

מיני פרויקט בבסיסי נתונים – עירייה



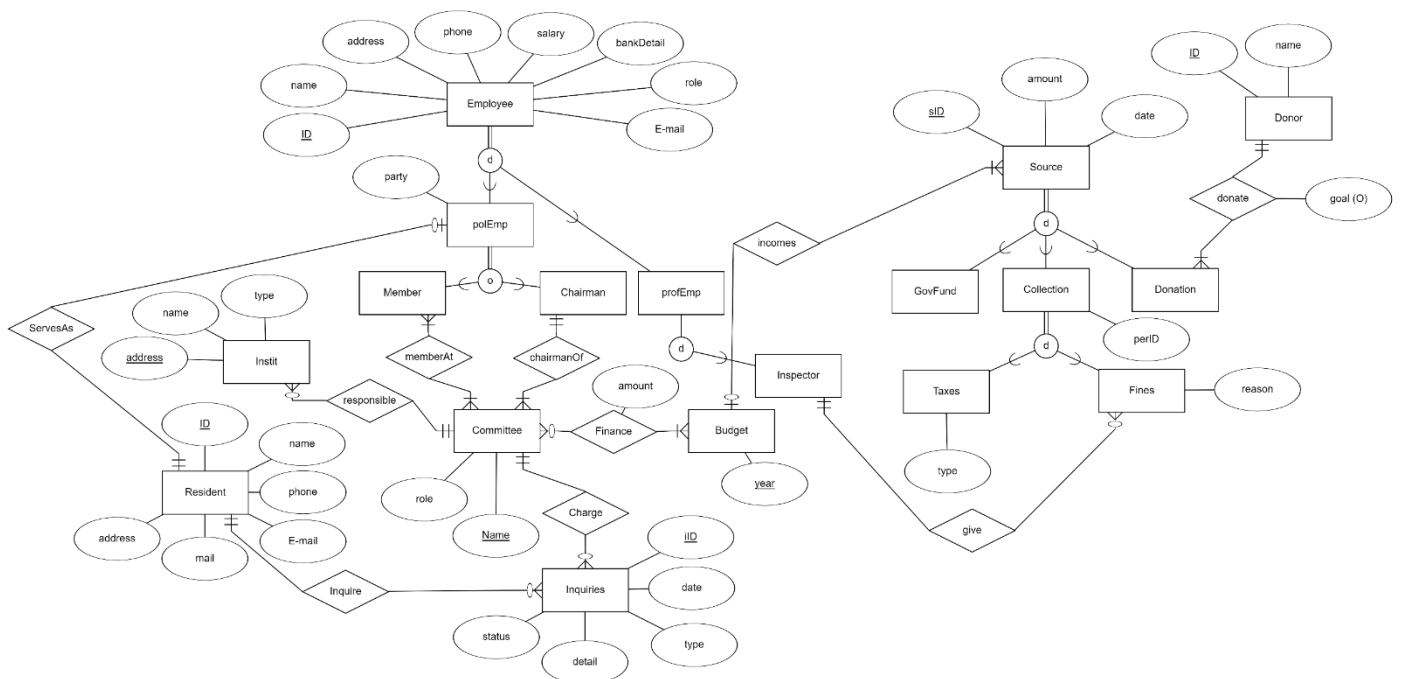
נועם שוובר – 214120032

נריה נימן – 214186801

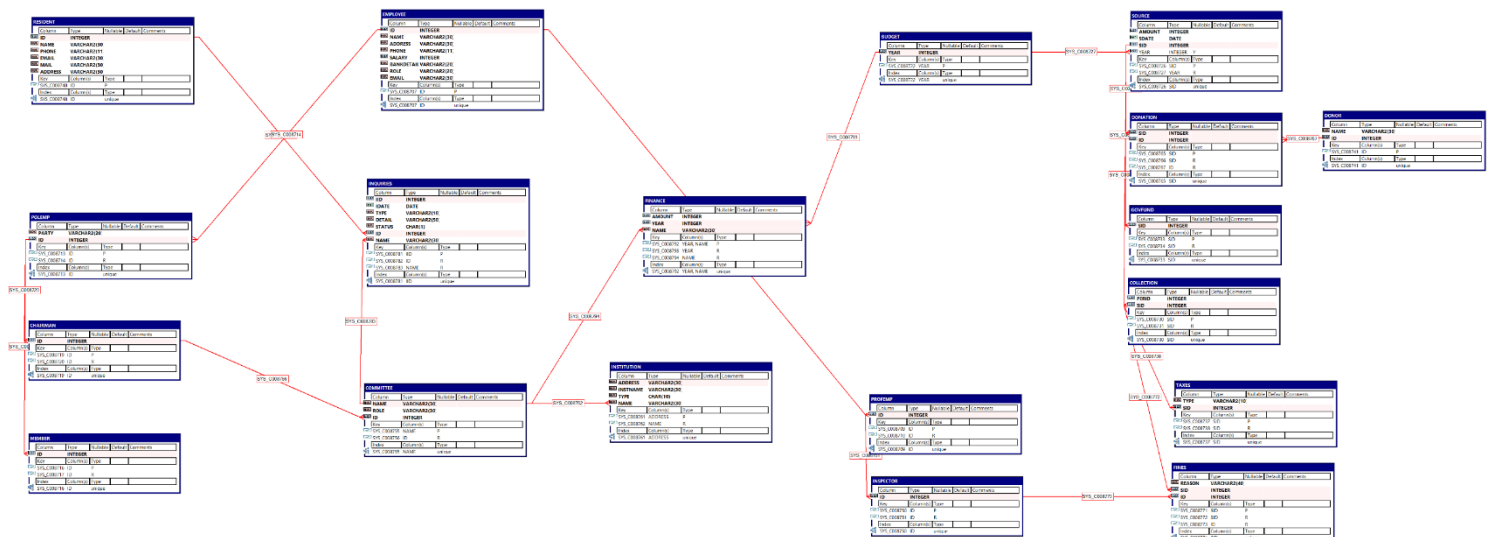
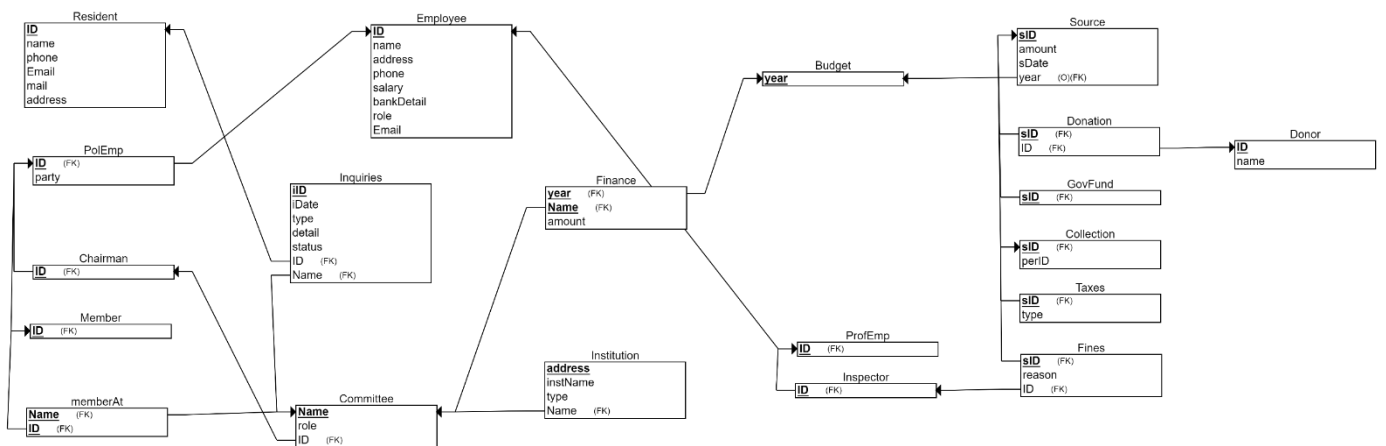
תוכן

2.....	תוכן עניינים
3.....	חלק א'.....
3.....	דיאגרמת ER.....
3.....	דיאגרמת DSD (יחסים).....
4.....	הסבר הישויות.....
13.....	חלק ב'.....
13.....	אכלוס הטבלאות.....
33.....	שאלות מיוחדות.....
33.....	שאלת מס' 1.....
34.....	שאלת מס' 2.....
35.....	שאלת מס' 3.....
36.....	שאלת מס' 4.....
37.....	שאלת מס' 5.....
38.....	שאלת מס' 6.....
39.....	שאלת מס' 7.....
40.....	שאלת מס' 8.....
41.....	אינדקסים.....
44.....	חלק ד'.....
44.....	תצפיות – Views.....
46.....	פונקציות – Functions.....
48.....	פרוצדורות – Procedures.....
50.....	טריגרים – Triggers.....

דיאגרמת ER



דיאגרמת DSD (יחסים)



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)
);
```

2. 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)
);
```

3. 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)
);
```

4. 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)
);
```

מייצג תקציב שנתי.

