

מערכת דואר-חבילות

מגישות:

מוריה מזרחי 212319693

הודיה אבידן 213841505

2	תוכן עניינים.....
3	מבוא
4	DSD-ERD
6	יצירת טבלאות
9	DSD על ידי ORACLE
11	הכנסת נתונים
13	גיבוי ושיחזור

בס"ד

מבוא:

מערכת הדואר עיסוקה העיקרי יהיה על שילוח חבילות, כל המערכת תעסוק סביב זה. המערכת תשמור נתונים על עובדים, כמו שם פרטי, משפחה, ת.ז (שגם יהיה המפתח), כתובת וכו'... כמו כן המערכת תשמור נתונים על לקוחות ששולחים את החבילות. כמובן שהמערכת תשמור נתונים על החבילות עצמן, המידע על הלקוחות שמקבלים את החבילות ישמר בטבלת החבילות. במידה ומקבל החבילה רוצה להחזיר את החבילה או לשלוח חבילה חדשה הוא נרשם כלקוח (עד עכשיו הפרטים שלו הופיעו רק בטבלת החבילה) ואותה חבילה נרשמת כחבילה חדשה ופרטי המקבל יהיו בטבלת החבילה. במידה וכתובת הנמען והטלפון לא נכונים (מופיעים בחבילה), הדואר ירשום את החבילה כחבילה חדשה עם פרטי הלקוח ששלח כפרטי מקבל המשלוח. בנוסף יהיה נתונים על מעקב אחר חבילות וזימון תורים עבור לקוחות שמוסרים חבילות.

תיאור הקשרים בדיאגרמה

Worker's job:

קשר המבטא את תיאור התפקיד של העובד על פי קוד תפקיד.
כל עובד חייב להיות משויך לתפקיד אחד, לעומת זאת אין כל הגבלה על התפקידים, יתכן וכמה עובדים יעבדו בתפקיד אחד, ויתכן תפקיד ללא עובדים כלל.

Service provider:

קשר המבטא עובד הנותן שירות בתור בדואר. כל תור חייב להיות משויך לעובד אחד, אך אין כל הגבלה על העובדים, יתכן ועובד יטפל בכמה תורים ויתכן שלא יטפל בתורים כלל.

receives service:

קשר המבטא לקוח המקבל שירות בתור בדואר. כל תור חייב להיות משויך ללקוח אחד, אך אין כל הגבלה על הלקוחות, יתכן ולקוח יזמין כמה תורים, ויתכן ולקוח לא יזמין תור כלל. (ישלח חבילה ללא תור)

Sender:

