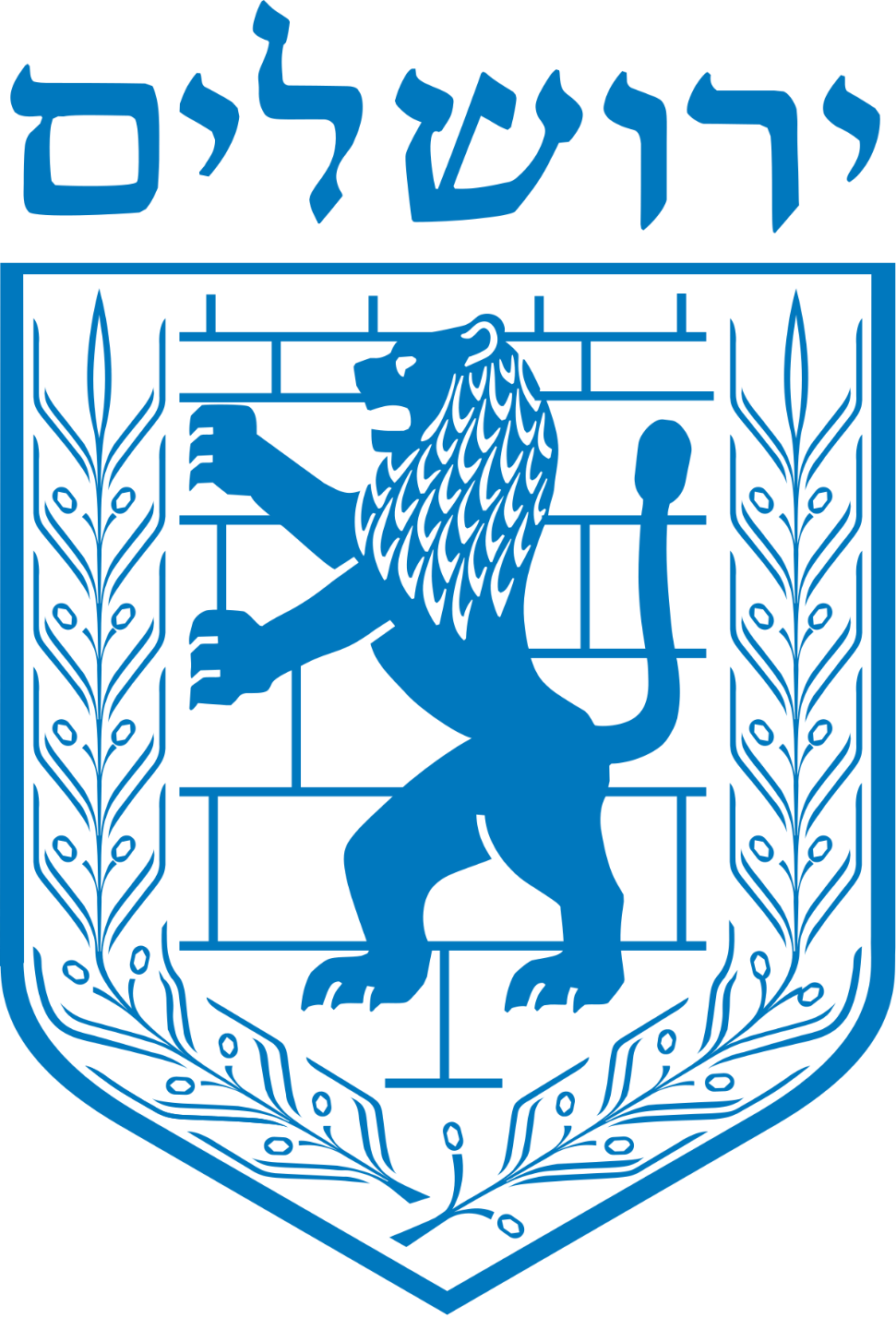
|  |
| --- |
|  |

מיני פרויקט בבסיסי נתונים – עירייה



**נועם שוובר – 214120032**

**נריה ניימן –214186801**

# תוכן עניינים

תוכן

[תוכן עניינים 2](#_Toc101972809)

[חלק א' 4](#_Toc101972810)

[דיאגרמת ER 5](#_Toc101972811)

[דיאגרמת DSD (יחסים) 5](#_Toc101972812)

[הסבר הישויות 6](#_Toc101972813)

[חלק ב' 15](#_Toc101972814)

[אכלוס הטבלאות 15](#_Toc101972815)

[שאילתות מיוחדות 40](#_Toc101972816)

[שאילתא מס' 1 40](#_Toc101972817)

[שאילתא מס' 2 41](#_Toc101972818)

[שאילתא מס' 3 42](#_Toc101972819)

[שאילתא מס' 4 43](#_Toc101972820)

[שאילתא מס' 5 44](#_Toc101972821)

[שאילתא מס' 6 45](#_Toc101972822)

[שאילתא מס' 7 46](#_Toc101972823)

[שאילתא מס' 8 47](#_Toc101972824)

[אינדקסים 48](#_Toc101972825)

# תוכן עניינים

תוכן

[תוכן עניינים 2](#_Toc101972809)

[חלק א' 4](#_Toc101972810)

[דיאגרמת ER 5](#_Toc101972811)

[דיאגרמת DSD (יחסים) 5](#_Toc101972812)

[הסבר הישויות 6](#_Toc101972813)

[חלק ב' 15](#_Toc101972814)

[אכלוס הטבלאות 15](#_Toc101972815)

[שאילתות מיוחדות 40](#_Toc101972816)

[שאילתא מס' 1 40](#_Toc101972817)

[שאילתא מס' 2 41](#_Toc101972818)

[שאילתא מס' 3 42](#_Toc101972819)

[שאילתא מס' 4 43](#_Toc101972820)

[שאילתא מס' 5 44](#_Toc101972821)