תכונות:

- year – שנת התקציב (מפתח)

:Create Table

```
CREATE TABLE Budget
(
    year INT NOT NULL,
    PRIMARY KEY (year)
);
```

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)
);
```

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)
);
```

8. 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)
);
```

9. Taxes

מייצג מיסים שנאספו. יורש מCollection.
תכונות:

- sID – מספר סידורי של התקציב (מפתח, מפתח זר של Collection)
- Type – סוג מיסים.

:Create Table

```
CREATE TABLE Taxes
(
    type VARCHAR(10) NOT NULL,
    sID INT NOT NULL,
    PRIMARY KEY (sID),
    FOREIGN KEY (sID) REFERENCES Collection(sID)
);
```

10. Donor

מייצג תורם לעירייה.
תכונות:

- ID – ת"ז (מפתח)
- Name – שם.

```
CREATE TABLE Donor
(
  name VARCHAR(30) NOT NULL,
  ID INT NOT NULL,
  PRIMARY KEY (ID)
);
```

Donation.11

מייצג תרומה. יורש Source.
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)
);
```

Inspector.12

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)
);
```


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)
);
```

Member.14

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)
);
```

Chairman.15

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)
);
```

Committee.16

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

- Name – שם הוועדה (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)
);
```

Institution.17

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)
);
```

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)
);
```

Inquiries.19

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 Committee)

: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)
);
```

Finance.20

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)
);
```

אנלוס הטבלאות

1. Employee – הנסה באמצעות Data Generator

EMPLOYEE

Owner

Table

Number of records

SYSTEM

EMPLOYEE

1000

Name	Type	Size	Data
ID	NUMBER		Random(100000000, 999999999)
NAME	VARCHAR2	30	FirstName + ' ' + LastName
ADDRESS	VARCHAR2	30	Address1
PHONE	VARCHAR2	11	['05'999999999]
SALARY	NUMBER		Random(5300, 530000)
BANKDETAIL	VARCHAR2	20	Random(0, 100000)
ROLE	VARCHAR2	20	[@@@@@@@@@@@@]
EMAIL	VARCHAR2	30	Email

EMPLOYEE

ID	NAME	ADDRESS	PHONE	SALARY	BANKDETAIL	ROLE	EMAIL
724907798	Ethan Ojeda	31 Fox	0522559872	230010	60074	LpGZafaGFS	ethan.o@tripwire.com
449210441	Annie Rio	1 Reilly Street	0581351311	172751	81440	uaCCLAUKfQ	annier@studiobproductions.com
398164800	Darren Whitwam	68 Coverdale Street	0508179530	421219	80010	DTrLwzjOQD	darrenw@coadvantageresources.e
414588106	Campbell Bloch	81 Lucien Street	0585755087	118685	24440	ylKkMDTOSC	cbloch@sweetproductions.uk
847079575	Gran Viterelli	867 Elwes Ave	0512316059	275610	21528	PoxlKmlnXy	gran.viterelli@nsd.com
530332245	Larenz Giraldo	64 Janssen Road	0543435642	460460	84384	nOBoQYBzKx	larenzg@trinityhomecare.be
611044062	Alfred Rhys-Davies	924 McGovern Ave	0562720773	508556	3803	RAdzHKKmiT	arhysd Davies@anworthmortgage.co
450344298	Chanté Downey	295 Folds Drive	0553580911	349253	12930	CMtTRYkomW	chant.downey@ibfh.com
152827729	Daryl Wilkinson	41st Street	0500306094	378832	99201	lVnJuHAPho	daryl.w@epiqsystems.com
723721031	Temuera Anderson	215 Patty Drive	0597425262	414298	94307	nYjnJsJwTJ	temuera.a@hfn.it
702137810	Boz Levin	53 Benicio Street	0505392767	199806	59674	WMxXMllUSD	boz.levin@mss.ca
139663330	Lari Diesel	51 Arquette Street	0543336217	138194	69586	voLqjFaNWM	lari.diesel@teamstudio.de
326974323	Wes Matarazzo	21 Cherepovets Drive	0551239962	335838	1740	EHSowdRZi	wes.matarazzo@qdatechnologies.

SQL

Output

Statistics

```
select * from EMPLOYEE t
```

ID

ENAME

EADDRESS

PHONE

SALARY

BANKDETAIL

EROLE

EMAIL

1	693900254	Mickey Depp	95 Salonga Blvd	0553045777	227096	19816	oTVqApiIMW	mickey.depp@amerisourcefunding
2	609876228	Queen Adler	34 Davidson Drive	0584208882	41726	40538	eEVERSQKvY	queen.adler@capitalbank.ar
3	888500354	Fred McAnally	40 Masur Drive	0550081748	95654	78895	QbNFttBbid	fred@air.at
4	368934617	Vanessa Pleasence	13rd Street	0582700589	216833	98421	mDKljBhrkm	vanessa.pleasence@wyeth.my
5	502815898	Nikki Twilley	62nd Street	0536676535	402844	16589	wUJsHzpsgw	n.twilley@capitalautomotive.ch
6	987873783	Bret Begley	72nd Street	0528872623	521051	41915	QoUJBLjnzU	b.begley@gapinc.au
7	315617543	Tilda Fox	38 Williamson Ave	0587852821	433804	24642	sUWLAcnTCO	t.fox@gra.com

Data from Textfile
Data to Oracle

File Data

447475653
242742117

Configuration

General

Fieldcount
1

☒ End at line-end
☐ Name in header
☒ Skip empty lines

Quote character
"

Comment line

Import lines
1 ..

Field1 (+0 .. "<cr>")

Filter

Result Preview

1
447475653
242742117

1 447475653
2 242742117
3 478264648
4 181589738
5 311812506
6 167479605
7 152626579
8 437220106
9 794572877
10 364865809
11 436613635
12 548170364
13 375049271
14 167499704

Data from Textfile Data to Oracle

General

Owner	Table	
SYSTEM ▼	PROFEMP	

Commit every...
☒ Overwrite duplicates ☐ Delete records
☐ Ignore duplicates ☐ Truncate table

100 ▲▼

Fields

Field1 -> ID (NUMBER)

SQL Output Statistics