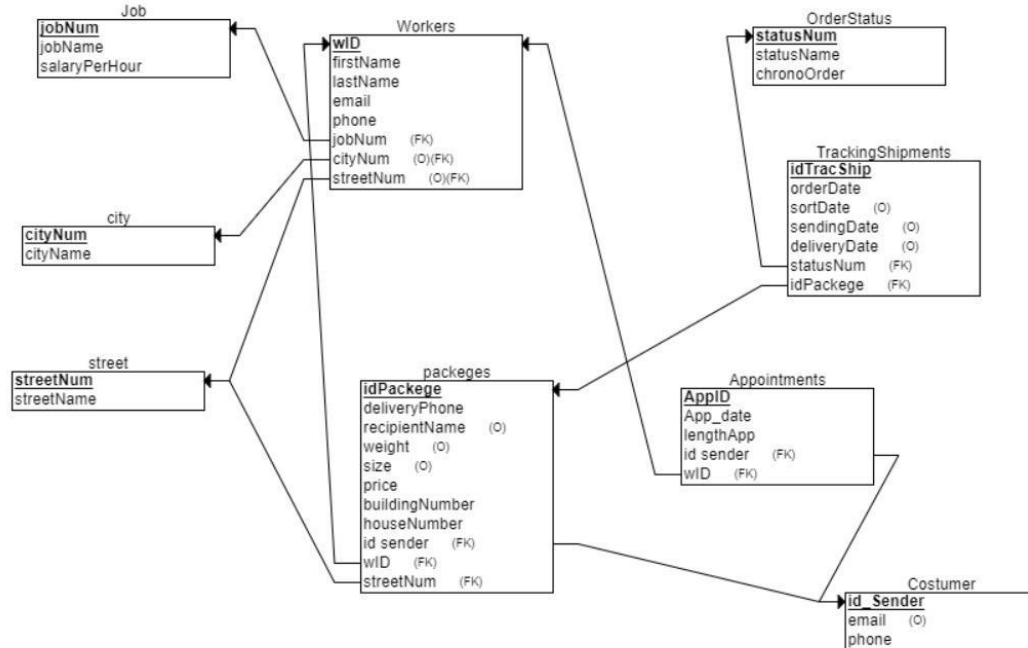
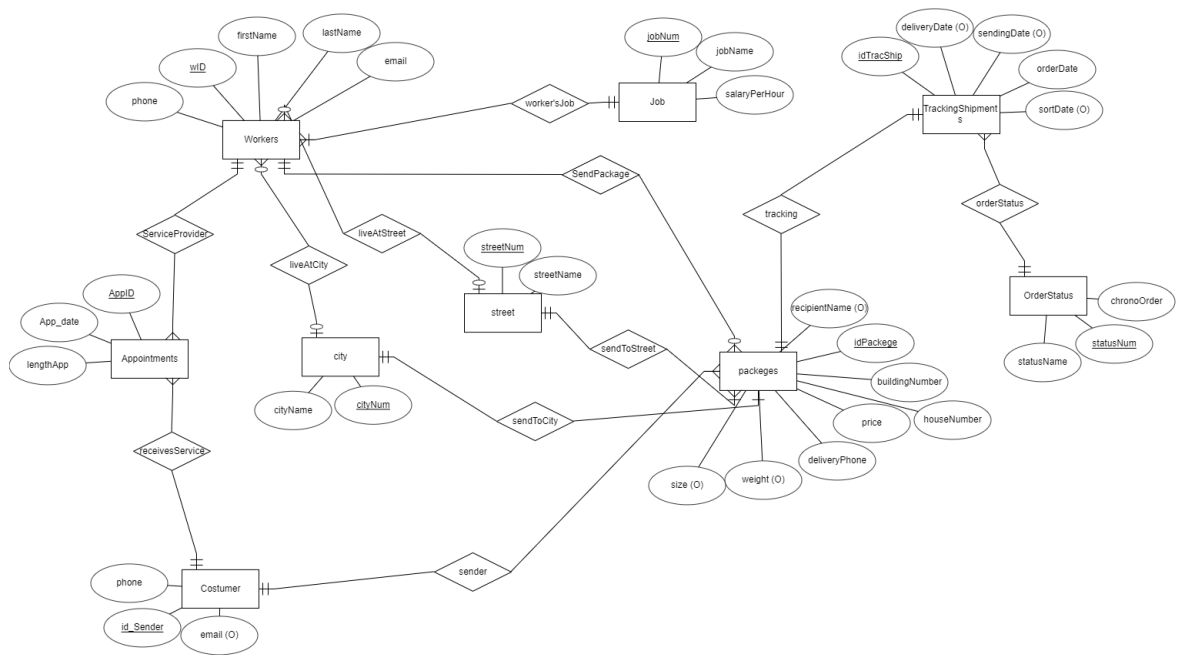
קשר המבטא שליחת חבילה על ידי לקוח. כל חבילה שייכת בהכרח ללקוח אחד בדיוק. לא יתכן חבילה שנשלחה על ידי כמה לקוחות, אך יתכן שאותו לקוח שלח מספר חבילות.

Tracking:

קשר המבטא מעקב אחר חבילה. כל חבילה חייבת שיהיה עבורה מעקב אחד בדיוק. לא יתכן יותר ממעקב אחד לחבילה ולא יתכן יותר מחבילה אחת למעקב מסוים.

orderStatus:

קשר המבטא סטטוס חבילה על ידי קוד של סטטוס מסוים. כל חבילה הנמצאת במעקב חייבת שיהיה עבורה סטטוס אחד ספציפי עדכני לזמן הנוכחי. לא יתכן חבילה במעקב ללא סטטוס, אך יתכן כמה חבילות עם אותו סטטוס ויתכן סטטוס ללא חבילות כלל.



