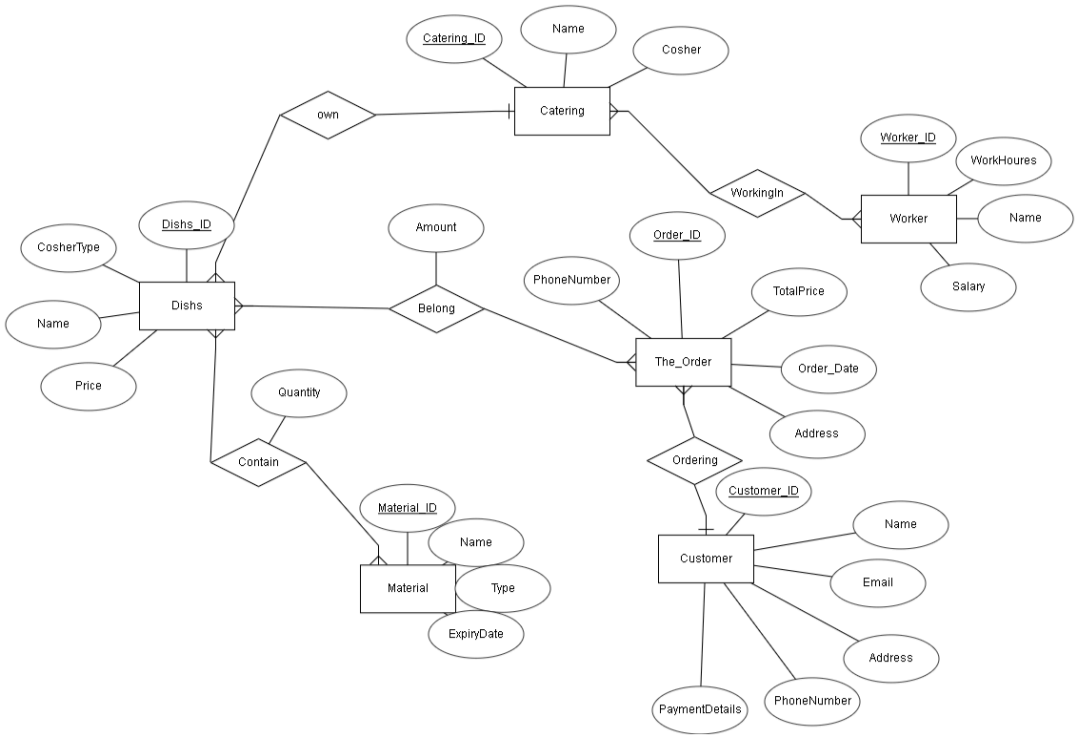
בס"ד

דו"ח מיני פרויקט בבסיסי נתונים

שרית תיק - 213230048 ואיטי ישראלי - 213712367

אוכל - קייטרינג

יצרנו דיאגרמת ERD המתארת פרויקט קייטרינג שבו בחרנו בכיתה.



catering - ישות הקייטרינג מכילה בתוכה 3 אטריביוטים. אטריביוט לזיהוי, לשם הקייטרינג וכשרות הקייטרינג קישרנו ישות זו לישות **Worker** בקשר של רבים לרבים משום שלכל קייטרינג יכול להיות אפס עובדים או יותר ויכול להיות עובש שעובד באפס קייטרינגים או יותר. וישות **Dishs** קישרנו אותה לקייטרינג בקשר של לכל היותר אחד כי לכל קייטרינג יכול להיות אפס או יותר מנות ולכל תבשיל חייב להיות במקסימום קייטרינג אחד שמכין אותה (תבשיל מיוחד לקייטרינג ספציפי):

Worker - ישות עובד מכילה בתוכה 4 אטריביוטים. אטריביוט לזיהוי, שם העובד, שעות עבודה ומשכורת

Dishs - ישות מנות מכילה בתוכה 4 אטריביוטים. אטריביוט לזיהוי, חלבי/בשרי, שם המאכל, ומחיר המאכל

קישרנו ישות זו לישות **Material** בקשר של רבים ליחיד משום שלכך תבשיל יש אפס או יותר מרכיבים ולכל מרכיב הוא יכול להיות באפס או תבשיל אחד בדיוק, וישות **The_Order** בקשר של רבים לרבים משום שלכל הזמנה יכול להיות אפס או יותר תבשילים וכל תבשיל יכול להיות באפס או יותר הזמנות:

Material - ישות חומר גלם מכילה בתוכה 4 אטריביוטים. אטריביוט לזיהוי, לשם החומר גלם, לסוג החומר גלם (צומח,אביזרי מכולת וכו) ותאריך תפוגת החומר גלם.