```
select * from PROFEMP t
```

SQL Query Editor Toolbar:

- Save (Disk icon)
- Run (Play icon)
- Refresh (Circular arrow icon)
- Undo (Curved arrow icon)
- Redo (Curved arrow icon)
- Zoom In (Magnifying glass icon)
- Zoom Out (Magnifying glass icon)
- Fullscreen (Four arrows icon)
- Help (Question mark icon)
- Database Explorer (Tree icon)
- Schema Explorer (Diagram icon)
- Table Explorer (Table icon)
- Query Explorer (Query icon)
- Log Explorer (Log icon)
- Network Explorer (Network icon)

ID	
1	102717015
2	107755377
3	112615906
4	116303908
5	126539544
6	129224522
7	129948619
8	132383180
9	133070595
10	135033462
11	136649947
12	141185465
13	142901643
14	146575324
15	152626579

3. Resident – הנסה באמצעות Data Generator

RESIDENT

Owner

Table

Number of records

SYSTEM

RESIDENT

5000

Name	Type	Size	Data
ID	NUMBER		Random(100000000, 999999999)
NAME	VARCHAR2	30	FirstName + ' ' + LastName
PHONE	VARCHAR2	11	['05'99999999]
EMAIL	VARCHAR2	30	Email
MAIL	VARCHAR2	30	ZIP
ADDRESS	VARCHAR2	30	Address1
*			

RESIDENT

ID	NAME	PHONE	EMAIL	MAIL	ADDRESS
847520434	Edwin Studi	0524529239	edwin@diageo.de	5439	13rd Street
513171683	Saffron MacDowell	0590613377	smacdowell@tlsservicebureau.fr	21322	32 Roscoe
386555046	Adrien Sorvino	0514836930	a.sorvino@stonetechprofessiona	8779	21 White Road
133629451	Rebeka Davidson	0565878313	r.davidson@wrgservices.com	31066	90 Winger Drive
877112475	Carolyn Hartnett	0548830246	chartnett@ccb.br	21386	33 Heche Drive
568061219	Jeroen Walsh	0566914674	jeroen.walsh@powerlight.com	40281	88 Warden Drive
145820351	Angie Elizondo	0572008074	a.elizondo@unitedasset.ca	3X3 8X9	35 Pomona Street
447946062	Dennis Chaykin	0596663392	dennis@infopros.tw	36737	100 Warren Road
314087806	Jennifer Downey	0511610590	jdowney@tracertechologies.com	34532	92nd Street
936357896	Joshua Judd	0529713821	j.judd@prahs.ch	18201	302 Garofalo Street
283790401	Micky Jovovich	0552244647	micky.jovovich@integramedameri	2641	8 Illeana Street
746656279	Ashton Hawkins	0531452251	ashton.hawkins@fpf.com	23294	102 Bruxelles Road

	ID	NAME	PHONE	EMAIL	MAIL	ADDRESS
1	872233292	Brittany Thompson	0544875776	brittany.thompson@dell.br	24148	3 Harnes Road
2	263289286	Carlene Pigott-Smith	0514394108	carlene@sm.it	29189	38 McLean Street
3	284489311	Bryan Russo	0513891118	bryan.r@pacificdatadesigns.uk	1K0 0T4	998 New Haven Street
4	645722813	Giancarlo Eldard	0593718388	giancarlo.eldard@deutschetelek	1598	58 Hawke Road
5	713369686	Machine Rea	0523315289	machine.rea@clubone.com	26668	57 Blackmore Road
6	959825073	Kurt Cornell	0586217541	kurt.cornell@epiqsystems.de	38394	67 Lennie Street
7	645595090	Mira Lynch	0546388391	mira.lynch@interfacesoftware.d	27836	12nd Street
8	725078664	Nile Costner	0570843572	nile.costner@benecongroup.de	23722	86 Doug Street
9	121403656	Ted Warwick	0513731101	ted.warwick@angieslist.com	20126	7 West Monroe Ave
10	179120284	Alessandro Fiorentino	0535912085	alessandro.fiorentino@ipsadvis	38739	96 Aglukark Road
11	537111283	Leon Copeland	0510832740	leon.c@actechologies.br	25153	99 Enschede Street
12	935398543	Joey Bassett	0501825668	joey.b@conagra.gr	39412	43 Fort Saskatchewan Road

4. PolEmp – הכנסה באמצעות Text Importer

Data from Textfile	Data to Oracle
File Data Yahadut Hatora,540381320 Avoda,461302434	
Configuration <div> <div> General Fieldcount 2 <input checked="" type="checkbox"/> End at line-end <input type="checkbox"/> Name in header <input checked="" type="checkbox"/> Skip empty lines </div> <div> Quote character " Comment line Import lines 1 .. </div> <div> Field1 (+0 .. ",") Field2 (+0 .. ",") </div> </div>	
1	Yahadut Hatora,540381320
2	Avoda,461302434
3	Avoda,984907020
4	Yesh Atid,692063529
5	Likud,357387081
6	Yahadut Hatora,272628715
7	Yahadut Hatora,462423599
8	Yahadut Hatora,646612513
9	Merets,893538157
10	Yahadut Hatora,827706241

Data from Textfile	Data to Oracle
General <div> <div> Owner SYSTEM </div> <div> Table POLEMP </div> <div> Commit every... 100 </div> <div> <input checked="" type="radio"/> Overwrite duplicates <input type="radio"/> Ignore duplicates </div> <div> <input type="checkbox"/> Delete records <input type="checkbox"/> Truncate table </div> <div> Initializing Script Finalizing Script </div> </div>	
Fields <div> <div> Field1 -> PARTY (VARCHAR2) Field2 -> ID (NUMBER) </div> <div> Field ID (NUMBER) Fieldtype Number Create SQL SQL function additional Oracle proces </div> </div>	

	PARTY	ID
1	Shas	928321429
2	Yahadut Hatora	540381320
3	Avoda	461302434
4	Avoda	984907020
5	Yesh Atid	692063529
6	Likud	357387081
7	Yahadut Hatora	272628715
8	Yahadut Hatora	462423599
9	Yahadut Hatora	646612513
10	Merets	893538157
11	Yahadut Hatora	827706241
12	Avoda	221573011

BUDGET

Owner	Table	Number of records
SYSTEM	BUDGET	52

Name	Type	Size	Data
YEAR	NUMBER		Sequence(1970)

BUDGET

YEAR
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979

```
select * from BUDGET t
```

	YEAR
1	1970
2	1971
3	1972
4	1973
5	1974
6	1975
7	1976
8	1977
9	1978
10	1979
11	1980
12	1981
13	1982

Text Importer – Source .6

Data from Textfile Data to Oracle

File Data

2000,2016-02-24,607,2018
800,1989-03-06,258,1989

Configuration

General

Fieldcount: 4

☒ End at line-end
☐ Name in header
☒ Skip empty lines

Quote character: "

Comment line:

Import lines: 1 ..

Field1 (+0 .. ",")
Field2 (+0 .. ",")
Field3 (+0 .. ",")
Field4 (+0 .. ",")

```

1 2000,2016-02-24,607,2018
2 800,1989-03-06,258,1989
3 4300,2010-12-28,542,2012
4 800,1993-08-18,321,1995
5 2100,1969-05-22,7,1971
6 5000,1989-03-01,257,1991
7 4700,2009-01-20,522,2010
8 3600,1986-08-24,236,1986
9 5000,1974-07-16,78,1974
10 3500,2004-02-16,452,2006
11 4500,1982-10-06,170,1983
12 4800,1985-11-13,224,1985
13 3600,2003-07-13,440,2003

```

Filter:

Result Preview

1	2	3	4
2000	2016-02-24	607	2018
800	1989-03-06	258	1989

Data from Textfile Data to Oracle

General

Owner: SYSTEM

Table: SOURCE

Commit every...: 100

☒ Overwrite duplicates
☐ Ignore duplicates

☐ Delete records
☐ Truncate table

Initializing Script:

Finalizing Script:

Fields

Field1 -> AMOUNT (NUMBER)
Field2 -> SDATE (DATE)
Field3 -> SID (NUMBER)
Field4 -> YEAR (NUMBER)

Field: AMOUNT (NUMBER)

Fieldtype: Number

Create SQL

select * from SOURCE t

	AMOUNT	SDATE	SID	YEAR
1	2000	24/02/2016	607	2018
2	800	06/03/1989	258	1989
3	4300	28/12/2010	542	2012
4	800	18/08/1993	321	1995
5	2100	22/05/1969	7	1971

7. Collection - הנסחה באמצעות Text Importer

Data from Textfile **Data to Oracle**

File Data

842854553,0
785827041,1

Configuration

General

Fieldcount:

Quote character:

Field1 (+0 .. ",")
Field2 (+0 .. ",")

☒ End at line-end
☐ Name in header
☒ Skip empty lines

Comment line:

Import lines: ..

Filter:

Result Preview

1	2
842854553	0
785827041	1

1 842854553,0
2 785827041,1
3 647840745,2
4 172496601,3
5 943932456,5
6 713369686,7
7 903808719,9
8 205299329,10
9 283648042,13
10 663986400,14
11 779872117,15
12 799659290,16
13 628497750,17

Data from Textfile **Data to Oracle**

General

Owner: Table:

Commit every...: ☒ Overwrite duplicates ☐ Delete records ☐ Ignore duplicates ☐ Truncate table

Initializing Script:
Finalizing Script:

Fields

Field1 -> ID (NUMBER)
Field2 -> SID (NUMBER)

Field:
Fieldtype:
Create SQL:
SQL function:
additional Oracle processing:

Result Preview

1	2
842854553	0
785827041	1

SQL **Output** Statistics

`select * from COLLECTION t`

	ID	SID
1	842854553	0
2	785827041	1
3	647840745	2
4	172496601	3

8. GovFund – הכנסה באמצעות Text Importer

Data from Textfile

Data to Oracle

File Data

12

32

Configuration

General

Fieldcount

1

☒ End at line-end

☐ Name in header

☒ Skip empty lines

Quote character

"

Comment line

Import lines

1 ..

Field1 (+0 .. "<cr>")

Filter

Result Preview

1

12

32

1 12

2 32

3 52

4 56

5 73

6 92

7 110

8 117

9 122

10 125

11 134

12 149

13 160

Data from Textfile

Data to Oracle

General

Owner

SYSTEM

Table

GOVFUND

Commit every...

100

☒ Overwrite duplicates

☐ Ignore duplicates

☐ Delete records

☐ Truncate table

Initializing Script

Finalizing Script

Fields

Field1 -> SID (NUMBER)

Field

Fieldtype

Create SQL

SQL function

additional Oracle processing,

Result Preview

1

12

32

1

12

2

32

3

52

4

56

5

73

SID

12

32

52

56

73

File Data	
Arnona, 1	
Fee, 2	

Configuration	
General	
Fieldcount 2	Quote character "
<input checked="" type="checkbox"/> End at line-end	Comment line "
<input type="checkbox"/> Name in header	Import lines 1 ..
<input checked="" type="checkbox"/> Skip empty lines	

Result Preview	
1	2
Arnona	0
Arnona	1

Data from Textfile		Data to Oracle	
General			
Owner SYSTEM	Table TAXES	Initializing Script	
Commit every... 100	<input checked="" type="radio"/> Overwrite duplicates <input type="radio"/> Ignore duplicates	<input type="checkbox"/> Delete records <input type="checkbox"/> Truncate table	
Fields		Finalizing Script	
Field1 -> TYPE (VARCHAR2) Field2 -> SID (NUMBER)		Field Fieldtype Create SQL SQL function additional Oracle processing,	
Result Preview			
1	2		
Arnona	0		
Arnona	1		

1 Arnona, 0
2 Arnona, 1
3 Fee, 2
4 Arnona, 3
5 Fee, 5
6 VAT, 7
7 Arnona, 9
8 Income, 10
9 Corporate, 13
10 VAT, 14
11 Corporate, 15
12 Arnona, 16
13 Corporate, 17