עבור טבלת הנתונים של WORKERS פתחנו עוד טבלה JOB עבור תפקיד העובד, בטבלה זו יהיה מפורט כל תפקידי העובדים שעובדים בדואר הרלוונטיים למערכת זו, כמו פקיד דואר, שליח, ממין וכו'. טבלה זו תהיה מקושרת עם טבלת WORKERS ותסמן את תפקיד העובד.

עבור טבלת הנתונים TRACKINGSHIPMENTS פתחנו עוד טבלה ORDERSTATUS, טבלה זו תפרט את מיקום ההזמנה. טבלה זו תהיה מקושרת עם טבלת TRACKINGSHIPMENTS ותעדכן כל פעם את מיקום החבילה העדכני.

בנוסף פתחנו עוד טבלאות CITY, STREET שהתחברו עם טבלאות שמופיעות בהן כותבות כמו עובדים וחבילות.

עבור חבילה יהיה מקושר לקוח, שולח החבילה ועובדים שמתעסקים עם החבילה.

```
CREATE TABLE Job
(
  jobNum NUMERIC(2) NOT NULL,
  jobName VARCHAR(15) NOT NULL,
  salaryPerHour FLOAT NOT NULL,
  PRIMARY KEY (jobNum)
);

CREATE TABLE Costumer
(
  id_sender NUMERIC(3) NOT NULL,
  email VARCHAR(15),
  phone NUMERIC(10) NOT NULL,
  PRIMARY KEY (id_sender)
);

CREATE TABLE OrderStatus
(
  statusNum NUMERIC(3) NOT NULL,
  statusName VARCHAR(15) NOT NULL,
  chronoOrder NUMERIC(1) NOT NULL,
  PRIMARY KEY (statusNum)
);

CREATE TABLE city
(
  cityName VARCHAR(15) NOT NULL,
  cityNum NUMERIC(3) NOT NULL,
  PRIMARY KEY (cityNum)
);

CREATE TABLE street
(
  streetNum NUMERIC(3) NOT NULL,
  streetName VARCHAR(15) NOT NULL,
  PRIMARY KEY (streetNum)
);

CREATE TABLE Workers
(
  wID NUMERIC(3) NOT NULL,
  firstName VARCHAR(15) NOT NULL,
```

```

lastName VARCHAR(15) NOT NULL,
email VARCHAR(15),
phone NUMERIC(10) NOT NULL,
jobNum NUMERIC(2) NOT NULL,
cityNum NUMERIC(3),
streetNum NUMERIC(3),
PRIMARY KEY (wID),
FOREIGN KEY (jobNum) REFERENCES Job(jobNum),
FOREIGN KEY (cityNum) REFERENCES city(cityNum),
FOREIGN KEY (streetNum) REFERENCES street(streetNum)
);

```

```

CREATE TABLE Appointments
(
  App_date DATE NOT NULL,
  AppID NUMERIC(3) NOT NULL,
  lengthApp NUMERIC(2) NOT NULL,
  id_sender NUMERIC(3) NOT NULL,
  wID NUMERIC(3) NOT NULL,
  PRIMARY KEY (AppID),
  FOREIGN KEY (id_sender) REFERENCES Costumer(id_sender),
  FOREIGN KEY (wID) REFERENCES Workers(wID)
);

```

```

CREATE TABLE packageges
(
  idPackege NUMERIC(3) NOT NULL,
  deliveryPhone NUMERIC(10) NOT NULL,
  recipientName VARCHAR(15),
  weight FLOAT,
  packsize FLOAT,
  price FLOAT,
  buildingNumber NUMERIC(2) NOT NULL,
  houseNumber NUMERIC(2),
  id_sender NUMERIC(3) NOT NULL,
  wID NUMERIC(3) NOT NULL,
  streetNum NUMERIC(3) NOT NULL,
  PRIMARY KEY (idPackege),
  FOREIGN KEY (id_sender) REFERENCES Costumer(id_sender),
  FOREIGN KEY (wID) REFERENCES Workers(wID),
  FOREIGN KEY (streetNum) REFERENCES street(streetNum)
);