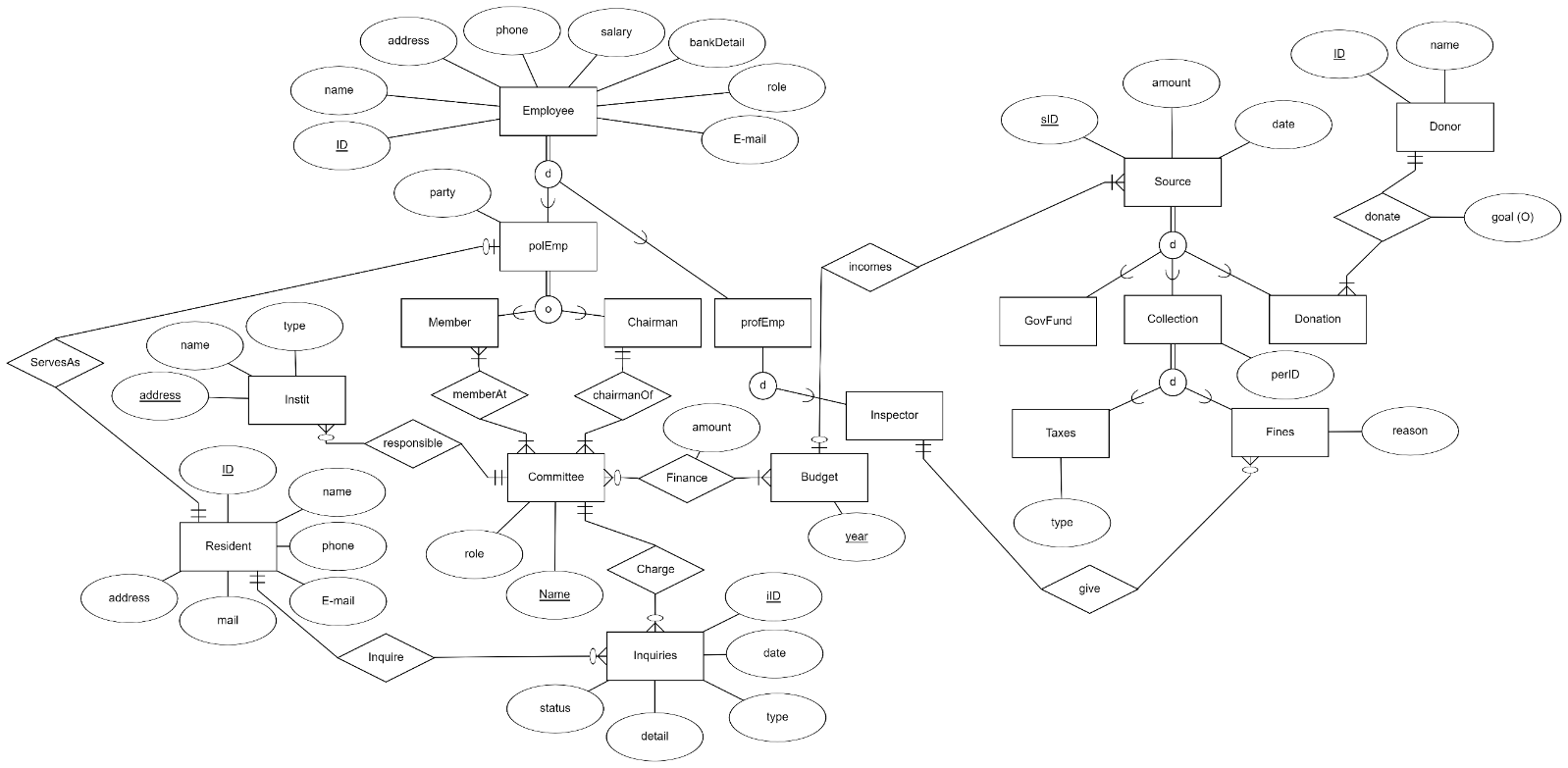
[שאילתא מס' 6 45](#_Toc101972822)

[שאילתא מס' 7 46](#_Toc101972823)

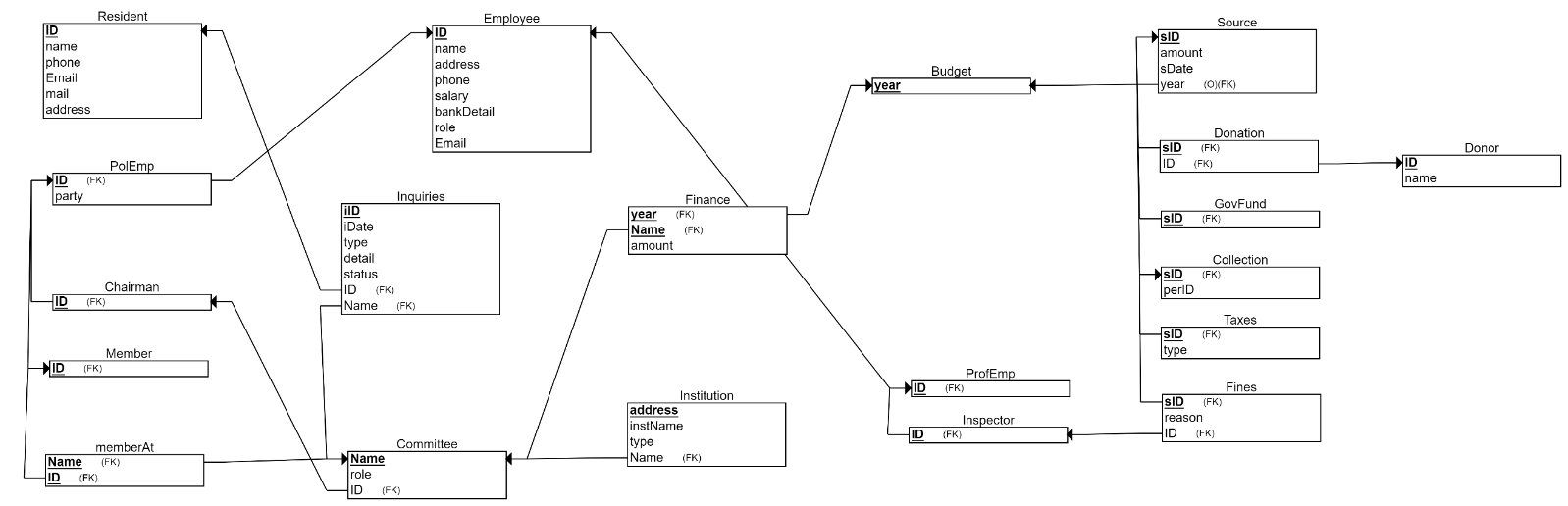
[שאילתא מס' 8 47](#_Toc101972824)

[אינדקסים 48](#_Toc101972825)

**חלק א'**

**דיאגרמת ER**

**דיאגרמת DSD (יחסים)**

****

תמונה שמכילה טקסט, מפה, מקורה

התיאור נוצר באופן אוטומטי

## הסבר הישויות

1. **Employee**

מייצג עובד כללי – בין אם גר בעירייה, בין אם לא.