```
select * from TAXES t
```

	TYPE	SID
1	Arnona	0
2	Arnona	1
3	Fee	2
4	Corporate	322
5	Arnona	323

DONOR

Owner	Table	Number of records
SYSTEM	DONOR	100

Name	Type	Size	Data
NAME	VARCHAR2	30	FirstName + ' ' + LastName
ID	NUMBER		Random(100000000,999999999)

DONOR

NAME	ID
Wayne Beck	616274384
Kasey Evans	709731441
Clarence Kutcher	289668867
Fats Skaggs	506317503
Casey Prinze	121654916
Gilbert Ferrell	179066017
Marina Pigott-Smith	457678334
Mike Sevigny	402900945
Deborah Benoit	791185092
Tcheky Stowe	293074962
Arnold Atlas	320162964

SQL

Output Statistics

```
select * from DONOR t
```



	NAME	ID
1	Pablo Lucien	381180820
2	Rory Tippe	753601950
3	Jeffery Kershaw	446221997
4	Robbie Biehn	840098188
5	George Levy	223231546
6	Joshua Cromwell	805990297
7	Renee O'Connor	426581428
8	Thin Vance	296891114

INSPECTOR

Owner	Table	Number of records
SYSTEM	INSPECTOR	100

Name	Type	Size	Data
ID	NUMBER		List(select ID from profEmp)

INSPECTOR

ID
218216379
368934617
901549877
243147104
649286448
707697160
244014561

SQL Output Statistics

```
select * from INSPECTOR t
```



ID
1 107755377
2 112615906
3 129224522
4 133070595
5 135033462
6 142901643
7 152626579

Data from Textfile Data to Oracle

File Data

Crime, 351, 899397565
Littering, 352, 508403262

Configuration

General

Fieldcount
3

Quote character
"

Comment line
--

Import lines
1 ..

☒ End at line-end
☐ Name in header
☒ Skip empty lines

Field1 (+0 .. ",")
Field2 (+0 .. ",")
Field3 (+0 .. ",")

Filter

Result Preview

1	2	3
Speeding	350	775426715
Crime	351	899397565

1 Speeding, 350, 775426715
2 Crime, 351, 899397565
3 Littering, 352, 508403262
4 Crime, 354, 256414545
5 No Mask, 355, 194568484
6 Parking, 357, 419873491
7 Other, 358, 647847654
8 Speeding, 360, 374962500
9 Other, 361, 912193291
10 Speeding, 362, 934123585
11 Parking, 364, 135033462
12 Littering, 366, 876828519
13 Littering, 367, 468450266

Data from Textfile Data to Oracle

General

Owner
SYSTEM

Table
FINES

Commit every...
100

☒ Overwrite duplicates
☐ Ignore duplicates

☐ Delete records
☐ Truncate table

Initializing Script
Finalizing Script

Fields

Field1 -> REASON (VARCHAR2)
Field2 -> SID (NUMBER)
Field3 -> IID (NUMBER)

Field
Fieldtype
Create SQL
SQL function
additional Oracle processing, f

Result Preview

1	2	3
Speeding	350	775426715
Crime	351	899397565

	REASON	SID	IID
1	Speeding	426	513760778
2	Parking	427	358987417
3	Parking	428	705344711
4	Littering	429	162729695
5	Parking	430	243147104

SQL	Output	Statistics
<pre> insert into SYSTEM.MEMBER(ID) values ('221573011'); insert into SYSTEM.MEMBER(ID) values ('940046023'); insert into SYSTEM.MEMBER(ID) values ('157970965'); insert into SYSTEM.MEMBER(ID) values ('858776225'); insert into SYSTEM.MEMBER(ID) values ('369589351'); insert into SYSTEM.MEMBER(ID) values ('682488428'); insert into SYSTEM.MEMBER(ID) values ('424428722'); insert into SYSTEM.MEMBER(ID) values ('492083823'); insert into SYSTEM.MEMBER(ID) values ('661579102'); insert into SYSTEM.MEMBER(ID) values ('420676691'); insert into SYSTEM.MEMBER(ID) values ('825619589'); insert into SYSTEM.MEMBER(ID) values ('272494951'); insert into SYSTEM.MEMBER(ID) values ('975286099'); insert into SYSTEM.MEMBER(ID) values ('782309922'); insert into SYSTEM.MEMBER(ID) values ('928321429'); commit; select * from MEMBER t </pre>		


ID
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15

SQL	Output	Statistics
<pre> insert into SYSTEM.CHAIRMAN(ID) values ('540381320'); insert into SYSTEM.CHAIRMAN(ID) values ('461302434'); insert into SYSTEM.CHAIRMAN(ID) values ('984907020'); insert into SYSTEM.CHAIRMAN(ID) values ('692063529'); insert into SYSTEM.CHAIRMAN(ID) values ('357387081'); insert into SYSTEM.CHAIRMAN(ID) values ('272628715'); insert into SYSTEM.CHAIRMAN(ID) values ('462423599'); insert into SYSTEM.CHAIRMAN(ID) values ('646612513'); insert into SYSTEM.CHAIRMAN(ID) values ('893538157'); insert into SYSTEM.CHAIRMAN(ID) values ('827706241'); commit; </pre>		

SQLOutputStatistics

select * from CHAIRMAN t

SQL	Output	Statistics
<pre> insert into SYSTEM.COMMITTEE (name,role,ID) values ('Health','Health','540381320'); insert into SYSTEM.COMMITTEE (name,role,ID) values ('Education','Education','461302434'); insert into SYSTEM.COMMITTEE (name,role,ID) values ('Finance','Finance','984907020'); insert into SYSTEM.COMMITTEE (name,role,ID) values ('Welfare','Welfare','692063529'); insert into SYSTEM.COMMITTEE (name,role,ID) values ('Sanitation','Sanitation','357387081'); insert into SYSTEM.COMMITTEE (name,role,ID) values ('Plenum','Plenum','272628715'); insert into SYSTEM.COMMITTEE (name,role,ID) values ('Audit','Audit','462423599'); insert into SYSTEM.COMMITTEE (name,role,ID) values ('Tourism','Tourism','646612513'); insert into SYSTEM.COMMITTEE (name,role,ID) values ('Sports','Sports','893538157'); insert into SYSTEM.COMMITTEE (name,role,ID) values ('Community','Community','827706241'); commit; </pre>		

SQL	Output	Statistics
<pre> select * from COMMITTEE t </pre>		
		
CNAME	ROLE	ID
1 Health	Health	540381320
2 Education	Education	461302434
3 Finance	Finance	984907020
4 Welfare	Welfare	692063529
5 Sanitation	Sanitation	357387081
6 Plenum	Plenum	272628715
7 Audit	Audit	462423599
8 Tourism	Tourism	646612513
9 Sports	Sports	893538157
10 Community	Community	827706241

INSTITUTION			
Owner	Table	Number of records	
SYSTEM	INSTITUTION	500	
Name	Type	Size	Data
ADDRESS	VARCHAR2	30	Address1
INSTNAME	VARCHAR2	30	[@@@@@@@@@@@@]
TYPE	VARCHAR2	20	List('Landfill','university','School','clinic','Bank','Museum','Nursing Home','Hospital','Visitors
COMMITNAME	VARCHAR2	30	List(select cname from committee)

INSTITUTION				
ADDRESS	INSTNAME	TYPE	COMMITNAME	
493 Galecki Street	QmSrCQIWep	Visitors center	Health	
97 Speaks Road	UPXwocMKTf	Hospital	Tourism	
68 Gilley Blvd	gAsPdNiNdb	university	Health	
52nd Street	wHjZetwPaC	Sports hall	Health	
48 Travolta Street	jCNwAJkFjL	Mall	Audit	
34 Henry Ave	snlleStySF	Museum	Plenum	
26 Utsunomiya Street	GiGPSpMWuy	Landfill	Sanitation	
44 Moreno Valley Street	pmszljNPzs	Mall	Plenum	
3 English Road	usbaaGiqVc	Nature site	Community	
23 Lizzy Road	SZAmvHicLA	School	Community	
688 Chesnutt Street	gUFYMTAGNE	clinic	Finance	