```

```

CREATE TABLE TrackingShipments

```

```
(  
  idTracShip NUMERIC(3) NOT NULL,  
  orderDate DATE NOT NULL,  
  sortDate DATE,  
  sendingDate DATE,  
  deliveryDate DATE,  
  statusNum NUMERIC(3) NOT NULL,  
  idPackege NUMERIC(3) NOT NULL,  
  PRIMARY KEY (idTracShip),  
  FOREIGN KEY (statusNum) REFERENCES OrderStatus(statusNum),  
  FOREIGN KEY (idPackege) REFERENCES packeges(idPackege)  
);
```


PL/SQL Developer - [New] - system@XE - [Command Window - New]

File XCS Project Edit Session Debug Tools Deployment Reports Window Help

Connections

Imported Fixed Users
Imported History
Recent

Objects

Files
CURRENT USER
All objects
APPPOINTMENTS
PACKAGES
TRACKINGSHIPMENTS
APPPOINTMENTS
JOB
WORKERS
Indexes
Comments

Templates

Window list Templates

Constants
Default
DDL statements
Error handling
Limits
Package elements
PL/SQL types
Program units
SQL Functions
Type elements
Variables

SQL: dear

Name	Type	Nullable	Default	Comments
CITYTOWN	NUMBER(3)	Y		
STREETOWN	NUMBER(3)	Y		
AFFID	DATE			
AFFID	NUMBER(3)			
LEADSCHAPP	NUMBER(2)			
ID_SENDER	NUMBER(3)			
WID	NUMBER(3)			

SQL: dear packages

Name	Type	Nullable	Default	Comments
IDPACKAGE	NUMBER(3)			
DELIVERYPHONE	NUMBER(10)			
RECIPIENTPHONE	VARCHAR2(15)	Y		
WEIGHT	FLOAT	Y		
PACKAGE	FLOAT	Y		
PRICE	FLOAT	Y		
BUILDINGNUMBER	NUMBER(2)			
HOUSENUMBER	NUMBER(2)	Y		
ID_SENDER	NUMBER(3)			
WID	NUMBER(3)			
STREETOWN	NUMBER(3)			

SQL: dear trackingshipments

Object trackingshipments does not exist.

SQL: dear trackingshipments

Object trackingshipments does not exist.

SQL: dear trackingshipments

Name	Type	Nullable	Default	Comments
IDTRACKSHIP	NUMBER(3)			
ORDERDATE	DATE			
SORTDATE	DATE	Y		
SENDINGDATE	DATE	Y		
DELIVERDATE	DATE	Y		
STATUSOWN	NUMBER(3)			
IDPACKAGE	NUMBER(3)			

SQL:

976 ECH TRM FDB VER APR HDG TWO system@XE login.sql executed

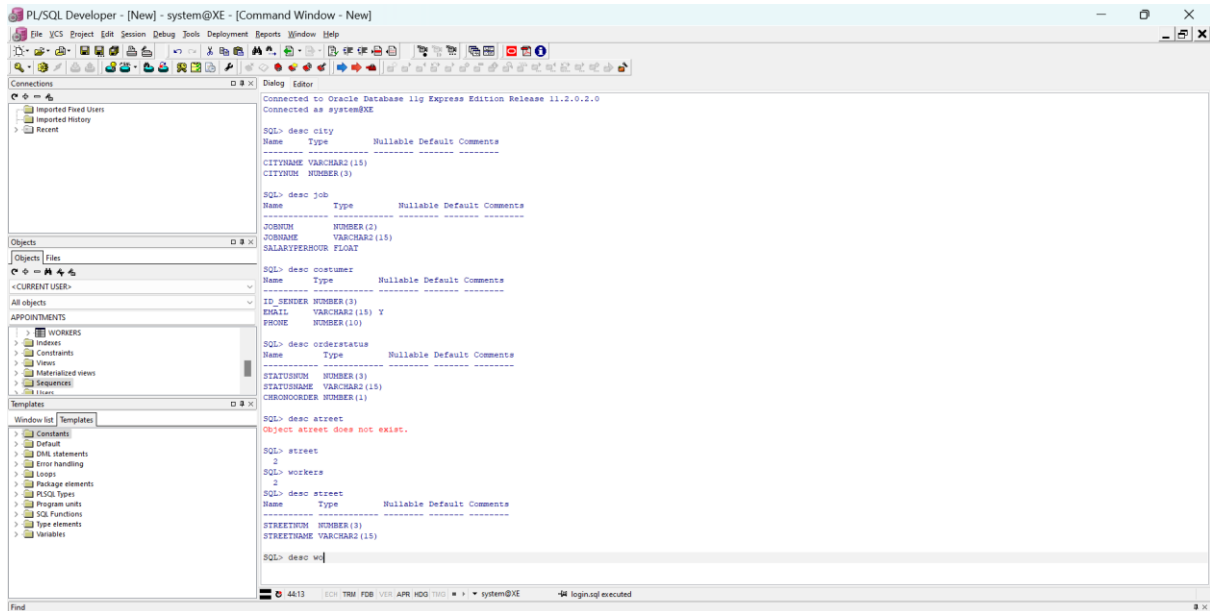
The screenshot shows the PL/SQL Developer application window. The title bar reads 'PL/SQL Developer - [New] - system@XE - [Command Window - New]'. The menu bar includes File, XCS, Project, Edit, Session, Debug, Tools, Deployment, Reports, Window, and Help. The toolbar contains various icons for file operations, editing, and development. The left-hand pane has three tabs: 'Connections', 'Objects', and 'Templates'. The 'Objects' tab is active, showing a tree view of database objects. The main editor area displays the output of the 'DESC' command for three tables: 'deac_street', 'deac_workers', and 'deac_packages'. The output shows columns, data types, and nullability for each table.

Name	Type	Nullable	Default	Comments
STREETNUM	NUMBER(3)			
STREETNAME	VARCHAR2(15)			