תכונות:

* ID – ת"ז (מפתח)
* Name – שם
* Address – כתובת
* Phone – מספר פלאפון
* Salary – משכורת
* Bank Details – פרטי בנק
* Role – תפקיד
* Email – כתובת מייל

Create Table:

CREATE TABLE Employee

(

ID INT NOT NULL,

name VARCHAR(30) NOT NULL,

address VARCHAR(30) NOT NULL,

phone VARCHAR(11) NOT NULL,

salary INT NOT NULL,

bankDetail VARCHAR(20) NOT NULL,

role VARCHAR(20) NOT NULL,

Email VARCHAR(30) NOT NULL,

PRIMARY KEY (ID)

);

1. **ProfEmp**

מייצג עובד מקצועי – יכול לגור בעירייה ויכול שלא. יורש מעובד כללי.

תכונות:

* ID – ת"ז (מפתח, מפתח זר של Employee)

CREATE TABLE profEmp

(

ID INT NOT NULL,

PRIMARY KEY (ID),

FOREIGN KEY (ID) REFERENCES Employee(ID)

);

1. **Resident**

מייצג תושב בעירייה.  
תכונות:

* ID – ת"ז (מפתח)
* Name – שם
* Phone – מספר פלאפון
* Email – כתובת מייל
* Mail – כתובת דואר
* Address – כתובת

Create Table:

CREATE TABLE Resident

(

ID INT NOT NULL,

name VARCHAR(30) NOT NULL,

phone VARCHAR(11) NOT NULL,

Email VARCHAR(30) NOT NULL,

mail VARCHAR(30) NOT NULL,

address VARCHAR(30) NOT NULL,

PRIMARY KEY (ID)

);

1. **PolEmp**

מייצג עובד פוליטי – בהכרח גר בעירייה. יורש מעובד כללי.  
תכונות:

* ID – ת"ז (מפתח, מפתח זר של Employee ושל Resident)
* Party – מפלגה

Create Table:

CREATE TABLE polEmp

(

party VARCHAR(20) NOT NULL,

ID INT NOT NULL,

PRIMARY KEY (ID),

FOREIGN KEY (ID) REFERENCES Employee(ID),

FOREIGN KEY (ID) REFERENCES Resident(ID)

);

1. **Budget**

מייצג תקציב שנתי.  
תכונות:

* year – שנת התקציב (מפתח)

Create Table:

CREATE TABLE Budget

(

year INT NOT NULL,

PRIMARY KEY (year)

);

1. **Source**

מייצג מקור כלכלי לתקציב.   
תכונות:

* sID – מספר סידורי של התקציב (מפתח)
* sDate – תאריך קבלת המקור.
* Amount – כמות המקור.
* Year – שנת התקציב (מפתח זר של Budget)

Create Table:

CREATE TABLE Source

(

amount INT NOT NULL,

sDate DATE NOT NULL,

sID INT NOT NULL,

year INT,

PRIMARY KEY (sID),

FOREIGN KEY (year) REFERENCES Budget(year)

);

1. **Collection**

מייצג אסיפת כסף כלשהי מאדם (לאו דווקא מתושב העירייה). יורש מSource.  
תכונות:

* sID – מספר סידורי של התקציב (מפתח, מפתח זר של Source)
* ID – מייצג את תעודת הזהות של האדם ממנו נאסף הכסף. (לא מקושר כיוון שיכול להיות מחוץ לעירייה)

Create Table:

CREATE TABLE Collection

(

ID INT NOT NULL,

sID INT NOT NULL,

PRIMARY KEY (sID),

FOREIGN KEY (sID) REFERENCES Source(sID)

);

1. **GovFund**

מייצג תקצוב מהממשלה. יורש מSource.  
תכונות:

* sID – מספר סידורי של התקציב (מפתח, מפתח זר של Source)

Create Table:

CREATE TABLE GovFund

(

sID INT NOT NULL,

PRIMARY KEY (sID),

FOREIGN KEY (sID) REFERENCES Source(sID)

);

1. **Taxes**

מייצג מיסים שנאספו. יורש מCollection.  
תכונות:

* sID – מספר סידורי של התקציב (מפתח, מפתח זר של Collection)
* Type – סוג מסים.

Create Table:

CREATE TABLE Taxes

(

type VARCHAR(10) NOT NULL,

sID INT NOT NULL,

PRIMARY KEY (sID),

FOREIGN KEY (sID) REFERENCES Collection(sID)

);

1. **Donor**

מייצג תורם לעירייה.  
תכונות:

* ID – ת"ז (מפתח)
* Name – שם.

Create Table:

CREATE TABLE Donor

(

name VARCHAR(30) NOT NULL,

ID INT NOT NULL,

PRIMARY KEY (ID)

);

1. **Donation**

מייצג תרומה. יורש מSource.  
תכונות:

* sID – מספר סידורי של התקציב (מפתח, מפתח זר של Source)
* dID – ת"ז של תורם (מפתח זר של Donor)

Create Table:

CREATE TABLE Donation

(

sID INT NOT NULL,

dID INT NOT NULL,

PRIMARY KEY (sID),

FOREIGN KEY (sID) REFERENCES Source(sID),

FOREIGN KEY (dID) REFERENCES Donor(ID)

);

1. **Inspector**

מייצג פקח בעירייה. יורש מעובד מקצועי.  
תכונות:

* ID – ת"ז (מפתח, מפתח זר של Employee)

Create Table:

CREATE TABLE Inspector

(

ID INT NOT NULL,

PRIMARY KEY (ID),

FOREIGN KEY (ID) REFERENCES profEmp(ID)

);

1. **Fines**

מייצג קנס שנאסף (בעקבות עבירה). יורש מCollection.  
תכונות:

* sID – מספר סידורי של התקציב (מפתח, מפתח זר של Source)
* Reason – סיבה לקבלת הקנס.
* iID – ת"ז של הפקח (מקושר לInspector)

Create Table:

CREATE TABLE Fines

(

reason VARCHAR(40) NOT NULL,

sID INT NOT NULL,

iID INT NOT NULL,

PRIMARY KEY (sID),

FOREIGN KEY (sID) REFERENCES Collection(sID),

FOREIGN KEY (iID) REFERENCES Inspector(ID)

);

1. **Member**

מייצג חבר וועדה – יורש מעובד פוליטי.  
תכונות:

* ID – ת"ז (מפתח, מפתח זר של PolEmployee)

Create Table:

CREATE TABLE Member

(

ID INT NOT NULL,

PRIMARY KEY (ID),

FOREIGN KEY (ID) REFERENCES polEmp(ID)

);

1. **Chairman**

מייצג יו"ר ועדה. יורש מעובד פוליטי.  
תכונות:

* ID – ת"ז (מפתח, מפתח זר של Employee ושל Resident)

Create Table:

CREATE TABLE Chairman

(

ID INT NOT NULL,

PRIMARY KEY (ID),

FOREIGN KEY (ID) REFERENCES polEmp(ID)

);

1. **Committee**

מייצג ועדה של נבחרי הציבור בעירייה. (לדוג' ועדת חינוך, ועדת תרבות)  
תכונות:

* Name – שם הועדה (מפתח)
* Role – תפקיד הוועדה [בפירוט]
* ID – ת"ז של יו"ר הוועדה (מפתח זר של Chairman)

Create Table:

CREATE TABLE Committee

(

Name VARCHAR(30) NOT NULL,

role VARCHAR(30) NOT NULL,

ID INT NOT NULL,

PRIMARY KEY (Name),

FOREIGN KEY (ID) REFERENCES Chairman(ID)

)

1. **Institution**

מייצג מוסד בעירייה – לכל מוסד יש וועדה כלשהי שדואגת לו.   
תכונות:

* Address – כתובת (מפתח)
* instName – שם המוסד
* Type – סוג המוסד.
* commitName – שם הוועדה המטפלת. (מפתח זר של Committee)

Create Table:

CREATE TABLE Institution

(

address VARCHAR(30) NOT NULL,

instName VARCHAR(30) NOT NULL,

type CHAR(10) NOT NULL,

commitName VARCHAR(30) NOT NULL,

PRIMARY KEY (address),

FOREIGN KEY (commitName) REFERENCES Committee(Name)

);

1. **MemberAt**

מייצג קשר בין חברי וועדה לוועדה.   
תכונות:

* Name – שם הוועדה (חלק ממפתח, מפתח זר של Committee)
* ID – ת"ז של חבר הוועדה (חלק ממפתח, מפתח זר של Member)

Create Table:

CREATE TABLE memberAt

(

Name VARCHAR(30) NOT NULL,

ID INT NOT NULL,

PRIMARY KEY (Name, ID),

FOREIGN KEY (Name) REFERENCES Committee(Name),

FOREIGN KEY (ID) REFERENCES Member(ID)

);

1. **Inquiries**

מייצג פנייה מתושב העירייה – כל פנייה מטופלת על ידי ועדה כלשהי.   
תכונות:

* iID – המספר הסידורי של הפנייה (מפתח)
* iDate – תאריך פתיחת הפנייה.
* Type – סוג הפנייה.
* Details – פרטי הפנייה.
* Status – סטטוס הפנייה.
* ID – ת"ז התושב הפונה (מפתח זר של Resident)
* commitName – שם הוועדה (מפתח זר של Commitee)

Create Table:

CREATE TABLE Inquiries

(

iID INT NOT NULL,

iDate DATE NOT NULL,

type VARCHAR(10) NOT NULL,

details VARCHAR(50) NOT NULL,

status CHAR(1) NOT NULL,

ID INT NOT NULL,

commitName VARCHAR(30) NOT NULL,

PRIMARY KEY (iID),

FOREIGN KEY (ID) REFERENCES Resident(ID),

FOREIGN KEY (commitName) REFERENCES Committee(Name)

);

1. **Finance**

מייצג תקצוב שנתי לוועדה, לפי וועדה ושנה.  
תכונות:

* year – שנת התקציב (חלק ממפתח, מפתח זר של Budget).
* Name – שם הוועדה (חלק ממפתח, מפתח זר של Committee).
* Amount – כמות התקציב מהשנה לוועדה.

Create Table:

CREATE TABLE Finance

(

amount INT NOT NULL,

year INT NOT NULL,

Name VARCHAR(30) NOT NULL,

PRIMARY KEY (year, Name),

FOREIGN KEY (year) REFERENCES Budget(year),

FOREIGN KEY (Name) REFERENCES Committee(Name)

);

# חלק ב'

## אכלוס הטבלאות

1. תמונה שמכילה טקסט

   התיאור נוצר באופן אוטומטי**תמונה שמכילה שולחן

   התיאור נוצר באופן אוטומטיתמונה שמכילה שולחן

   התיאור נוצר באופן אוטומטי****Employee** – הכנסה באמצעות Data Generator
2. תמונה שמכילה טקסט

   התיאור נוצר באופן אוטומטיתמונה שמכילה טקסט

   התיאור נוצר באופן אוטומטי**תמונה שמכילה טקסט

   התיאור נוצר באופן אוטומטי**תמונה שמכילה טקסט

   התיאור נוצר באופן אוטומטי**ProfEmp** – הכנסה באמצעות Text Importer
3. **תמונה שמכילה שולחן

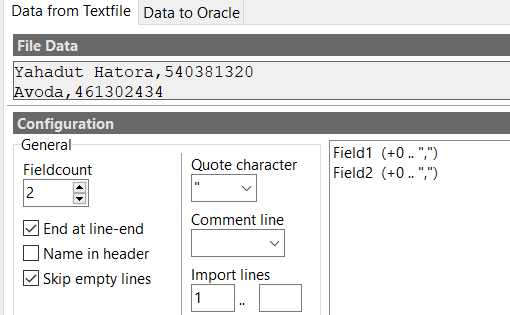
   התיאור נוצר באופן אוטומטיResident** – הכנסה באמצעות Data Generator

**תמונה שמכילה טקסט

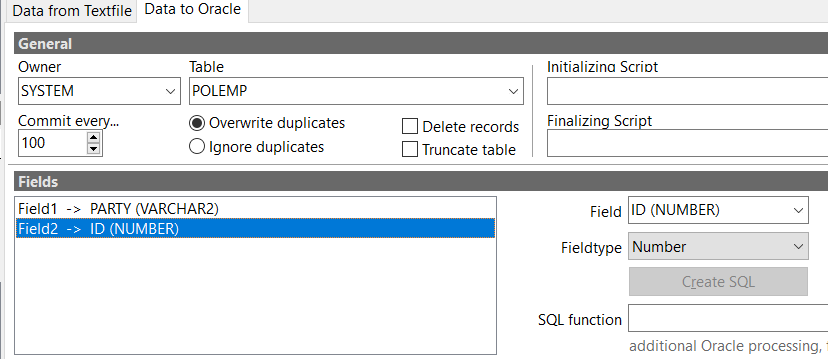
התיאור נוצר באופן אוטומטיתמונה שמכילה טקסט