```
select * from INSTITUTION t
```

ADDRESS	INSTNAME	TYPE	COMMITNAME	
1 34 Bultman Place	TLrqiZRuQN	Landfill	Sanitation	
2 91132 Everett Lane	APwRLRIIZE	university	Education	
3 39160 Cambridge Trail	mIrcnVzsxQ	School	Education	
4 71030 Monica Crossing	CVnivvcFBe	clinic	Health	
5 8 Sachtjen Way	LDNgCyssiL	School	Education	
6 8053 Northfield Way	zCgHwxXcr	Bank	Finance	
7 87 Larry Hill	OkjYlvdIqK	Landfill	Sanitation	
8 5518 Onsgard Center	yPqeHkILxE	Museum	Tourism	
9 419 Kingsford Alley	yNwTnFiffq	Nursing Home	Welfare	
10 782 Meadow Ridge Alley	IlbDWUnHAf	Museum	Tourism	

SQL	Output	Statistics
<pre> insert into SYSTEM.MEMBERAT (name, ID) values ('Finance','661579102'); insert into SYSTEM.MEMBERAT (name, ID) values ('Welfare','682488428'); insert into SYSTEM.MEMBERAT (name, ID) values ('Welfare','492083823'); insert into SYSTEM.MEMBERAT (name, ID) values ('Welfare','221573011'); insert into SYSTEM.MEMBERAT (name, ID) values ('Welfare','661579102'); insert into SYSTEM.MEMBERAT (name, ID) values ('Sanitation','420676691'); insert into SYSTEM.MEMBERAT (name, ID) values ('Sanitation','825619589'); insert into SYSTEM.MEMBERAT (name, ID) values ('Sanitation','424428722'); insert into SYSTEM.MEMBERAT (name, ID) values ('Sanitation','682488428'); insert into SYSTEM.MEMBERAT (name, ID) values ('Plenum','940046023'); insert into SYSTEM.MEMBERAT (name, ID) values ('Plenum','221573011'); insert into SYSTEM.MEMBERAT (name, ID) values ('Plenum','782309922'); insert into SYSTEM.MEMBERAT (name, ID) values ('Plenum','272494951'); insert into SYSTEM.MEMBERAT (name, ID) values ('Audit','221573011'); insert into SYSTEM.MEMBERAT (name, ID) values ('Audit','661579102'); insert into SYSTEM.MEMBERAT (name, ID) values ('Audit','782309922'); insert into SYSTEM.MEMBERAT (name, ID) values ('Audit','975286099'); insert into SYSTEM.MEMBERAT (name, ID) values ('Tourism','661579102'); insert into SYSTEM.MEMBERAT (name, ID) values ('Tourism','940046023'); insert into SYSTEM.MEMBERAT (name, ID) values ('Tourism','682488428'); insert into SYSTEM.MEMBERAT (name, ID) values ('Tourism','858776225'); insert into SYSTEM.MEMBERAT (name, ID) values ('Sports','825619589'); insert into SYSTEM.MEMBERAT (name, ID) values ('Sports','420676691'); insert into SYSTEM.MEMBERAT (name, ID) values ('Sports','975286099'); insert into SYSTEM.MEMBERAT (name, ID) values ('Sports','157970965'); insert into SYSTEM.MEMBERAT (name, ID) values ('Community','940046023'); insert into SYSTEM.MEMBERAT (name, ID) values ('Community','424428722'); insert into SYSTEM.MEMBERAT (name, ID) values ('Community','272494951'); insert into SYSTEM.MEMBERAT (name, ID) values ('Community','682488428'); commit; </pre>		

SQL	Output	Statistics
<pre> select * from MEMBERAT t </pre>		

	NAME	ID
1	Audit	221573011
2	Audit	661579102
3	Audit	782309922
4	Audit	975286099
5	Community	272494951

INQUIRIES

Owner	Table	Number of records
SYSTEM	INQUIRIES	10..20

Name	Type	Size	Data
IID	NUMBER		Sequence(0)
IDATE	DATE		Random(01/01/1970, 31/12/2021)
TYPE	VARCHAR2	10	[a]
DETAILS	VARCHAR2	100	Text(44, 5, 1)
STATUS	CHAR	1	List('0', '1')
ID	NUMBER		List(select id from resident)
COMMITNAME	VARCHAR2	30	List(select cname from committee)

INQUIRIES

IID	IDATE	TYPE	DETAILS	STATUS	ID	COMMITNAME
0	30-09-1974	C	b5foCJLCA2tXISoLv2y1y3MKCBga7zFjoPASiekvcLd0	1	512223292	Sanitation
1	15-10-2009	Y	5IXAR3Kpj3wPz8QeTWDLPgqfY8aciqLKKiZOWlhQ5mIK	1	850191507	Sanitation
2	23-12-1981	O	1fiO2pvgle2pjlui7omkkTey08dzvYSqxTrqo8REkkAr	1	492135813	Sanitation
3	26-03-1987	S	BhJxZ8pkYVAUaUNpXCawcGhR708vokyZ45KjDniDTDzO	0	760775761	Tourism
4	18-02-2005	D	gujLjDR0bK4JPgWRUmuuuLpP9cG86Suduz0hXZVe7YiC	0	148730976	Sanitation
5	13-03-1998	V	YjM34EriagO5le50vzlgBcKIIBMxL0pFPybBsGMQqAOq	1	829737520	Health
6	08-10-1996	M	P2V7xKqQiRM0fJY0muWOvSshCw8M9ZE2Q4g4uwjDWqfD	1	975286099	Plenum
7	17-05-2009	Q	ofuTijrDXZFe6XqcoBO4UtJklEBQnCY67WJLgJbXY9J	1	858881585	Plenum
8	05-03-2018	V	L4ElrmX8cEsJwHFuJABxvizMkHJo7JVohNjE2wfHpSF	0	338455169	Audit
9	04-04-1988	L	gq9YCok4nNaQXlcZZh8ivxpQb7e6iSXJfLTXAQbG9X4	0	233688229	Sanitation
10	08-01-2010	L	cu4fbsAIT4PZsqdcyotbyMgBpjVUcoVHjP9rmnOnscjj	1	489728323	Welfare
11	05-09-1995	A	YZSOjsoMI5JAfHrp5sVMNo9Msxy59eVbnVXVQbTPpkhr	0	727316756	Welfare
12	15-01-2011	E	F0B82WhyY0vCQb7DzhO73fIQXxHsZHdM92OPFJXN5g	1	768270136	Sanitation
13	10-10-1988	G	84FJ25Ina6QGHGPM5fC9kNiaYgIBFsPDUUzEMyZxVU7y	1	696893920	Welfare
14	03-03-1987	G	IPKekBpsNwDordumYLXY8ldpfjbBoJ41xcofz63aDvab	0	446186142	Welfare
15	27-03-1981	A	1lfzlezhr5mDRTKqBHHMBcojiADP2V59WGt0GPJ5OgVy	1	133548137	Tourism
16	02-10-2013	R	HkncS7t75twln5G8Fair1nTzdlJcmFuf9z7A4n80tVGt	1	771799883	Audit

IID	IDATE	TYPE	DETAILS	STATUS	ID	COMMITNAME
1	1003 27/10/2020	D	kXgFjAoVGDaUy7GF3BEzSDgJmqNYFKAxJAXLiq8VJc	1	960676293	Sanitation
2	1004 08/08/2020	C	9cTh00g4mnXXDse4IQj1fPudb8YQ8AJHVvpW2xHXyKa	1	244479870	Tourism
3	1005 28/12/2017	F	QyzwjBBdPk9hHSCYCW5c26wbQioyECC6sZlek3xNhtV	1	772844399	Welfare
4	1006 03/01/2022	B	GvoQCGhwADV664J1A3CH6K83Zo9ChUYy2krY1qCJmQx	1	760775761	Welfare
5	1007 01/09/2017	G	nTrFbhrfHRFHUOROoInYFNIXhfN6NFzKOeYOG7MA8Sr	1	165925600	Welfare
6	1008 07/02/2019	E	3pfx9tqSikyJVXInJmYJFSTPj55rqLVX1zxVdluC6v	1	790580812	Plenum
7	1009 23/10/2020	C	0u9UWJf0PkmtD07UW65iLtfmDDBY2I7NX6CqJdEuCgr	0	848880540	Plenum
8	1010 24/07/2021	A	dFwXXcQs7x9IYCw1W0H0wibAoQpWFnwURCRyPTIUSy	0	686288621	Health
9	1011 26/08/2017	F	hXjB88cTCzVIN6Mblmqawdkzh8rjsE09K6ZHy11W6B	1	131737742	Sanitation
10	1012 13/08/2020	G	w8BWkDCKANzOAMJgJ5A0fKmp2T9uZBe6scAisRaNFBG	1	829546853	Tourism
11	1013 02/03/2018	A	UJvzn1YzEoYMQ3Gsj5s86vVITaTsA2qRENNtgzK0mFa	0	199312716	Tourism
12	1014 16/07/2021	C	oPrMH6vhR1YNH6IRwCa40K1ib3YCoc4FsEt4Y5Mn1v	0	735976247	Welfare
13	1015 05/11/2017	F	lyae25a9WEsPgJxzO9d6wMGWqcZ78jmR6oQcJUV6SS	0	615189393	Education
14	1016 28/03/2022	D	8VzNWwxni6vEEM2XiWaPS9e79jg91jDS0S24Q7eOuQ	1	466583755	Finance
15	1017 27/12/2019	F	8QnETjc7AwOCmmFqaGAixtFkjzFM5DXmaLEjFd6iLya	1	187263904	Audit
16	1018 18/09/2019	E	iPDBofYrt8zrFGW2s1tUvGzihj5XcEapOE3zHLHPBAZ	0	550931510	Tourism

Data from Textfile Data to Oracle

File Data

225247,1970,Health
206246,1970,Education
152379,1970,Finance
155233,1970,Welfare
146796,1970,Sanitation
120242,1970,Plenum
101091,1970,Audit
100618,1970,Tourism
97304,1970,Sports
797444,1970,Community
628059,1971,Health
586186,1971,Education
573869,1971,Finance

Configuration

General
Fieldcount: 3
☒ End at line-end
☐ Name in header
☒ Skip empty lines

Quote character: " Comment line: Import lines: 1 -

Field1 (+0..")
Field2 (+0..")
Field3 (+0..")

Filter:

Result Preview

1	2	3
225247	1970	Health
206246	1970	Education

1 225247,1970,Health
2 206246,1970,Education
3 152379,1970,Finance
4 155233,1970,Welfare
5 146796,1970,Sanitation
6 120242,1970,Plenum
7 101091,1970,Audit
8 100618,1970,Tourism
9 97304,1970,Sports
10 797444,1970,Community
11 628059,1971,Health
12 586186,1971,Education
13 573869,1971,Finance

Data from Textfile Data to Oracle

General

Owner: SYSTEM Table: FINANCE

Commit every...: 100 ☒ Overwrite duplicates ☐ Delete records ☐ Truncate table

Initializing Script: Finalizing Script:

Fields

Field1 -> AMOUNT (NUMBER)
Field2 -> YEAR (NUMBER)
Field3 -> NAME (VARCHAR2)

Field: Fieldtype: Create SQL: SQL function: additional Oracle processing

Result Preview

1	2	3
225247	1970	Health
206246	1970	Education

SQL Output Statistics