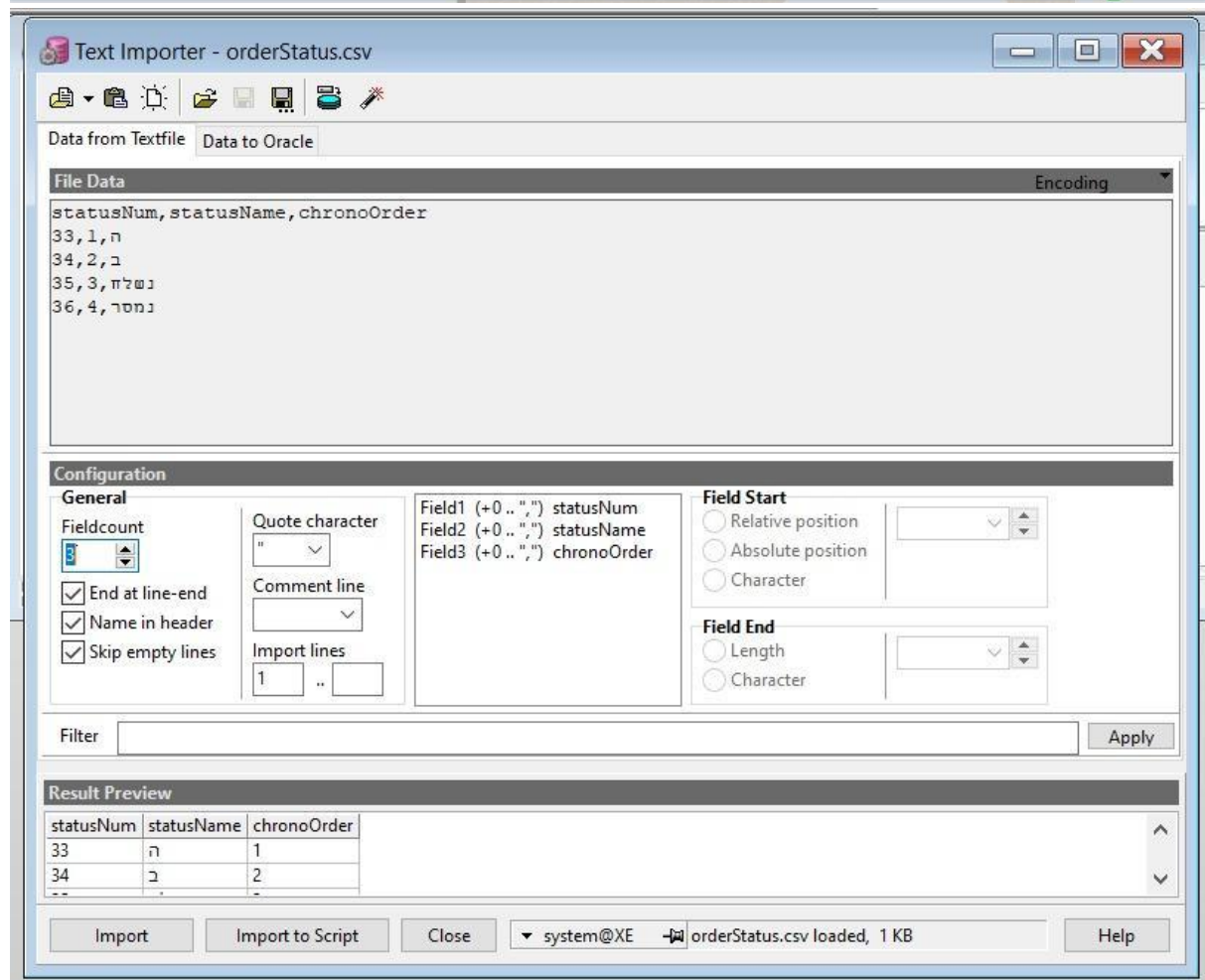
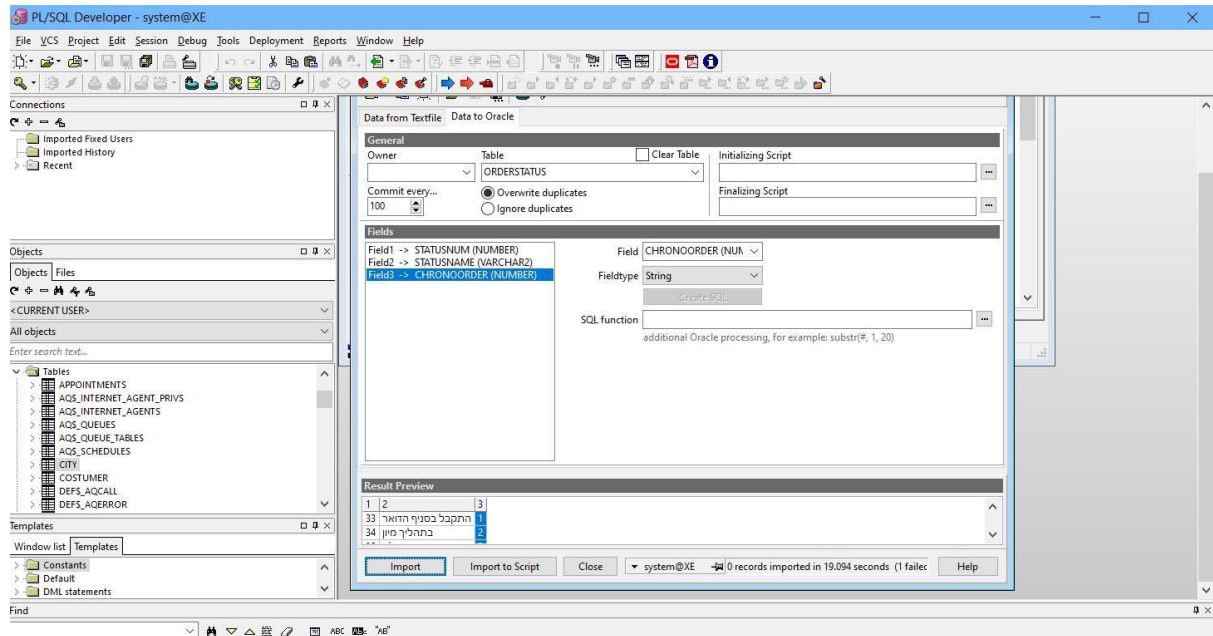
Name	Type	Nullable	Default	Comments
WID	NUMBER(3)			
FIRSTNAME	VARCHAR2(15)			
LASTNAME	VARCHAR2(15)			
EMAIL	VARCHAR2(15)	Y		
PHONE	NUMBER(10)			
JOBNUM	NUMBER(2)			
CITYNUM	NUMBER(3)	Y		
STREETNUM	NUMBER(3)	Y		