התיאור נוצר באופן אוטומטי=**

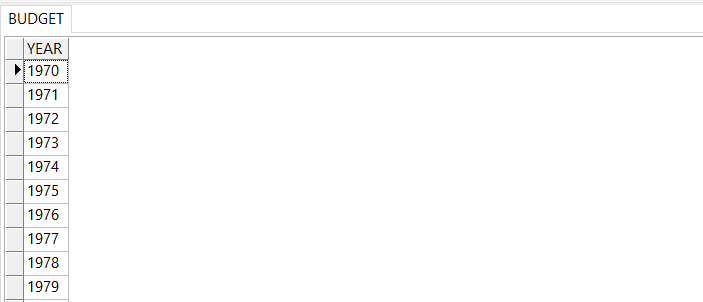
1. **תמונה שמכילה טקסט

   התיאור נוצר באופן אוטומטיPolEmp** – הכנסה באמצעות Text Importer

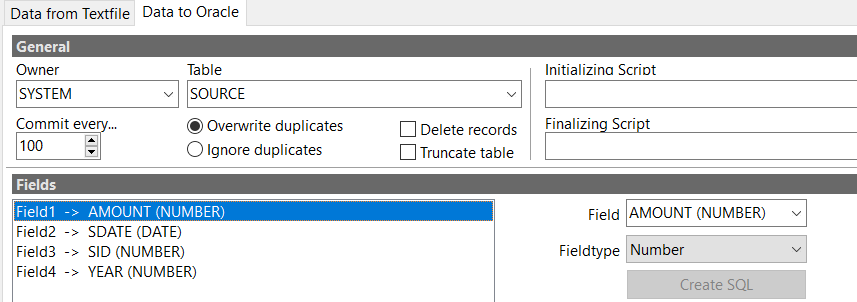
**תמונה שמכילה טקסט

התיאור נוצר באופן אוטומטי**

1. **תמונה שמכילה טקסט

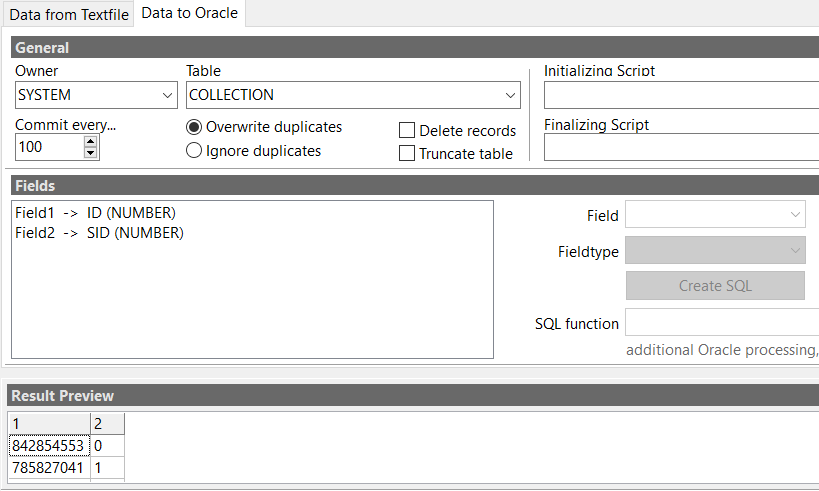
   התיאור נוצר באופן אוטומטיתמונה שמכילה טקסט

   התיאור נוצר באופן אוטומטיBudget** – הכנסה באמצעות Data Generator
2. **תמונה שמכילה שולחן

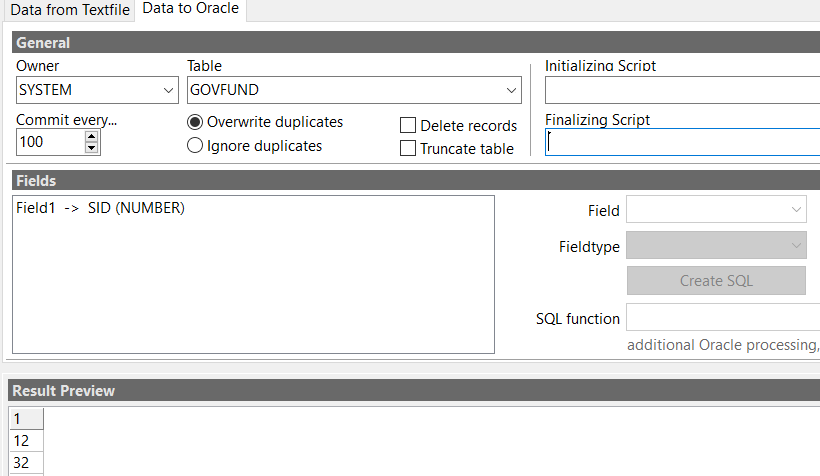
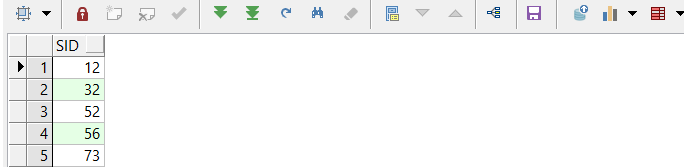
   התיאור נוצר באופן אוטומטיתמונה שמכילה טקסט

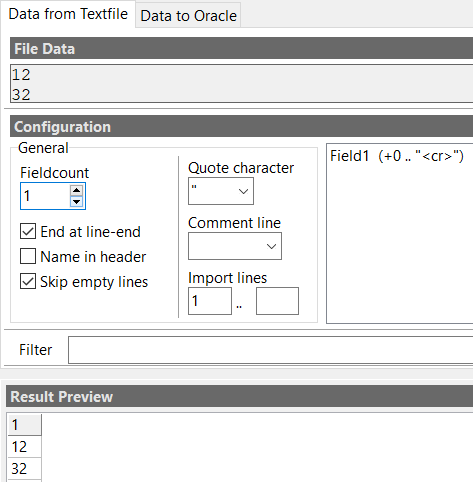
   התיאור נוצר באופן אוטומטיתמונה שמכילה טקסט