```
select * from FINANCE t
```

	AMOUNT	YEAR	NAME	
1	225247	1970	Health	...
2	206246	1970	Education	...
3	152379	1970	Finance	...
4	155233	1970	Welfare	...
5	146796	1970	Sanitation	...

שאלת 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:

	NAME	SAVING
1	Community	1145208.4
2	Health	306302.56
3	Education	273636.32
4	Finance	246087.73
5	Subcontractor	223527.6
6	Welfare	219703.26
7	Sanitation	194653.32
8	Supervisor	191440.4

system@XE [18:58:08] 222 rows selected in 0.126 seconds

זמן ביצוע: 0.126 שניות

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;
```

תוצאה:

	ID	PHONE	EMAIL	
▶ 1	794552919	0531185098	avenged@captechventures.de	...
2	568149774	0595419263	m.lipnicki@benecongroup.jp	...
3	570823216	0526986035	annie.s@asa.com	...
4	388479614	0517462934	rosario.wong@fam.jp	...
5	834461674	0594713149	ethand@capitalbank.com	...
6	658531796	0503547009	carl.drive@extremepizza.ca	...
7	177324272	0546709418	mshepherd@bestever.de	...
8	990956935	0533990930	stanleyh@kimberlyclark.com	...

14:1 system@XE [19:15:00] 400 rows selected in 0.305 seconds

זמן ביצוע: 0.305 שניות

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))) ;
```

תוצאה:

	NAME	EMAIL	MAIL
1	Carl Rhys-Davies	c.rhysdavies@verizon.fi	21151
2	Arnold Peniston	a.peniston@logisticare.de	24801
3	Corey Monk	coreym@tilsonlandscape.com	30222
4	Gilbert Pfeiffer	gilbert@oriservices.uk	9E8 7A3
5	Illeana Mac	illeanam@ezecastlesoftware.com	35939
6	Domingo Finney	domingo.finney@doraldentalusa.	3546
7	Deborah Purefoy	deborah.purefoy@kmart.au	13950
8	Garry Ali	garry@hudsonriverbancorp.ca	9C7 1D9
9	Jay Walsh	jay@terrafirma.com	33796

1 of 420 system@XE [19:23:41] 420 rows selected in 0.297 seconds

זמן ביצוע: 0.297 שניות

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)))));
```

תוצאה:

	ID	NAME	ADDRESS
▶ 1	872233292	Brittany Thompson	3 Harnes Road
2	263289286	Carlene Pigott-Smith	38 McLean Street
3	284489311	Bryan Russo	998 New Haven Street
4	645722813	Giancarlo Eldard	58 Hawke Road
5	713369686	Machine Rea	57 Blackmore Road
6	959825073	Kurt Cornell	67 Lennie Street
7	645595090	Mira Lynch	12nd Street
8	725078664	Nile Costner	86 Doug Street
9	121403656	Ted Warwick	7 West Monroe Ave

1 of 296 system@XE [19:35:24] 296 rows selected in 0.181 seconds

זמן ביצוע: 0.181 שניות

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;
```

	TYPE	SUM(F.AMOUNT)
1	Community center	593676034560
2	Park	445257025920
3	Synagogue	420520524480
4	Nursing Home	306266344440
5	Landfill	208590497712
6	School	199207240960
7	Mall	192440604860
8	Hospital	171529433600
9	clinic	159277331200

system@XE [19:36:45] 16 rows selected in 0.329 seconds

תוצאה:

זמן ביצוע: 0.329 שניות

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;
```

תוצאה:

	ID	SUM(AMOUNT)
1	864581767	192000
2	381180820	77000
3	161595151	93000
4	618108092	168000
5	512478228	77000
6	649674677	95000
7	778960157	89000
8	805990297	80000
9	226510528	3600

1 of 517 system@XE [19:42:32] 517 rows selected in 0.281 seconds

זמן ביצוע: 0.281 שניות

עכשיו שתי שאילות עם המון נתונים, כדי לבדוק גם שאילות 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;
```

תוצאה:

		INSTNAME	YEAR	PERCENT_OF_BUDGET
▶	1	APwRLRIIZE	1970	0.0980909350328165
	2	APwRLRIIZE	1971	0.0916775101657804
	3	APwRLRIIZE	1972	0.0945145355341744
	4	APwRLRIIZE	1973	0.0833946878740829
	5	APwRLRIIZE	1974	0.0863771061912226
	6	APwRLRIIZE	1975	0.0915732687619424
	7	APwRLRIIZE	1976	0.0880528033046966
	8	APwRLRIIZE	1977	0.0960214608086949
	9	APwRLRIIZE	1978	0.0993390529382706

1 of 26000 0:13 system@XE [19:48:22] 26000 rows selected in 13.481 seconds

זמן ביצוע: 13.481 שניות

תיאור: פורים מגיע, ולו נרצה לבצע משחק וגמד ענק בין כל תושבי העירייה, יש שאלתא 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;
```

תוצאה:

		NAME1	ADDRESS1	NAME2	ADDRESS2
▶	1	Aaron Adkins	91st Street	Sammy Union	56 Farris Ave
	2	Aaron Trevino	7 Davidson Drive	Taye Shannon	6 Thorton
	3	Adam Cara	36 Bello Street	Courtney Heald	50 Thame Road
	4	Adam Elliott	77 Marshall Street	Busta McFadden	94 Apple Road
	5	Adam Moody	83rd Street	Rowan Morales	88 Boothe Drive
	6	Adam Wopat	246 Mortensen Road	Brent Hudson	34 Franz Ave
	7	Adina Bancroft	48 Pfeiffer Ave	Stellan Heche	68 Bronx Drive
	8	Adina Bonneville	10 Helen Road	Giovanni Sandler	48 Cornell Road
	9	Adina Cherry	56 Young Road	Rhona Loggins	3 Krumholtz Street
	10	Adina Melvin	26 Pusan-city	Shannyn Arkenstone	58 Bruneck Drive
	11	Adina Shatner	29 Lena Blvd	Busta McGriff	1 Moss Road

< [16:54:43] 2498 rows selected in 3.892 seconds

זמן ביצוע: 3.892 שניות.

1. האינדקס:

```
create index inst_indx on INSTITUTION ( commitname );
```

שאלתא לדוגמה שמתמשת בו: [שאלתא מס' 7](#).

השוואת זמנים:

97:48	0:11	system@XE	[22:22:30]	26000 rows selected in 11.890 seconds
-------	------	-----------	------------	---------------------------------------

לפני	אחרי	שיפור?
13.481	11.890	V

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

2. האינדקס:

```
create unique index clc_indx on COLLECTION ( sid, id );
```

שאלתא לדוגמה שמתמשת בו: [שאלתא מס' 4](#).

השוואת זמנים:

1 of 296	system@XE	[14:39:28]	296 rows selected in 0.131 seconds
----------	-----------	------------	------------------------------------

לפני	אחרי	שיפור?
0.181	0.131	V

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

בס"ד

3. האינדקס:

```
create index inq_idx on INQUIRIES ( commitname );
```

שאלתא לדוגמה שמתמשת בו: [שאלתא מס' 3](#).