Name	Type	Nullable	Default	Comments
APP_DATE	DATE			
APPID	NUMBER(3)			
LENGTH	NUMBER(2)			
ID_SENDER	NUMBER(3)			
WID	NUMBER(3)			

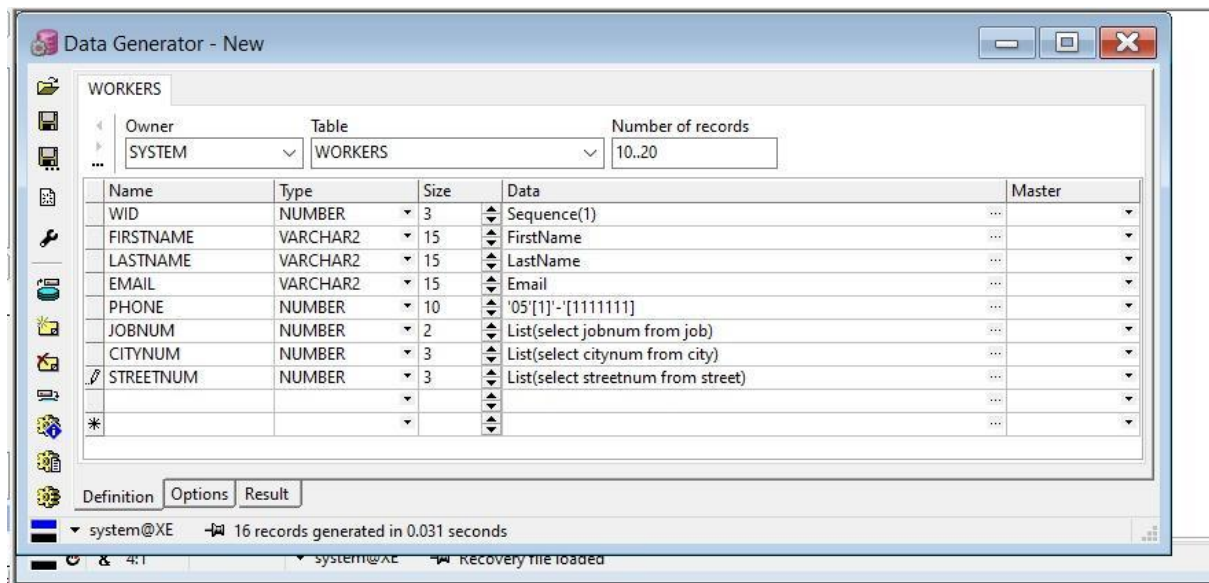
Name	Type	Nullable	Default	Comments
IMPACFGE	NUMBER(3)			
DELIVERYPHONE	NUMBER(10)			
RESIDENTNAME	VARCHAR2(15)	Y		
WEIGHT	FLOAT	Y		
PACKSIZE	FLOAT	Y		
PRICE	FLOAT	Y		
BUILDINGNUMBER	NUMBER(2)			
HOUSENUMBER	NUMBER(2)	Y		
ID_SENDER	NUMBER(3)			
WID	NUMBER(3)			
STREETNUM	NUMBER(3)			



הכנסת הנתונים: 1. בחרנו להכניס נתונים על ידי קבצי CSV



2. שיטת data generator



3.