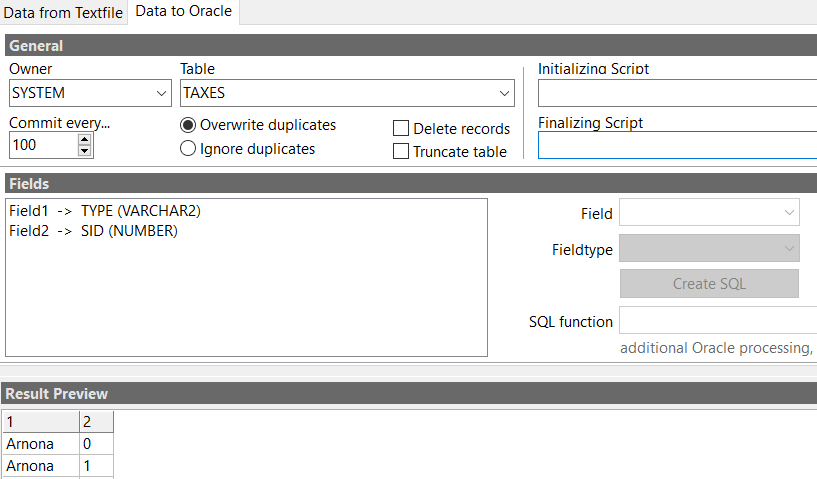
   התיאור נוצר באופן אוטומטיSource** – הכנסה באמצעות Text Importer
3. **תמונה שמכילה טקסט

   התיאור נוצר באופן אוטומטיתמונה שמכילה טקסט

   התיאור נוצר באופן אוטומטיתמונה שמכילה טקסט

   התיאור נוצר באופן אוטומטיCollection** –הכנסה באמצעות Text Importer
4. ****תמונה שמכילה שולחן

   התיאור נוצר באופן אוטומטי**GovFund** – הכנסה באמצעות Text Importer
5. **תמונה שמכילה טקסט

   התיאור נוצר באופן אוטומטיתמונה שמכילה טקסט

   התיאור נוצר באופן אוטומטי**תמונה שמכילה טקסט

   התיאור נוצר באופן אוטומטי**Taxes** – הכנסה באמצעות Text Importer
6. **תמונה שמכילה טקסט

   התיאור נוצר באופן אוטומטיתמונה שמכילה שולחן

   התיאור נוצר באופן אוטומטיתמונה שמכילה שולחן

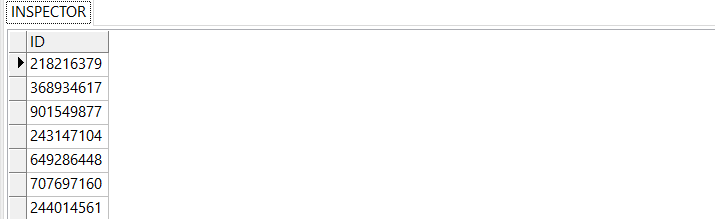
   התיאור נוצר באופן אוטומטיDonor** – הכנסה באמצעות Data Generator
7. **תמונה שמכילה טקסט

   התיאור נוצר באופן אוטומטיתמונה שמכילה טקסט

   התיאור נוצר באופן אוטומטיתמונה שמכילה טקסט

   התיאור נוצר באופן אוטומטי**תמונה שמכילה טקסט

   התיאור נוצר באופן אוטומטי**Donation** – הכנסה באמצעות Text Importer
8. **תמונה שמכילה טקסט

   התיאור נוצר באופן אוטומטיתמונה שמכילה שולחן

   התיאור נוצר באופן אוטומטיInspector** – הכנסה באמצעות Data Generator

1. **תמונה שמכילה טקסט

   התיאור נוצר באופן אוטומטיתמונה שמכילה טקסט

   התיאור נוצר באופן אוטומטיתמונה שמכילה טקסט

   התיאור נוצר באופן אוטומטי**תמונה שמכילה טקסט

   התיאור נוצר באופן אוטומטי**Fines** – הכנסה באמצעות Text Importer
2. **תמונה שמכילה טקסט

   התיאור נוצר באופן אוטומטיתמונה שמכילה טקסט

   התיאור נוצר באופן אוטומטיMember** – הכנסה באמצעות Insert
3. **תמונה שמכילה שולחן

   התיאור נוצר באופן אוטומטיתמונה שמכילה טקסט

   התיאור נוצר באופן אוטומטיChairman** – הכנסה באמצעות Insert

1. **תמונה שמכילה טקסט, שולחן

   התיאור נוצר באופן אוטומטיCommittee** – הכנסה באמצעות Insert

**תמונה שמכילה טקסט

התיאור נוצר באופן אוטומטי**

1. **תמונה שמכילה שולחן

   התיאור נוצר באופן אוטומטיתמונה שמכילה שולחן

   התיאור נוצר באופן אוטומטיתמונה שמכילה טקסט

   התיאור נוצר באופן אוטומטיInstitution** – הכנסה באמצעות Data Generator

1. **תמונה שמכילה שולחן

   התיאור נוצר באופן אוטומטיתמונה שמכילה טקסט

   התיאור נוצר באופן אוטומטיMemberAt** – הכנסה באמצעות Insert

1. **תמונה שמכילה טקסט

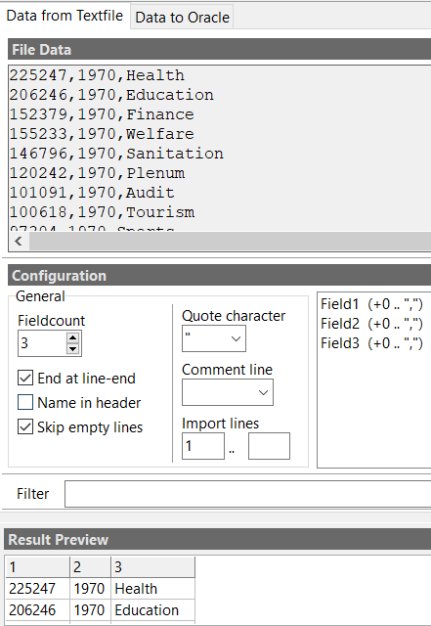
   התיאור נוצר באופן אוטומטיתמונה שמכילה שולחן

   התיאור נוצר באופן אוטומטיתמונה שמכילה טקסט

   התיאור נוצר באופן אוטומטיInquiries** – הכנסה באמצעות Data Generator
2. תמונה שמכילה טקסט

   התיאור נוצר באופן אוטומטיתמונה שמכילה טקסט

   התיאור נוצר באופן אוטומטי**תמונה שמכילה טקסט

   התיאור נוצר באופן אוטומטי****Finance** – הכנסה באמצעות Text Importer

## שאילתות מיוחדות

שאילתא מס' 1

**תיאור**: אם נקצץ 20% ממשכורות סוגי השונים העובדים או 1% מתקציבי הוועדות, אילו ועדות וסוגי עובדים שלהם נקצץ בתקציב ייתן לנו יותר?  
**קוד**:

SELECT \*

FROM ((SELECT EROLE AS NAME, SUM(SALARY) - SUM(SALARY \* 0.8) AS SAVING

FROM EMPLOYEE

GROUP BY EROLE)

UNION

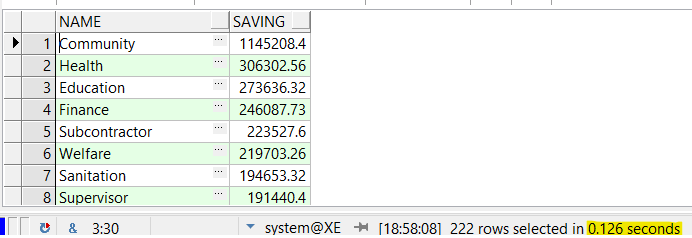
(SELECT C.CNAME AS NAME, SUM(F.AMOUNT) - SUM(AMOUNT \* 0.99) AS SAVING

FROM COMMITTEE C, FINANCE F

WHERE F.NAME = C.CNAME

GROUP BY C.CNAME)) T

ORDER BY SAVING DESC;

 **תוצאה**:

**זמן ביצוע**: 0.126 שניות

שאילתא מס' 2 **תיאור**: בדיקת פרטי 400 תושבים בעייתיים שלא שילמו מיסים כלל לעירייה או ששילמו יותר קנסות ממיסים.  
  
  
**קוד**:

SELECT ID, PHONE, EMAIL

FROM RESIDENT R

WHERE R.ID NOT IN (SELECT ID

FROM COLLECTION C, TAXES T

WHERE C.SID = T.SID)

OR ((SELECT SUM(AMOUNT) FROM COLLECTION CL NATURAL JOIN SOURCE NATURAL JOIN FINES WHERE CL.ID = R.ID)

-

(SELECT SUM(AMOUNT) FROM COLLECTION CL NATURAL JOIN SOURCE NATURAL JOIN TAXES WHERE CL.ID = R.ID))

> 0

ORDER BY R.ADDRESS

FETCH FIRST 400 ROWS ONLY;

תמונה שמכילה שולחן

התיאור נוצר באופן אוטומטי **תוצאה**:

**זמן ביצוע**: 0.305 שניות

שאילתא מס' 3  
  
**תיאור**: בדיקת פרטי האזרחים ששלחו פניות לוועדות שקיבלו הכי הרבה פניות.  
  
**קוד**:

SELECT NAME, EMAIL, MAIL

FROM RESIDENT R

WHERE ID in (SELECT I.ID

FROM COMMITTEE C, INQUIRIES I

WHERE C.CNAME = I.COMMITNAME

AND C.CNAME IN (SELECT CNAME

FROM COMMITTEE C, INQUIRIES I

WHERE C.CNAME = I.COMMITNAME

GROUP BY CNAME

HAVING COUNT(\*) = (SELECT MAX(A)

FROM (SELECT COUNT(\*) AS A

FROM COMMITTEE C, INQUIRIES I

WHERE C.CNAME = I.COMMITNAME

GROUP BY CNAME) T)));

תמונה שמכילה שולחן

התיאור נוצר באופן אוטומטי **תוצאה**:

**זמן ביצוע**: 0.297 שניות

שאילתא מס' 4  
  
**תיאור**: פרטי התושבים שקיבלו קנסות או שילמו מיסים, אך לא גם קיבלו קנסות וגם שילמו מיסים.  
  
**קוד**:

SELECT ID, NAME, ADDRESS

FROM RESIDENT

WHERE ID IN (((SELECT R.ID FROM RESIDENT R, COLLECTION C, SOURCE S, FINES F

WHERE C.ID = R.ID AND C.SID = S.SID AND F.SID = C.SID)

UNION

(SELECT R.ID FROM RESIDENT R, COLLECTION C, SOURCE S, TAXES T

WHERE C.ID = R.ID AND C.SID = S.SID AND T.SID = C.SID))

EXCEPT

(SELECT ID FROM RESIDENT R

WHERE ID IN ((SELECT ID

FROM COLLECTION C, TAXES T

WHERE C.SID = T.SID)

INTERSECT

(SELECT ID

FROM COLLECTION C, FINES F

WHERE C.SID = F.SID))));

תמונה שמכילה שולחן

התיאור נוצר באופן אוטומטי **תוצאה**:

**זמן ביצוע**: 0.181 שניות

שאילתא מס' 5  
  
**תיאור**: טבלת סיכום כל סוגי המוסדות לפי התקציב שלהן, כל עוד הוועדה שאחראית על ועדה זו קיבלה יותר מ-10 פניות.  
  
**קוד**:

SELECT I.TYPE, SUM(F.AMOUNT)

FROM INSTITUTION I, COMMITTEE C, FINANCE F, INQUIRIES INQ

WHERE I.COMMITNAME = C.CNAME AND F.NAME = C.CNAME AND INQ.COMMITNAME = C.CNAME

GROUP BY I.TYPE

HAVING COUNT(INQ.IID) > 10

ORDER BY SUM(F.AMOUNT) DESC;

תמונה שמכילה טקסט, שולחן

התיאור נוצר באופן אוטומטי **תוצאה**:

**זמן ביצוע**: 0.329 שניות

שאילתא מס' 6  
  
**תיאור**: בדיקה אילו אנשים (לא בהכרח תושבי העירייה) הכניסו לעירייה הכי הרבה כספים, כל עוד ממוצע הכספים שהם הביאו לעירייה גדול מ-1000.  
  