השוואת זמנים:

system@XE [15:27:33] 420 rows selected in 0.228 seconds

שיפור?	אחרי	לפני
V	0.228	0.297

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

4. האינדקס:

```
create index inq_idx on RESIDENT ( name );
```

שאלתא לדוגמה שמתמשת בו: [שאלתא מס' 8](#).

השוואת זמנים:

1 of 2498 0:02 system@XE [16:56:54] 2498 rows selected in 2.790 seconds

שיפור?	אחרי	לפני
V	2.790	3.892

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!

5. האינדקס:

```
create index inq_idx on RESIDENT ( name );
```

שאלתא לדוגמה שמתמשת בו: [שאלתא מס' 5](#).

השוואת זמנים:

1 of 16	system@XE [17:04:58]	16 rows selected in 0.337 seconds
---------	----------------------	-----------------------------------

שיפור?

אחרי

לפני

X

0.337

0.329

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.

6. האינדקס:

```
create index src_ne_idx on SOURCE ( amount );
```

שאלתא לדוגמה שמתמשת בו: [שאלתא מס' 6](#).

השוואת זמנים:

1 of 517	system@XE [17:13:19]	517 rows selected in 0.348 seconds
----------	----------------------	------------------------------------

שיפור?

אחרי

לפני

X

0.348

0.281

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.

תצפיות – Views

1. תיאור מילולי – נבנה תצפית המייצגת את התקציב של כל שנה כסכום של ההכנסות של כל שנה, כך שעבור השקיפות של העירייה יהיה ניתן לראות בקלות ובבהירות מה התקציב השנתי עבור כל שנה ושנה. בנוסף, גם האחראי על התקציבים יוכל לקחת מכך השראות לסכומים של התקציבים לשנים הבאות.

יצירת התצפית –

```
CREATE VIEW year_budget as
SELECT B.year as YEAR, sum(amount) AS amount
FROM BUDGET B, SOURCE S
WHERE B.YEAR = S.YEAR
GROUP BY B.YEAR;
```

בדיקה שאכן התצפית נוצרה והערכים נכנסו -

SQL	Output	Statistics
SELECT * FROM year_budget		
	YEAR	AMOUNT
1	2018	4016500
2	1989	2596100
3	2012	4237400
4	1995	1592800
5	1971	6394000

תיאור שאילתא על התצפית – נרצה לחפש מה התקציבים שגדולים מהממוצע של התקציבים סה"כ, על מנת ליצור השנה תקציב יותר גדול ממנו, כיוון שהשנה ההכנסות של העירייה גדלו ב"ה ☺

השאילתא על התצפית –

```
SELECT *
FROM year_budget
WHERE amount >= (SELECT AVG (AMOUNT)
                  FROM year_budget)
ORDER BY amount DESC;
```

הרצת השאילתא על התצפית -

	YEAR	AMOUNT
1	2009	10214800
2	1984	10008400
3	1978	9919100
4	2010	9807000

2. תיאור מילולי – נבנה תצפית המציינת מי ראשי הוועדות המוניציפליות השונים, מה שמותיהם, מה תפקידיהם ומה המייל שלהם, על מנת לאפשר לציבור הרחב לתקשר ולשלוח פניות לראש הוועדה הרלוונטי על ידי השגת פרטיו בקלות בעזרת התצפית הזאת.

יצירת התצפית –

```
CREATE VIEW chairman_name_role_email AS
SELECT name, cname AS commitee_name, email
FROM RESIDENT NATURAL JOIN COMMITTEE;
```


בדיקה שאכן התצפית נוצרה והערכים נכנסו -

SQL

Output

Statistics

SELECT * FROM chairman_name_role_email



	NAME	COMMITTEE_NAME	EMAIL
1	Madeleine Bergen	Health	madeleineb@johnson.com
2	Spike Rush	Education	spike.rush@trusecure.cr
3	Patti Clinton	Finance	patti@sensortechnologies.jp
4	Grace Trevino	Audit	grace.trevino@nhhc.com

תיאור שאילתא על התצפית – נרצה לבדוק מי ומה פרטיו של יו"ר הוועדה המוניציפלית שאחראית על ספורט ותרבות על מנת לשלוח למייל שלו בקשה לשפץ ולשדרג את מגרש הכדורגל הקטן והמסכן במכון.

השאילתא על התצפית –

```
SELECT name, email
FROM chairman_name_role_email
WHERE commitee_name LIKE 'Sport%'
```

הרצת השאילתא על התצפית -

	NAME	EMAIL
1	Temuera Rickles	temuera.rickles@medamicus.uk

וגילינו בהצלחה שזהו האדון טאמורה ריקלס הנכבד.

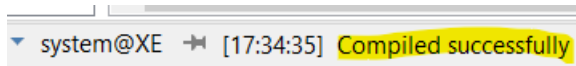
1. נגדיר פונקציה על מנת לבצע העלאה במשכורת של כמות אחוזים כלשהי לכל העובדים שאחראים על תפקיד מסוים (role). הפונקציה בנוסף תחזיר כמה תעלה ההעלאה לחודש, כלומר, מהו סכום ההעלאות הכללי על מנת שנוכל להיערך בהתאם לכך.

קוד הפונקציה:

```
create or replace function update_salaries(emp_role in VARCHAR2, percent in INTEGER)
return NUMBER is
  FunctionResult NUMBER;
  cur_raise number;
  sum_of_raises number := 0;

begin
  for emp in (select *
              from employee
              where erole=emp_role)
  loop
    cur_raise := ceil(emp.salary * (percent) / 100);
    sum_of_raises := sum_of_raises + cur_raise;
    update employee e set e.salary = e.salary + cur_raise
    where e.id = emp.id;
  end loop;
  functionResult := sum_of_raises;
  return(functionResult);
end update_salaries;
```

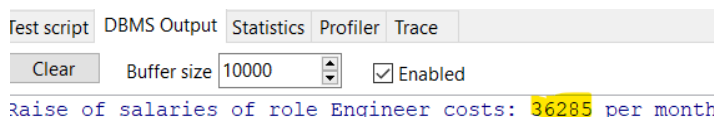
הפונקציה עברה קומפילציה בהצלחה:



נריץ את הפונקציה:

```
declare
  emp_role VARCHAR2(20) := 'Engineer';
  percent INTEGER := 2;
  result NUMBER;
begin
  result := update_salaries(emp_role => emp_role, percent => percent);
  DBMS_OUTPUT.put_line('Raise of salaries of role ' || emp_role || ' costs: ' ||
:result || ' per month ');
end;
```

תוצאה:



	ID	SALARY		ID	SALARY
▶ 1	167479605	86691	▶ 1	167479605	88425
2	939012754	29791	2	939012754	30387
3	540381320	50043	3	540381320	51044
4	984907020	101893	4	984907020	103931
5	462423599	104759	5	462423599	106855
6	205820636	93813	6	205820636	95690
7	129948619	133970	7	129948619	136650
8	419963007	17567	8	419963007	17919
9	865751896	98449	9	865751896	100418
10	341900812	23129	10	341900812	23592

נראה שהמשכורות התעדכנו:

ט"סב

2. נגדיר פונקציה המקבלת כפרמטר סכום כלשהו ובודקת האם קיים נבחר ציבור שמקבל יותר מהסכום הזה כמשכורת, ואם קיים מחזירה 'YES', אחרת – 'NO'. כך התושבים בעירייה יכולים לבדוק ביעילות ובמהירות האם המשכורות בעירייה גבוהות מדי והאם צריך לצאת למחאה על כך שנבחרו הציבור שלהם בעירייה מושחתים.

קוד הפונקציה:

```

create or replace function check_bigger(amount in INTEGER) return varchar2 is
    cursor getSalary (eid INTEGER) is
        SELECT salary
        FROM EMPLOYEE
        WHERE id = eid;

    FunctionResult varchar2(4);
    max_salary INTEGER := 0;
    cur_salary INTEGER;
begin
    for pol_emp in (select *
                    from POLEMP)
        loop
            OPEN getSalary(pol_emp.id);
            fetch getSalary INTO cur_salary;
            IF (max_salary < cur_salary) THEN
                max_salary := cur_salary;
            END IF;
            CLOSE getSalary;
        END LOOP;

    IF (max_salary < amount) THEN
        FunctionResult := 'NO';

    ELSE
        FunctionResult := 'YES';
    END IF;

    return(FunctionResult);
end check_bigger;

```

הפונקציה עברה קומפילציה בהצלחה:

system@XE ➦ [19:01:05] Compiled successfully

תוכנית לבדיקת הפונקציה:

```

declare
    amount INTEGER := 100000;
    result VARCHAR2(5);
begin
    result := check_bigger(amount => amount);
    DBMS_OUTPUT.put_line('Is there a political employee that earns more than ' || amount || '? '
|| result);
end;
```

בדיקת הפונקציה:

Test script DBMS Output Statistics Profiler Trace

Clear Buffer size 10000 ☒ Enabled

Is there a political employee that earns more than 100000? YES

SQL Output Statistics

```
SELECT * FROM EMPLOYEE NATURAL JOIN POLEMP WHERE SALARY > 100000;
```

	ID	ENAME	ADDRESS	PHONE	SALARY	BAN
▶	1	492083823 Peter Evans	40 Horsham Road	0585982132	135177	26-8
2	420676691 Elijah Kweller	728 Columbia Road	0530118935	135388	12-3	


Test script DBMS Output Statistics Profiler Trace

Clear Buffer size 10000 ☒ Enabled

Is there a political employee that earns more than 500000? NO

SQL Output Statistics

```
SELECT * FROM EMPLOYEE NATURAL JOIN POLEMP WHERE SALARY > 500000;
```



	ID	ENAME	EADDRESS	PHONE	SALARY	BANKDETAIL	EROLE	EMAIL
--	----	-------	----------	-------	--------	------------	-------	-------

1. ניצור פרוצדורה המקבלת שם של וועדה ותאריך כלשהו ומוחקת את כל הפניות שהתקבלו בוועדה שהתקבלה כפרמטר לפני התאריך שהתקבל כפרמטר וטופלו (כלומר '1'=status). פרוצדורה זו יעילה על מנת לחסוך במשאבים של נתוני העירייה שלנו.

