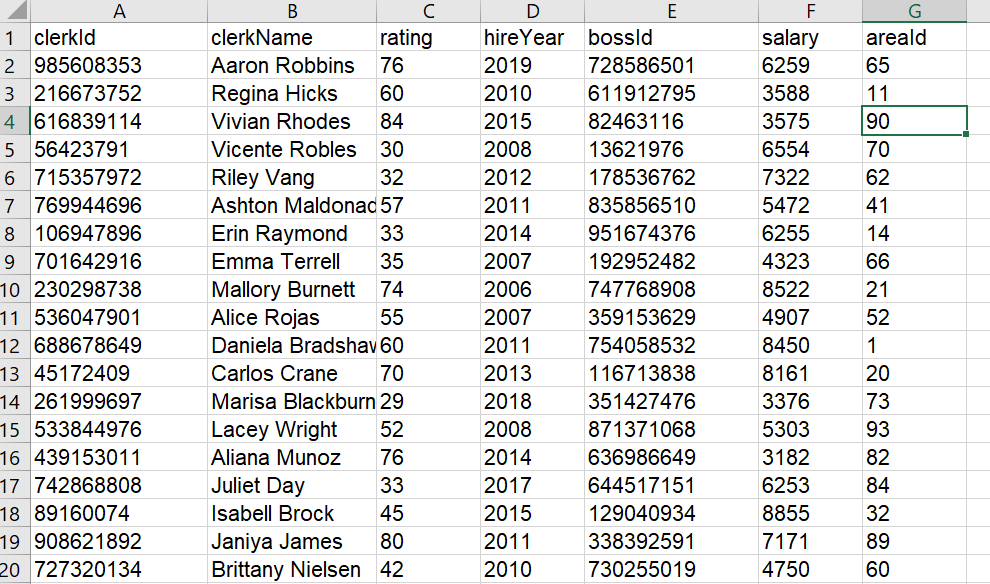


### התאמת הנתונים לטבלה:

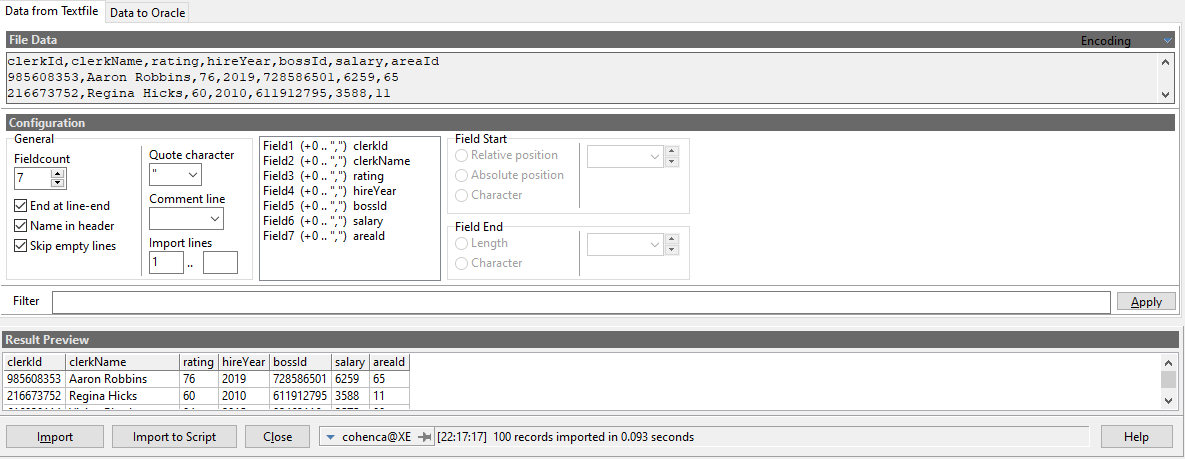


## טבלת clerk:

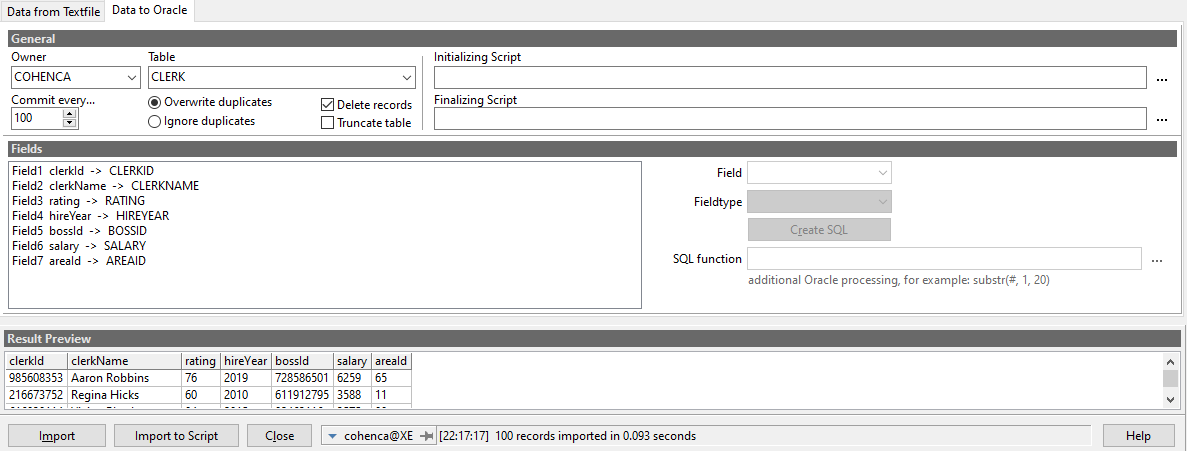
### יצרנו קובץ csv באמצעות excel:



### יבוא הקובץ באמצעות PLSQL:



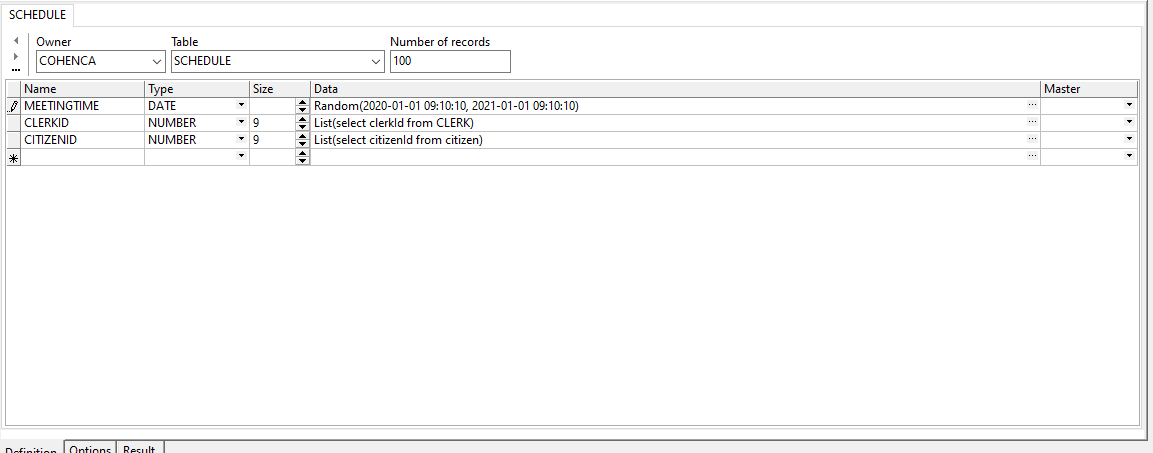
### התאמת הנתונים לטבלה:



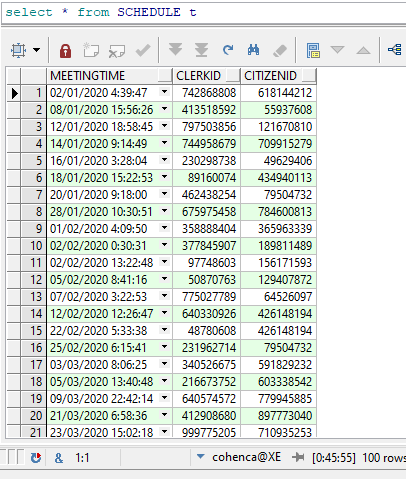
## טבלת schedule:

באמצעות מחולל הנתונים המובנה ב-PLSQL

### הגדרת המחולל:

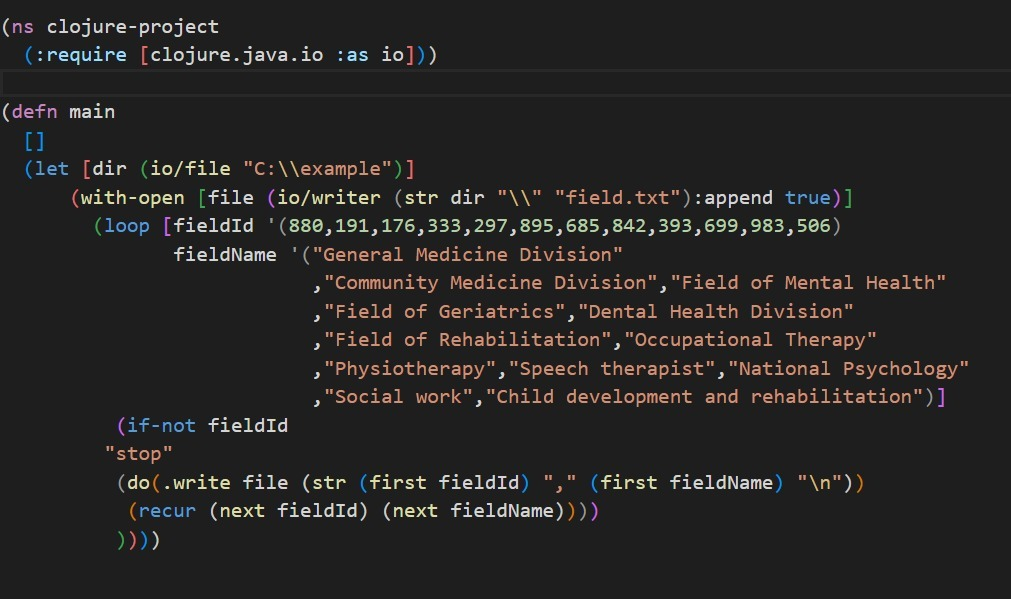


### הטבלה לאחר האכלוס:

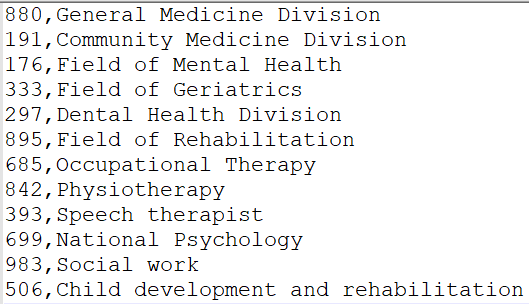
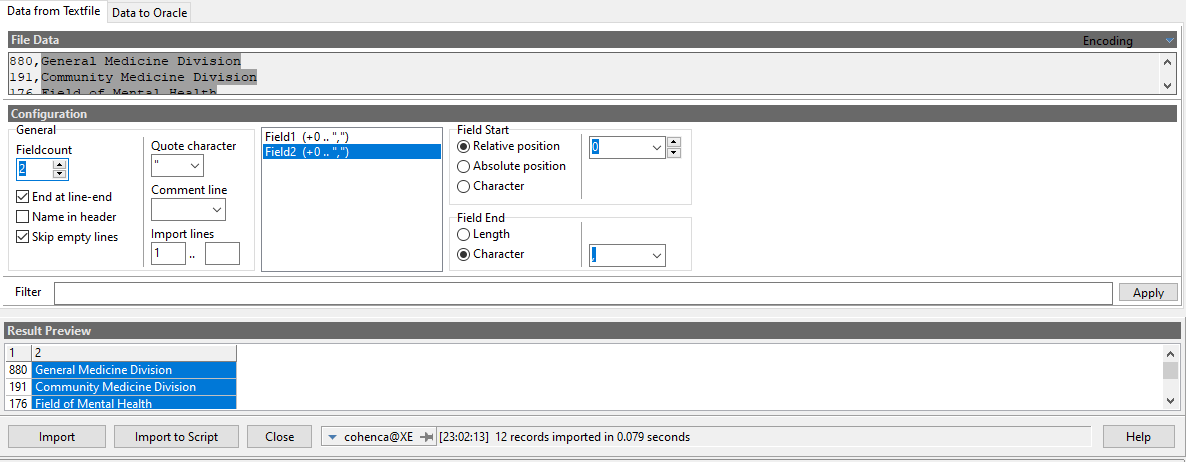


## טבלת field:

### יצרנו קובץ טקסט באמצעות clojure:



### קובץ הטקסט:



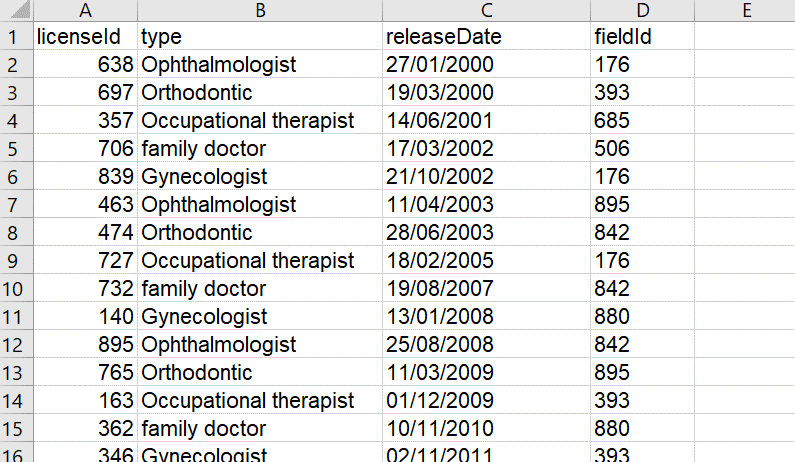
### ייבוא הקובץ:

## 

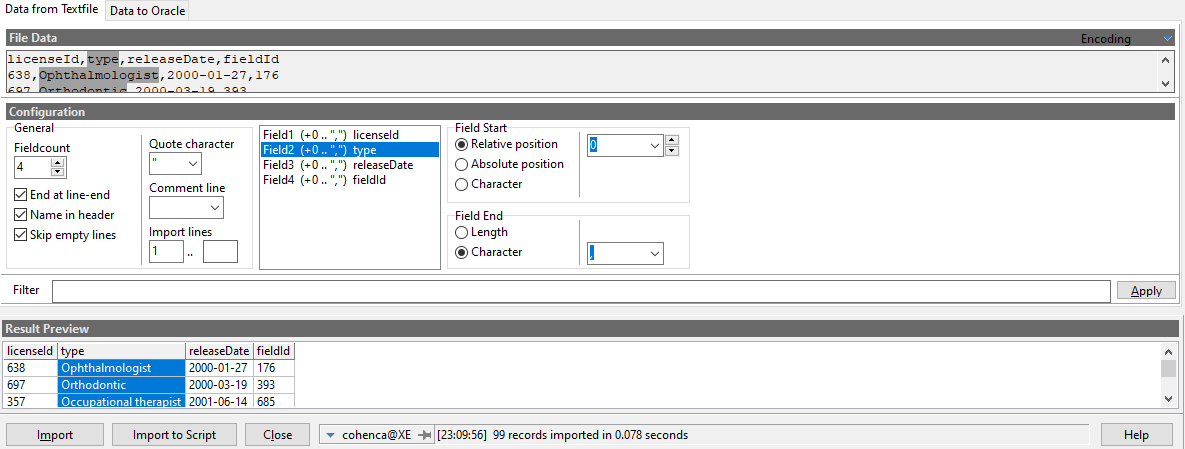
### התאמה לטבלה:

## טבלת licensing:

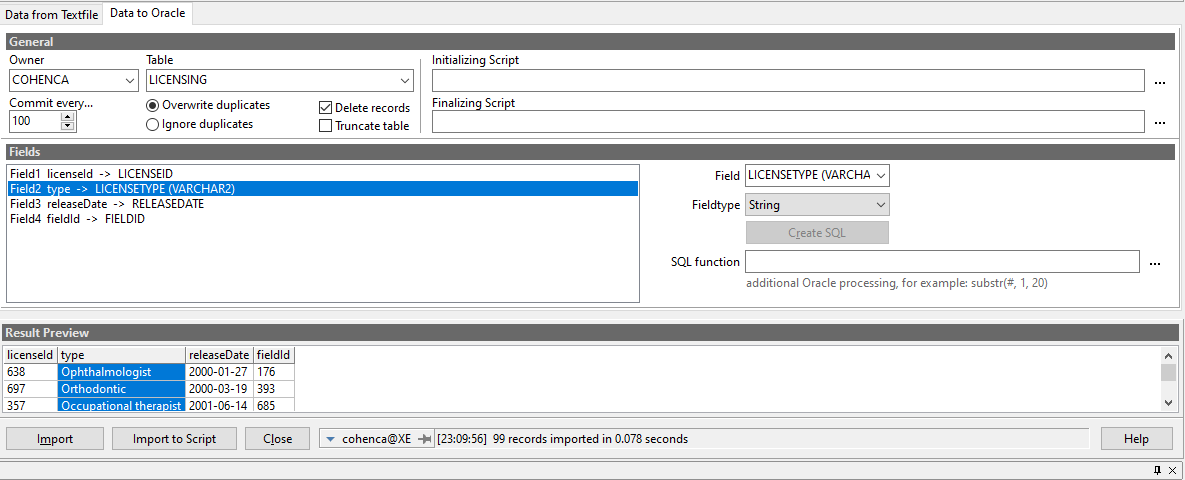
### יצירת csv באמצעות excel:



### ייבוא:

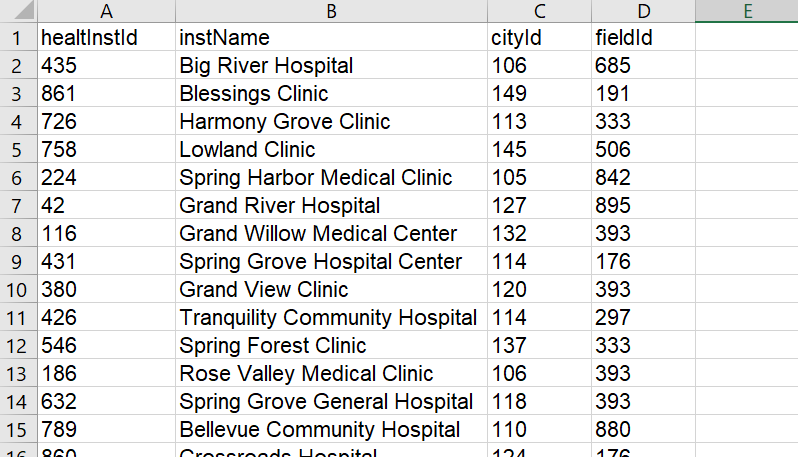


### התאמה לטבלה:

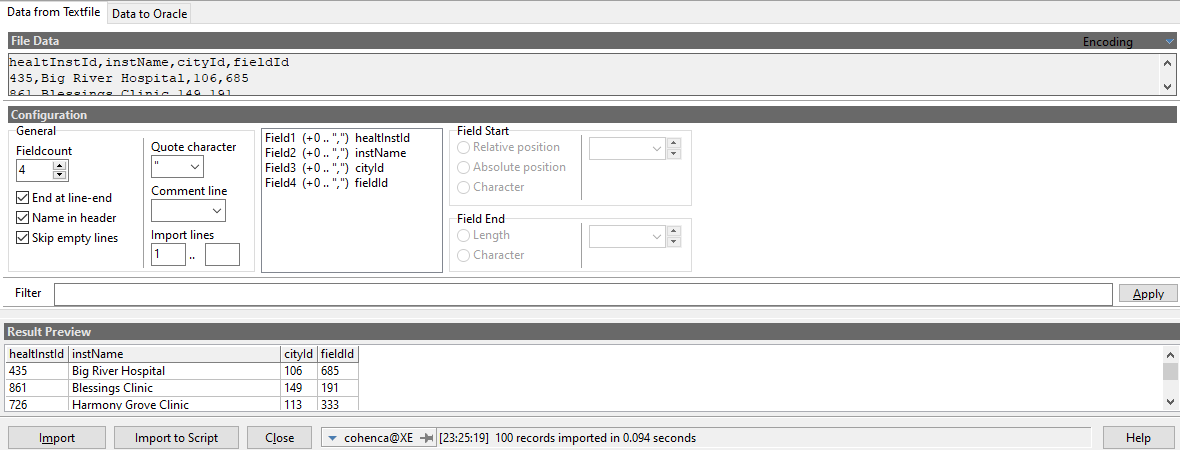


## טבלת :health institution

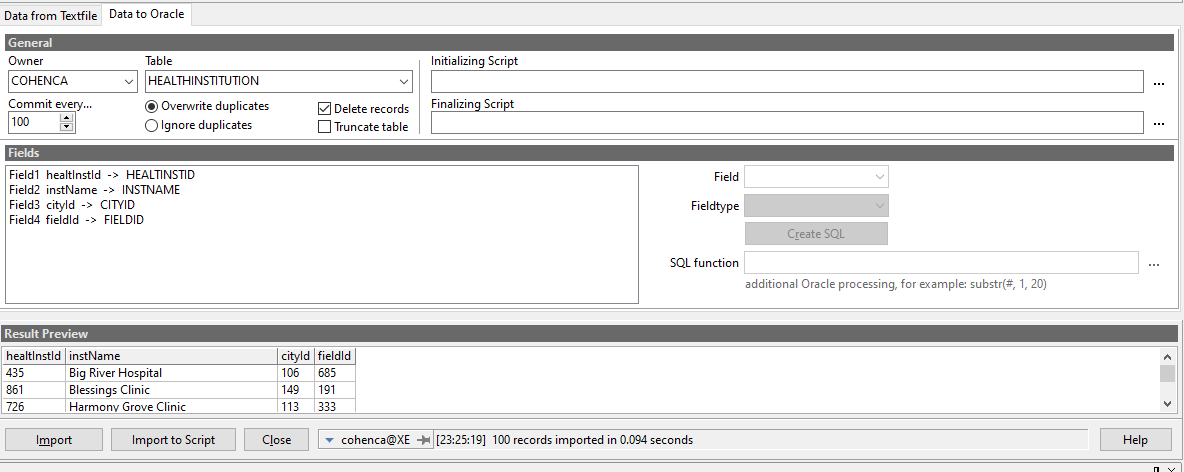
יצירת קובץ csv באמצעות excel:



### ייבוא:

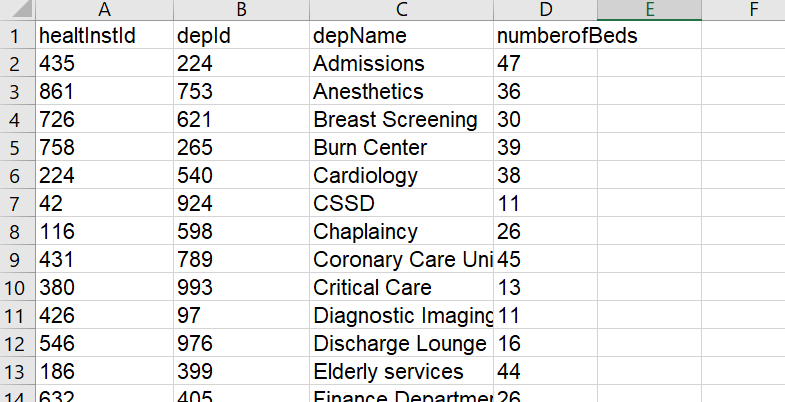


### התאמה לטבלה:

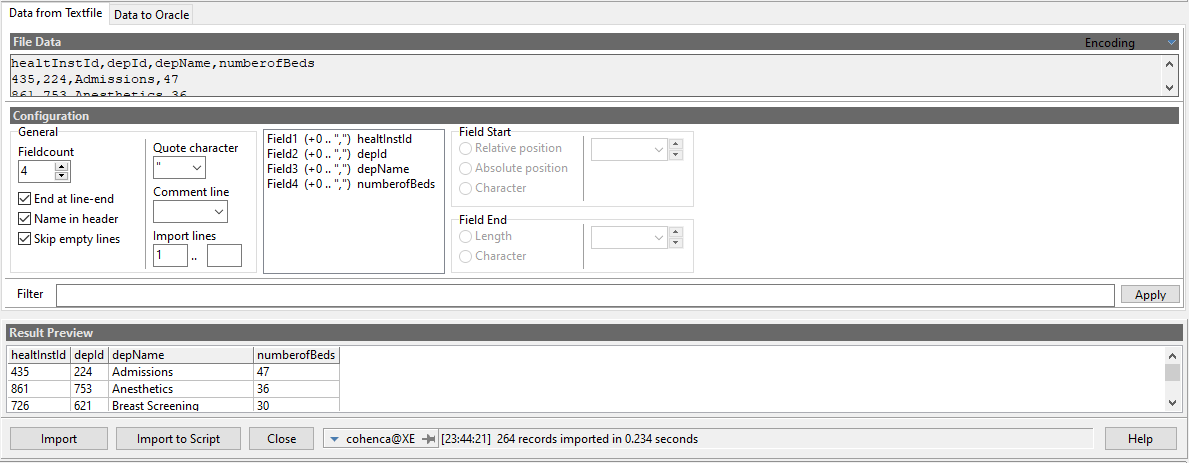


## טבלת department:

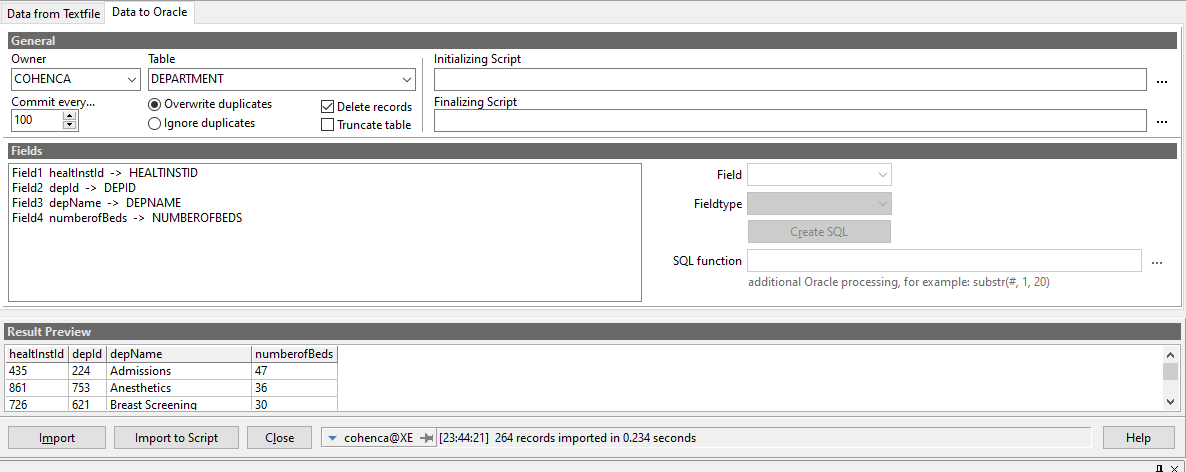
יצירת csv באמצעות excel:



### ייבוא:

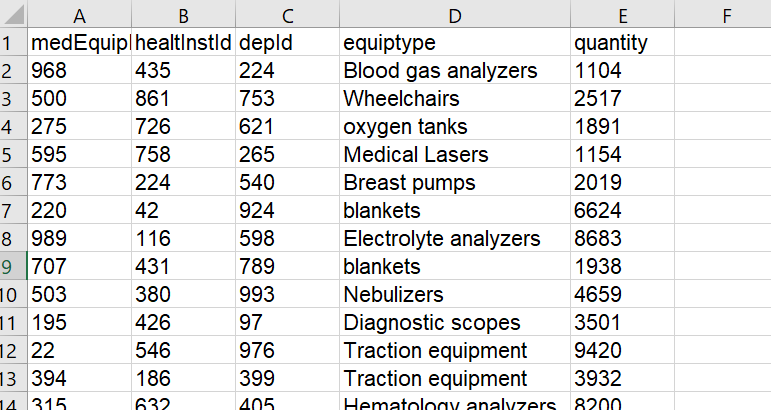


### התאמה לטבלה:

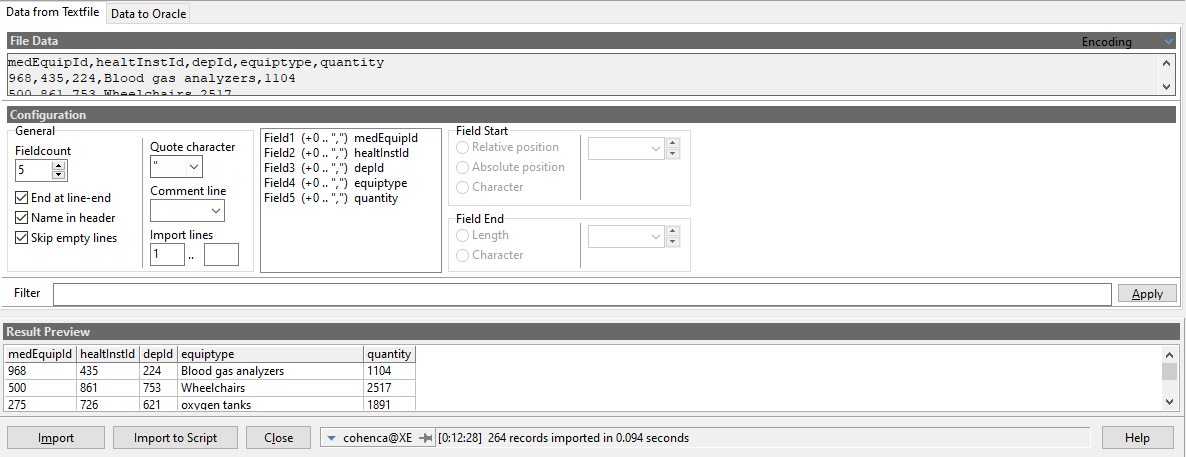


## טבלת medical equipment:

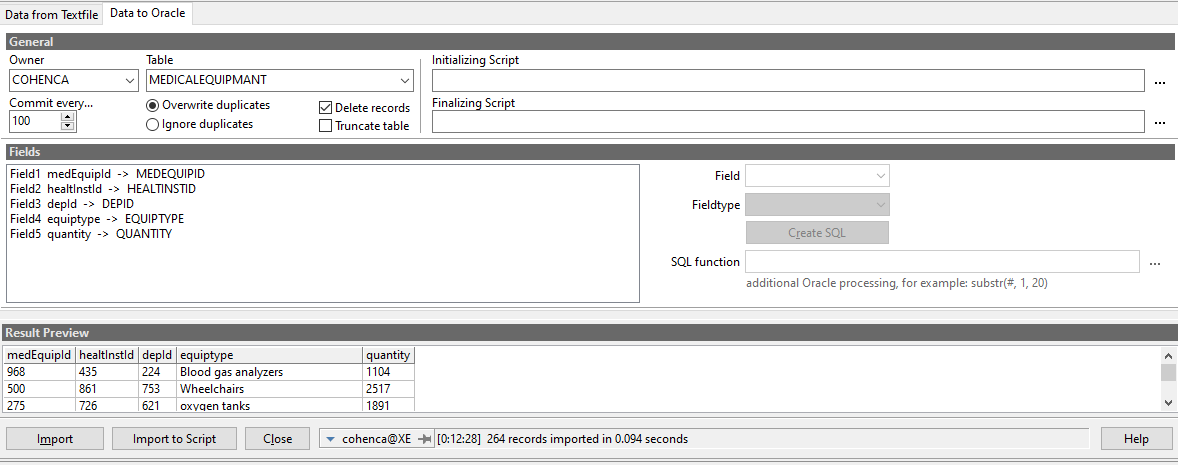
יצירת csv באמצעות excel:



### ייבוא:

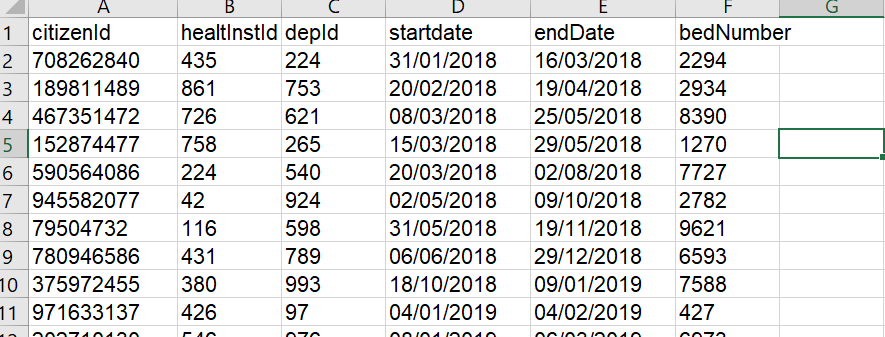


### התאמה לטבלה:

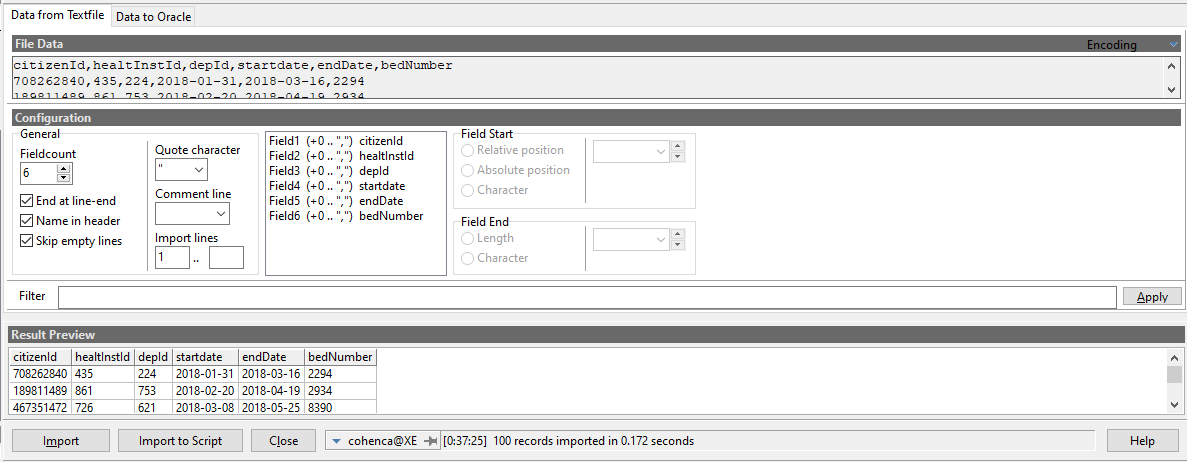


## טבלת hospitalized at:

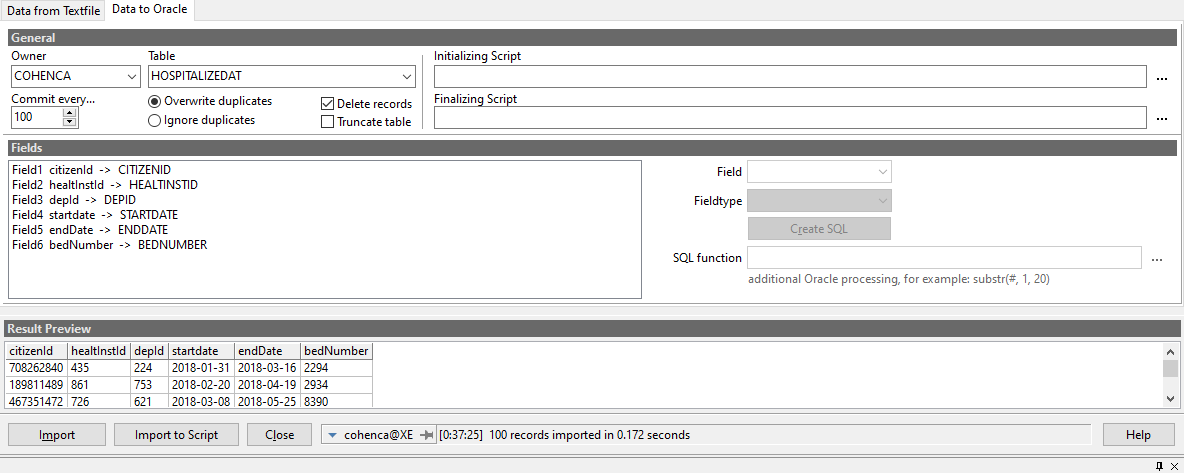
יצירת csv דרך excel:



### יבוא הקובץ ל-PLSQL:

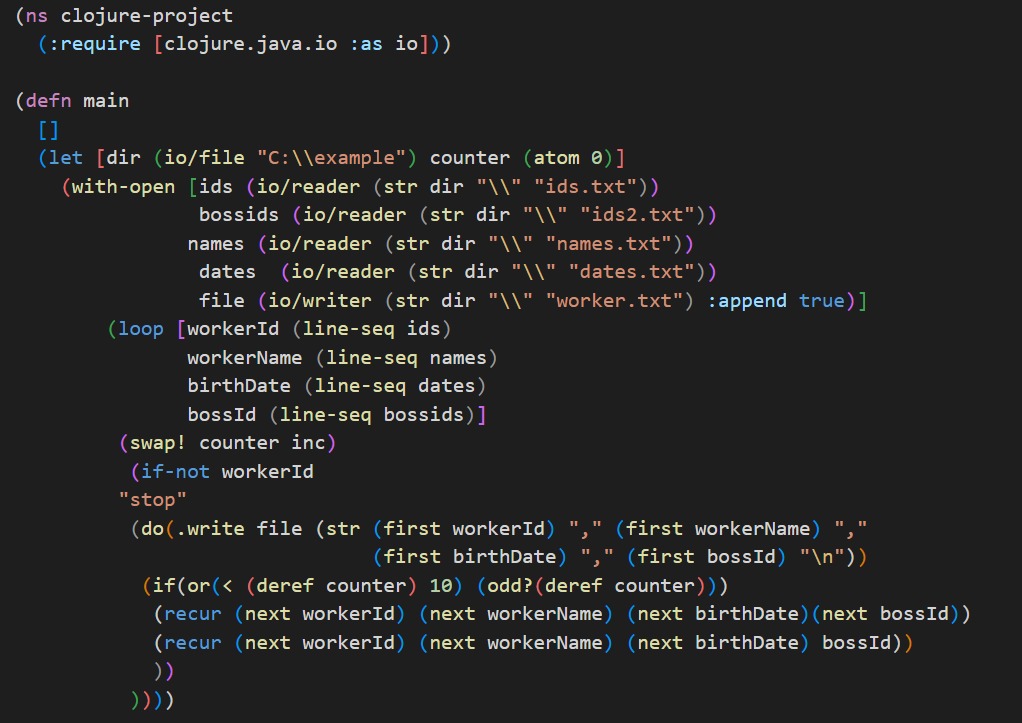


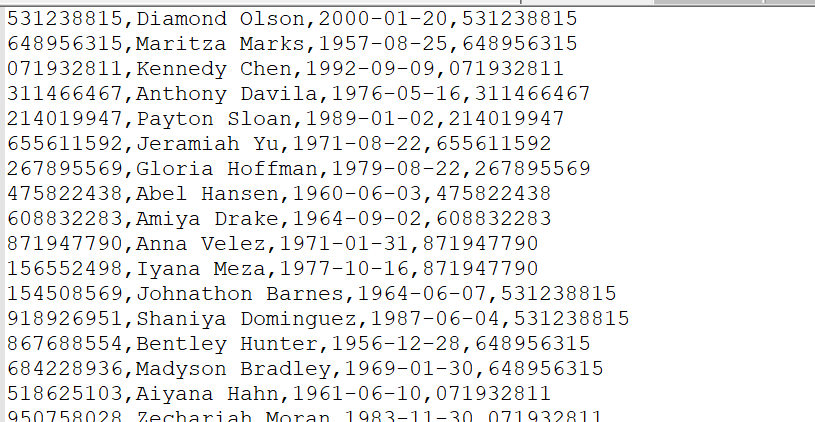
### התאמת הנתונים לטבלה:

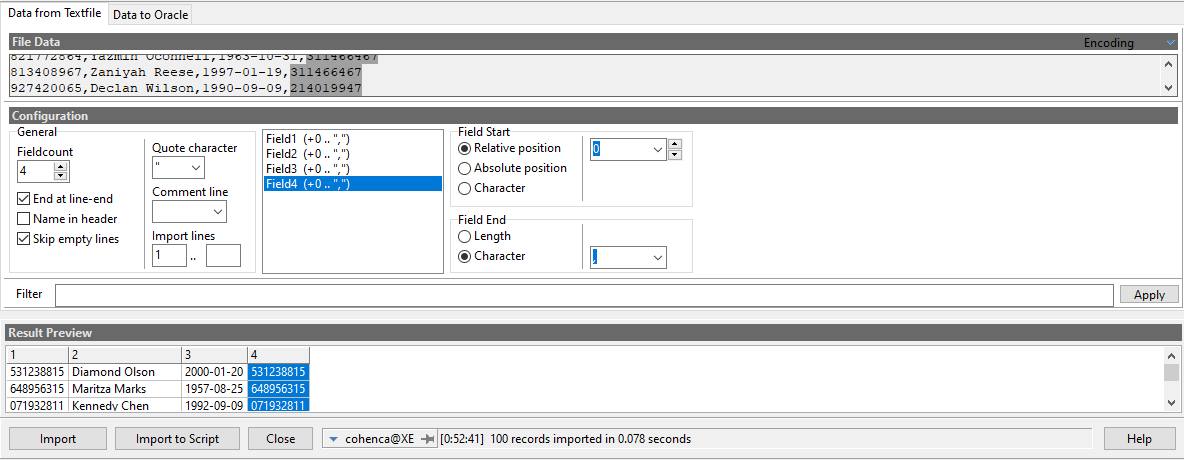


## טבלת worker:

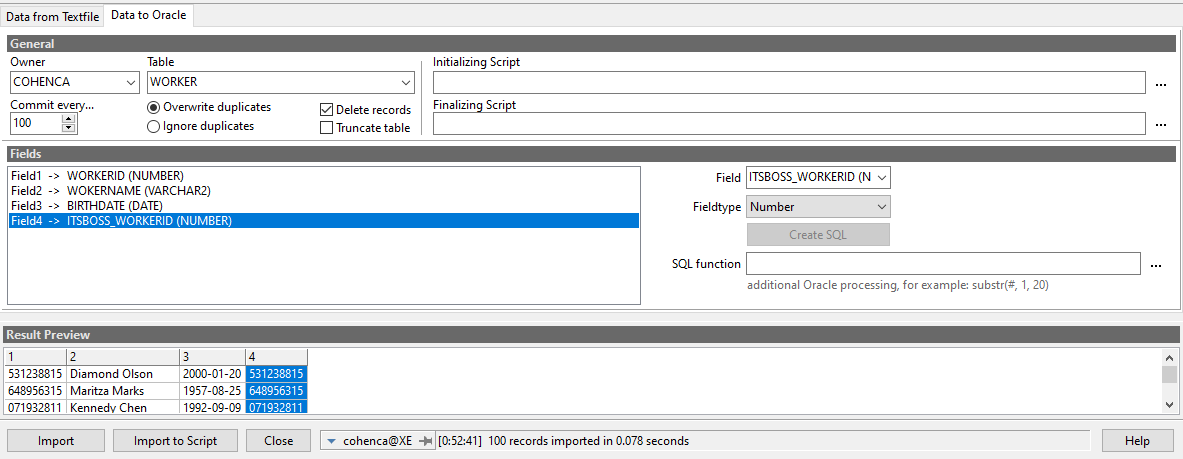
יצירת קובץ טקסט באמצעות clojure:



קובץ הטקסט שנוצר:



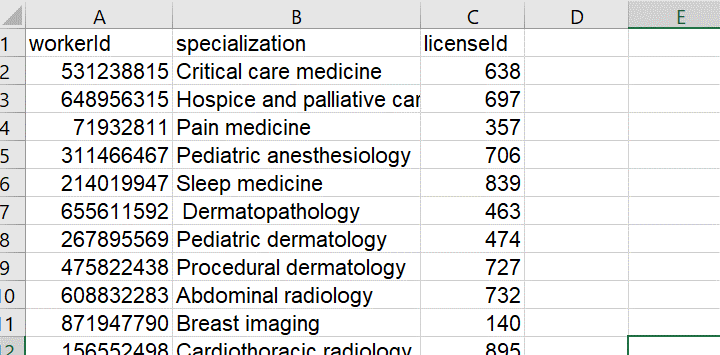
ייבוא:



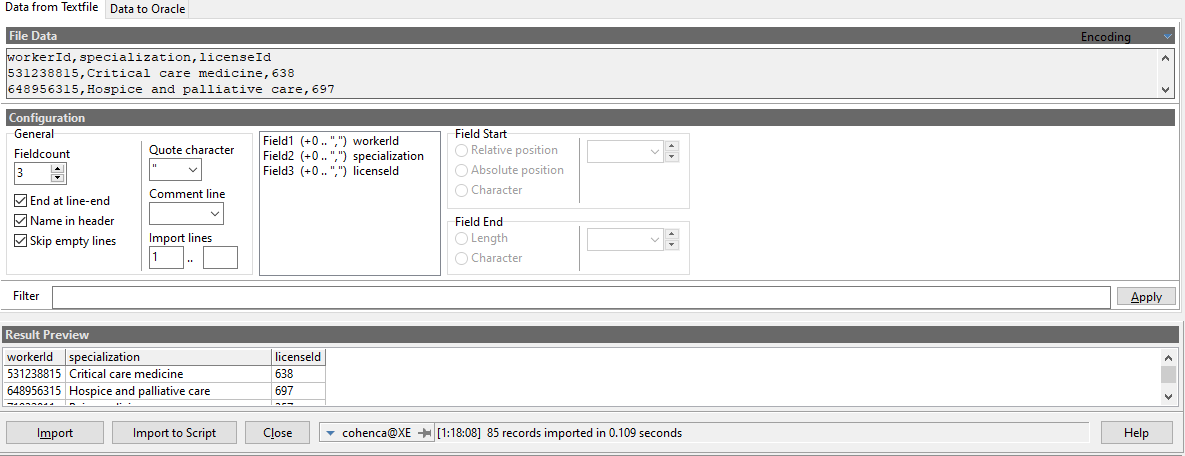
התאמה לטבלה:

## טבלת doctor:

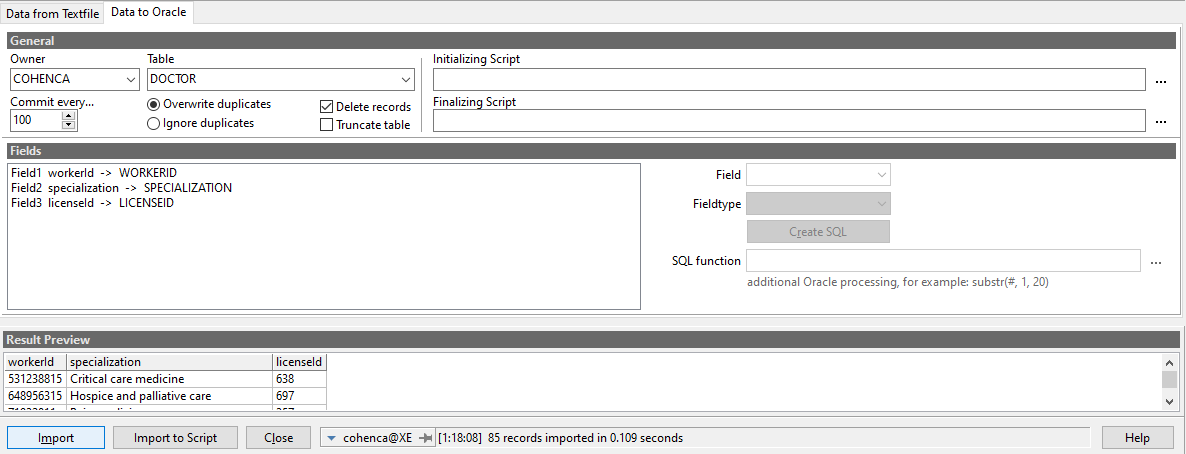
יצירת csv באמצעות excel:



### ייבוא:

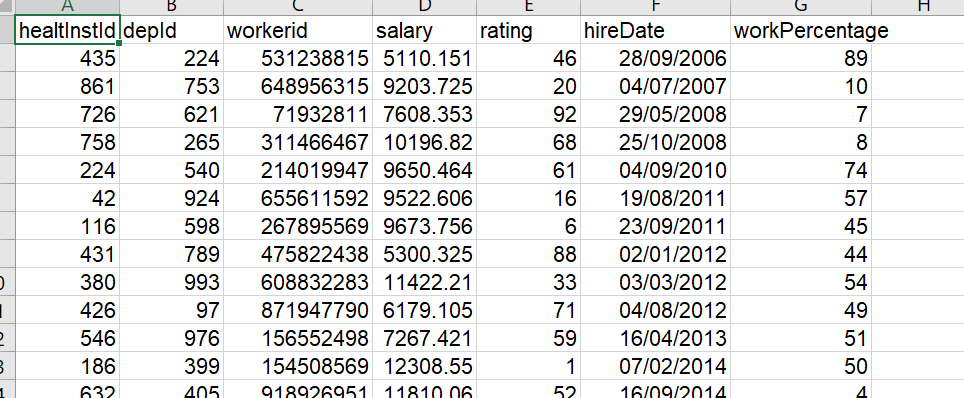


### התאמת הנתונים לטבלה:

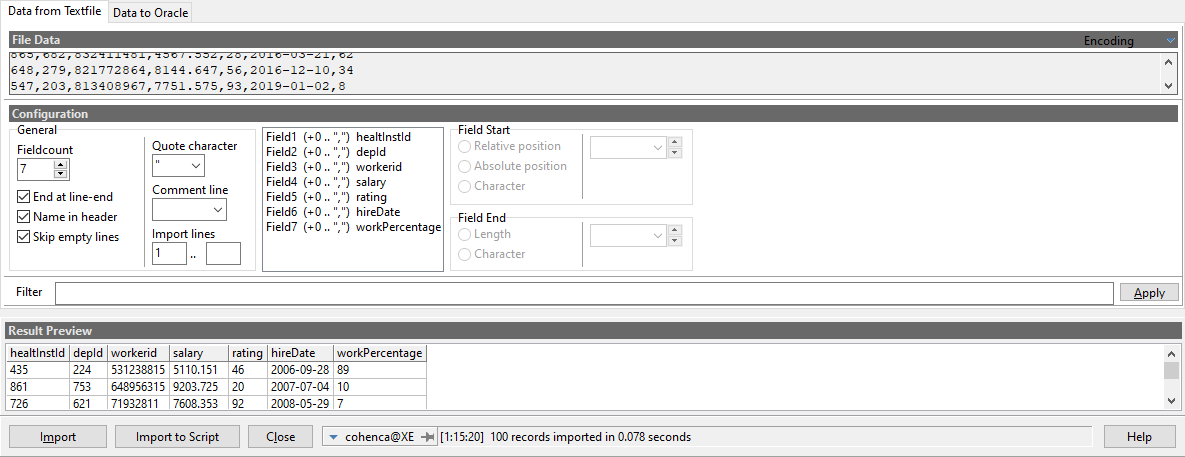


## טבלת workAt:

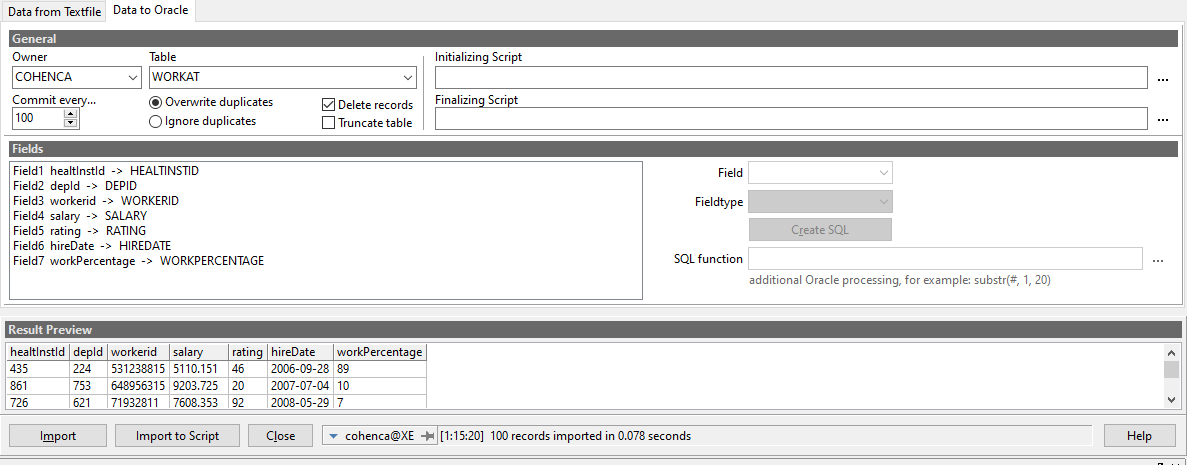
יצירת csv באמצעות excel:



### ייבוא:

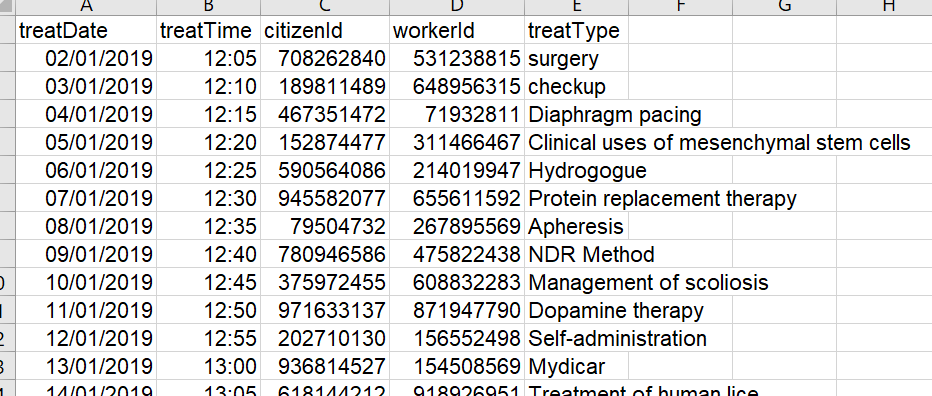


### התאמת הנתונים לטבלה:

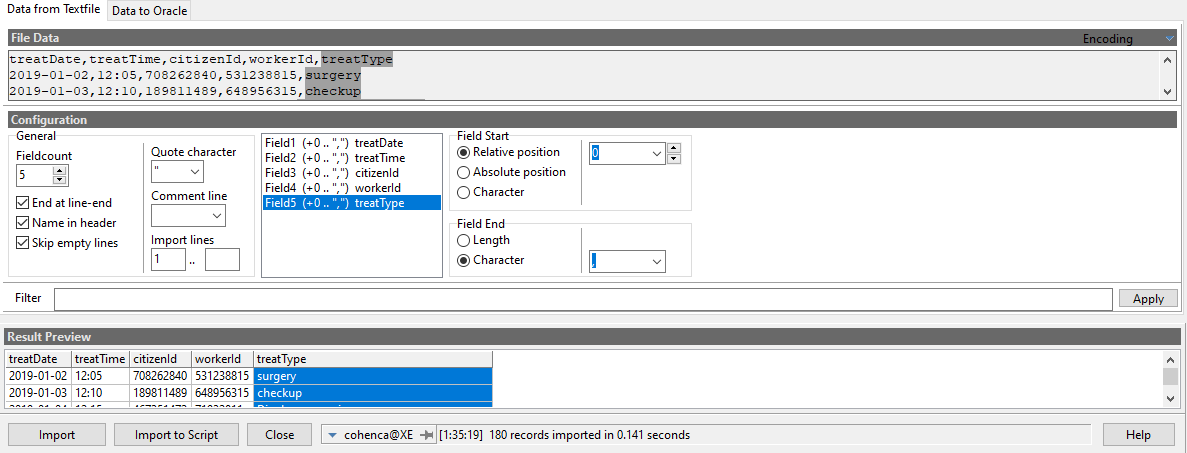


## טבלת treatedBy:

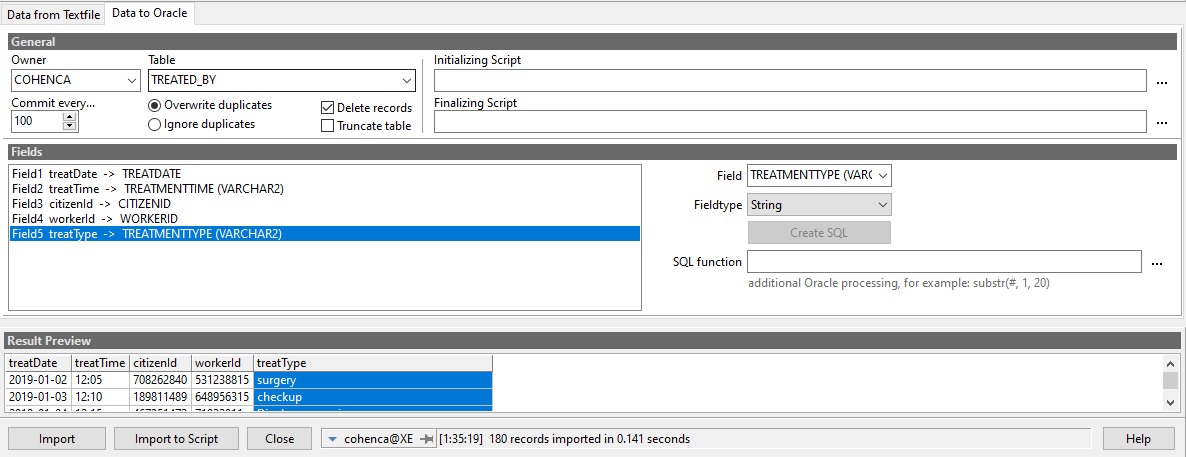
יצירת csv באמצעות excel:



### ייבוא:



### התאמת הנתונים לטבלה:



נרמול

הסכמה מנורמלת בצורת 3NF משום שהיא עומדת בכללים הבאים:

1NF- אין שדה מורכב – כלו' כל שדה בטבלא מכיל את יחידת המידע הבסיסית לגבי המידע באותו שדה.

2NF- השדות תלויים בכל המפתח – כלו' גם כאשר יש מפתח שמורכב מיותר משדה אחד כל השדות באותה טבלא תלויים בכל המפתח ולא רק בחלקו.

3NF- אין שדות שתלויים בשדה שאיננו מפתח.