**קוד**:

SELECT ID, SUM(AMOUNT)

FROM ((SELECT F.SID, S.AMOUNT, F.REASON AS TYPE, C.ID AS ID

FROM FINES F, COLLECTION C, SOURCE S

WHERE F.SID = C.SID AND C.SID = S.SID)

UNION ALL

(SELECT T.SID, S.AMOUNT, T.TYPE AS TYPE, C.ID AS ID

FROM TAXES T, COLLECTION C, SOURCE S

WHERE T.SID = C.SID AND C.SID = S.SID)

UNION ALL

(SELECT D.SID, S.AMOUNT, 'Donation' AS TYPE, D.DID AS ID

FROM DONATION D, SOURCE S

WHERE D.SID = S.SID)) T

GROUP BY ID

HAVING AVG(AMOUNT) > 1000;

תמונה שמכילה טקסט, שולחן

התיאור נוצר באופן אוטומטי **תוצאה**:

**זמן ביצוע**: 0.281 שניות

**עכשיו שתי שאילתות עם המון נתונים, כדי לבדוק גם שאילתות בסדרי גודל של שניות**

שאילתא מס' 7  
  
**תיאור**: בדיקת מהו אחוז התקציב של כל מוסד עבור כל שנה מסך התקציב הכללי, ממויין לפי שמות בסדר אלפביתי.  
   
**קוד**:

SELECT INS.INSTNAME, F.YEAR,   
SUM(F.AMOUNT)/(SELECT SUM(AMOUNT) FROM FINANCE WHERE YEAR = F.YEAR) AS PERCENT\_OF\_BUDGET

FROM INSTITUTION INS, FINANCE F, COMMITTEE C

WHERE INS.COMMITNAME = C.CNAME AND F.NAME = C.CNAME

GROUP BY INS.INSTNAME, F.YEAR

ORDER BY INSTNAME;

תמונה שמכילה שולחן

התיאור נוצר באופן אוטומטי **תוצאה**:

**זמן ביצוע**: 13.481 שניות

שאילתא מס' 8  
  
**תיאור**: פורים מגיע, ולו נרצה לבצע משחק וגמד ענק בין כל תושבי העירייה, יש שאילתא לביצוע הגרלת זוגות אוטומטי 😊

**קוד**:

SELECT R1.NAME AS NAME1, R1.ADDRESS AS ADDRESS1, R2.NAME AS NAME2, R2.ADDRESS AS ADDRESS2

FROM (SELECT ID, NAME, ADDRESS, ROW\_NUMBER() OVER (ORDER BY id) RN FROM (SELECT \*

FROM RESIDENT

ORDER BY ID

FETCH FIRST 2500 ROWS ONLY) T) R1,

(SELECT ID, NAME, ADDRESS, ROW\_NUMBER() OVER (ORDER BY id) RN FROM (SELECT \*

FROM RESIDENT

ORDER BY ID DESC

FETCH FIRST 2500 ROWS ONLY) T) R2

WHERE MOD(R1.RN + FLOOR(dbms\_random.value(1, 10)), (SELECT COUNT(\*) FROM RESIDENT)) = R2.RN;

ORDER BY R1.NAME;

**תמונה שמכילה שולחן

התיאור נוצר באופן אוטומטי תוצאה**:

**זמן ביצוע**: 3.892 שניות.

## אינדקסים

1. **האינדקס**:  
   create index inst\_indx on INSTITUTION ( commitname );  
     
   **שאילתא לדוגמה שמשתמשת בו**: [שאילתא מס' 7](#שאילתא7).  
     
     
   **השוואת זמנים**:  
   תמונה שמכילה שולחן

   התיאור נוצר באופן אוטומטי

|  |  |  |
| --- | --- | --- |
| **לפני** | **אחרי** | **שיפור?** |
| 13.481 | 11.890 | V |

**נימוק**: ניתן לראות שיפור עצום של שניה וחצי כיוון שאכן יש שימוש רב בעמודת ה-commitname בשאילתא שעליה אנחנו בודקים.

1. **האינדקס**:

create unique index clc\_indx on COLLECTION ( sid, id );  
  
  
**שאילתא לדוגמה שמשתמשת בו**: [שאילתא מס' 4](#שאילתא4).  
  
  
**השוואת זמנים**:  


|  |  |  |
| --- | --- | --- |
| **לפני** | **אחרי** | **שיפור?** |
| 0.181 | 0.131 | V |

**נימוק**: ניתן לראות שיפור בגלל שיש הרבה השוואות בשאילתא זו בין id ו-sid וכך ההשוואות מתבצעות מהר יותר.

1. **האינדקס**:

create index inq\_indx on INQUIRIES ( commitname );  
  
  
**שאילתא לדוגמה שמשתמשת בו**: [שאילתא מס' 3](#שאילתא3).  
  
  
**השוואת זמנים**:  
תמונה שמכילה שולחן

התיאור נוצר באופן אוטומטי

|  |  |  |
| --- | --- | --- |
| **לפני** | **אחרי** | **שיפור?** |
| 0.297 | 0.228 | V |

**נימוק**: ישנו שיפור בגלל שכשאנחנו מוסיפים להשוואה את commitname בבדיקה של ההשוואה של ה-Inquiries שלנו.

1. **האינדקס**:

create index inq\_indx on RESIDENT ( name );  
  
  
**שאילתא לדוגמה שמשתמשת בו**: [שאילתא מס' 8](#שאילתא8).  
  
  
**השוואת זמנים**:  


|  |  |  |
| --- | --- | --- |
| **לפני** | **אחרי** | **שיפור?** |
| 3.892 | 2.790 | V |

**נימוק**: בגלל המיון על פי name, כשאנו מבצעים אינדקס של NAME זה גורם למיון מהיר יותר, מה שבפועל הוביל שחסכנו כמעט שניה של זמן ריצה!

1. **האינדקס**:

create index inq\_indx on RESIDENT ( name );  
  
  
**שאילתא לדוגמה שמשתמשת בו**: [שאילתא מס' 5](#שאילתא5).  
  
  
**השוואת זמנים**:  
תמונה שמכילה שולחן

התיאור נוצר באופן אוטומטי

|  |  |  |
| --- | --- | --- |
| **לפני** | **אחרי** | **שיפור?** |
| 0.329 | 0.337 | X |

**נימוק**: כיוון ש-name הוא כבר חלק מהמפתח כבר יש אינדקס עבור הזוג (name, year), ובנוסף כמעט ואין שורות בטבלה זו ולכן האינדקס גם כן לא משפיע כל כך.

1. **האינדקס**:

create index src\_ne\_indx on SOURCE ( amount );  
  
  
**שאילתא לדוגמה שמשתמשת בו**: [שאילתא מס' 6](#שאילתא6).  
  
  
**השוואת זמנים**:  


|  |  |  |
| --- | --- | --- |
| **לפני** | **אחרי** | **שיפור?** |
| 0.281 | 0.348 | X |

**נימוק**: בגלל שהשימוש בשדה ה-amount הוא לא על מנת להשוות או למיון, אלא סתם בשביל חישוב הוא לא יוצר שום שיפור בזמן הריצה כמובן אלא רק סרבולים מיותרים.