קוד הפרוצדורה:

```
create or replace procedure delete_inquiries(committee_name in out varchar2, before_date in out
Date) is
    cursor getInquiries(cname varchar2) IS
        SELECT *
        FROM INQUIRIES
        WHERE committee_name = commitname;
    inq_id INQUIRIES.IID%type;
    count_inq INTEGER := 0;

begin
    FOR inq in getInquiries(committee_name)
    LOOP
        IF (inq.idate < before_date AND inq.status = '1') THEN
            inq_id := inq.iid;
            DELETE FROM INQUIRIES I WHERE I.iid = inq_id;
            count_inq := count_inq + 1;
        END IF;
    END LOOP;

    DBMS_OUTPUT.put_line('Deleted ' || count_inq || ' inquiries successfully.');
```

הפרוצדורה עברה קומפילציה בהצלחה:

system@XE [19:58:57] Compiled successfully

הקוד שמריץ את הפרוצדורה:

```
declare
    cname VARCHAR2(20) := 'Health';
    before_date DATE := to_date('31-12-2015', 'DD-MM-YYYY');
begin
    delete_inquiries(committee_name => cname, before_date => before_date);
end;
```

כלומר, נמחק את כל הפניות שהתקבלו בוועדת הבריאות לפני 31.12.2015.

נבדוק שעובד – לפני ההרצה יש לנו 2 תוצאות שעומדות בתנאים:

SQL Output Statistics

```
SELECT * FROM INQUIRIES WHERE idate < to_date('31-12-2015', 'DD-MM-YYYY') AND COMMITNAME = 'Health' AND STATUS = '1'
```

	IID	IDATE	TYPE	DETAILS	STATUS	ID	COMMITNAME
1	2000	04/03/2015	D	YOBvvgxgNmYKfHE41XeGP7vQHtRavY8bfKwRjA	1	898506855	Health
2	2001	04/03/2015	E	Ic31k2wWDCnG1Lm0BeKG89e5k5tOcDeMwk8hAg	1	946240528	Health

Test script DBMS Output Statistics Profiler Trace

Clear Buffer size 10000 ☒ Enabled

Deleted 2 inquiries successfully.

ולאחריה אין כלל:

SQL Output Statistics

```
SELECT * FROM INQUIRIES WHERE idate < to_date('31-12-2015', 'DD-MM-YYYY') AND COMMITNAME = 'Health' AND STATUS = '1'
```

	IID	IDATE	TYPE	DETAILS	STATUS	ID	COMMITNAME
--	-----	-------	------	---------	--------	----	------------

2. ניצור פרוצדורה המקבלת שם של טבלה, שם של עמודה ושם נוסף ומשנה את שם העמודה בטבלה שהתקבל כפרמטר לשם החדש. פרוצדורה זו יעילה למקרים בהם אנשים שפחות מבינים ב-SQL ובפרט ב-ORACLE SQL יוכלו בקלות ובלי סיבוכים מיותרים לשנות שמות של עמודות בטבלאות בלי שתהיה בעיה. בנוסף, בפרוצדורה זו יש בדיקת שגיאות, כלומר היא גם מדפיסה האם השינוי התבצע בהצלחה או לא.

קוד הפרוצדורה:

```
create or replace procedure change_column_name(table_name in out varchar2, old_name in out
varchar2, new_name in out varchar2) is
  command varchar(100);
begin
  command := 'alter table ';
  command := command || table_name;
  command := command || ' rename column ';
  command := command || old_name;
  command := command || ' to ';
  command := command || new_name;

  EXECUTE IMMEDIATE command;
  COMMIT;

  DBMS_OUTPUT.put_line('Column name changed successfully!');

EXCEPTION
  WHEN OTHERS THEN
    DBMS_OUTPUT.put_line('Error in chaging name.');
```

end change_column_name;

הפרוצדורה עברה קומפילציה בהצלחה:

system@XE [22:21:52] Compiled successfully

הקוד שמריץ את הפרוצדורה:

```
declare
  table_name varchar(30) := 'Donor';
  old_name varchar(30) := 'id';
  new_name varchar(30) := 'donor_id';
begin
  change_column_name(table_name => table_name,
                    old_name => old_name,
                    new_name => new_name);
end;
```

כלומר, הקוד מריץ כפרמטרים לפונקציה כך שבטבלה Donor ישתנה שם העמודה "id" ל"donor_id".

נבדוק שעובד:

אחרי ההרצה:

SQL		
Output		
Statistics		
SELECT * FROM DONOR;		
<div> <div>Test script</div> <div>DBMS Output</div> <div>Statistics</div> <div>Profiler</div> <div>Trace</div> </div> <div> <div>Clear</div> <div>Buffer size</div> <div>10000</div> <div>Enabled</div> </div> <div>Column name changed successfully!</div>		
	NAME	DONOR_ID
1	Pablo Lucien	381180820
2	Pablo Lucien	381180820

לפני ההרצה:

SQL		
Output		
Statistics		
SELECT * FROM DONOR;		
<div> <div>Test script</div> <div>DBMS Output</div> <div>Statistics</div> <div>Profiler</div> <div>Trace</div> </div> <div> <div>Clear</div> <div>Buffer size</div> <div>10000</div> <div>Enabled</div> </div> <div>Column name changed successfully!</div>		
	NAME	ID
1	Pablo Lucien	381180820
2	Pablo Lucien	381180820

ט"סב

2. ניצור טריגר שיוצר ערעור (דהיינו Inquiry) אוטומטי על כל קנס (דהיינו Fine) שניתן, על מנת שהתושב בעירייה לא יצטרך לעשות זאת בעצמו, ובכך ייחסך לו ולאחרים הנוגעים בעניין זמן יקר ועוגמת נפש רבה והעיר כולה צהלה ושמחה 😊

קוד הטריגר:

```

create or replace trigger auto_appeal
after insert
on fines
for each row
declare
    last_iid INTEGER;
    fine_date DATE;
    fined_id INTEGER;
    details VARCHAR(100);
begin
    SELECT MAX(IID) INTO last_iid FROM INQUIRIES;
    SELECT sdate INTO fine_date from SOURCE WHERE sid = :new.sid;
    SELECT id INTO fined_id FROM COLLECTION WHERE sid = :new.sid;
    details := 'Appeal for fine number ' || :new.sid;

    INSERT INTO INQUIRIES (IID, IDATE, TYPE, DETAILS, STATUS, ID, COMMITNAME)
    VALUES (last_iid + 1, fine_date, 'A', details, '1', fined_id, 'Welfare');
    DBMS_OUTPUT.put_line('Automatic appeal was sent!');
end auto_appeal;

```

הטריגר עובר קומפילציה בהצלחה:

```
system@XE ➦ [00:09:02] Compiled successfully
```

נבדוק שהוא אכן עובד – ננסה לבצע את הפקודה הבאה:

SQL	Output	Statistics
INSERT INTO FINES (REASON, SID, IID) VALUES ('Parking', 678, 788814968);		

ניתן לראות שאכן בחלון ה-output מופיעה לנו ההודעה שצינינו:

SQL Output Statistics

Clear Buffer size 10000

Automatic appeal was sent!

ולבסוף נבדוק שאכן נוצר ערעור / פנייה כזאת:

SQL Output Statistics

```
SELECT * FROM INQUIRIES I WHERE I.DETAILS LIKE 'Appeal for fine number%';
```

	IID	IDATE	TYPE	DETAILS	STATUS	ID	COMMITNAME
▶ 1	2000	17/05/2021	A	Appeal for fine number 678 ...	1	836681447	Welfare ...

ואכן הערעור נותר בהצלחה וליהודים היתה אורה ושמחה וששון ויקר!