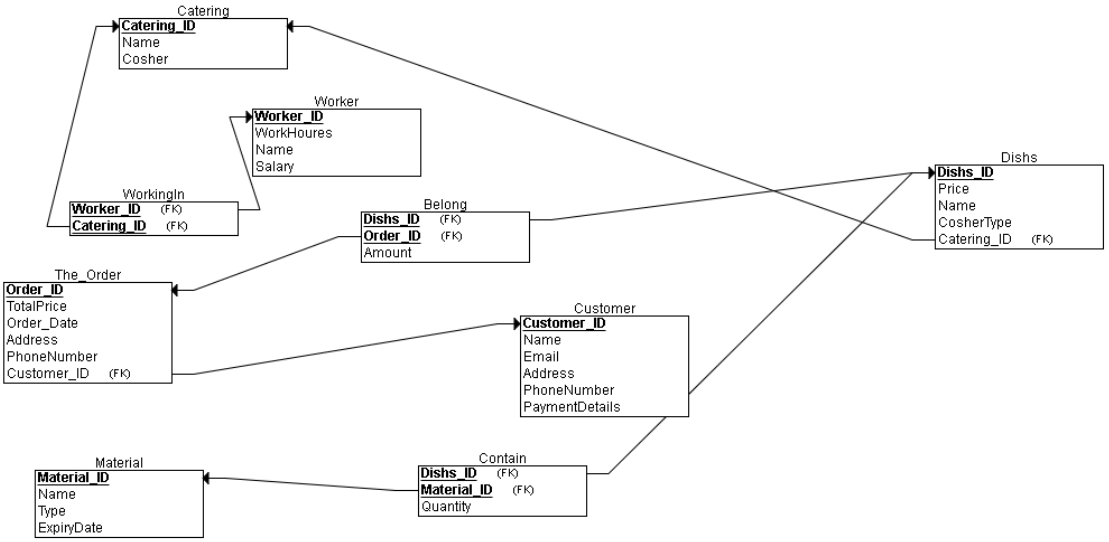
The_Order - ישות הזמנות מכילה בתוכה חמישה אטריביוטים. אטריביוט לזיהוי, כתובת המזמין, טלפון המזמין, תאריך ההזמנה ומחיר כולל של ההזמנה

קישרנו ישות זו לישות **Customer** בקשר של בדיוק אחד משום שכל הזמנה שייכת ללקוח אחד בלבד:

Customer - ישות לקוח מכילה בתוכה שישה אטריביוטים. אטריביוט לזיהוי, שם הלקוח, כתובת מייל של הלקוח, כתובת הלקוח, מספר הטלפון של הלקוח, ופרטי תשלום (אשראי, פייפאל וכו).

בס"ד

תרשים הDSD:



פקודת הCreateTable:

הוספנו אילוצים על שני טבלאות:

Catering - על שדה הכשרות הוספנו את האופציה שיוכלו רק להכניס Y/N, הכוונה האם הקייטרינג כשר או לא.

Dishs - על שדה הסוג כשרות, שיוכלו להכניס את האופציות חלבי/ בשרי/ או אף אחד - פרווה, וככה ידעו

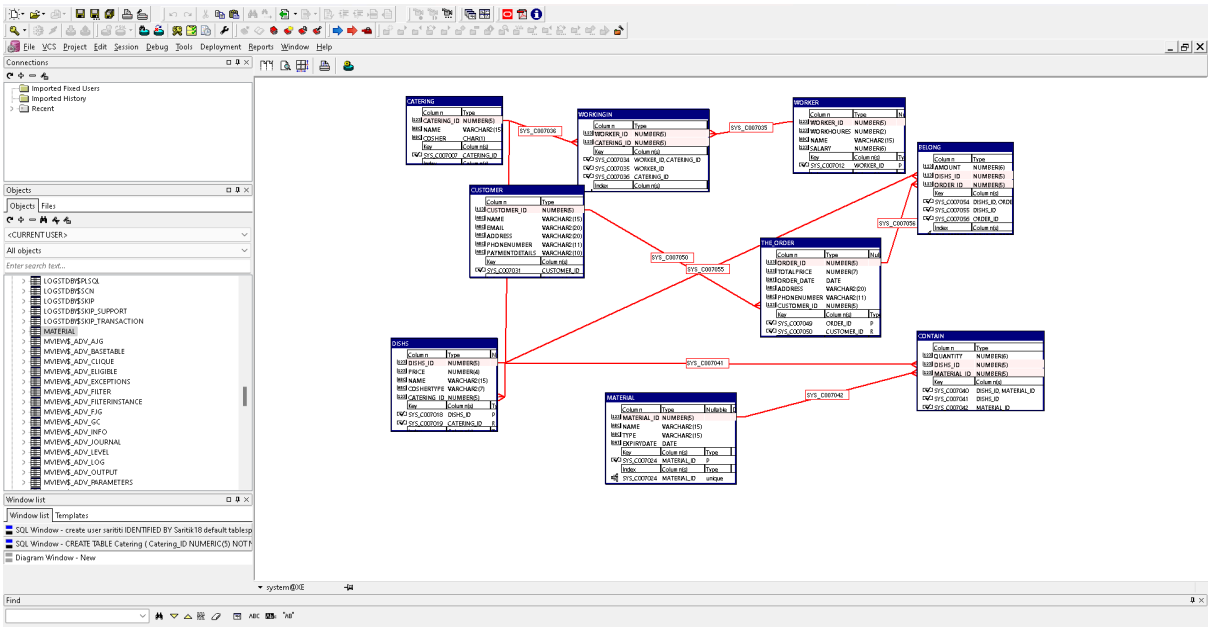
באיזה מאכל מדובר, מאכל חלבי, מאכל בשרי או מאכל פרווה.

```
CREATE TABLE Catering
(
  Catering_ID NUMERIC(5) NOT NULL,
  Name VARCHAR(15) NOT NULL,
  Cosher CHAR(1) NOT NULL CHECK(cosher='Y' or cosher='N'),
  PRIMARY KEY (Catering_ID)
);

CREATE TABLE Worker
(
  Worker_ID NUMERIC(5) NOT NULL,
  WorkHoures NUMERIC(2) NOT NULL,
  Name VARCHAR(15) NOT NULL,
  Salary NUMERIC(6) NOT NULL,
  PRIMARY KEY (Worker_ID)
);

CREATE TABLE Dishs
(
  Dishs_ID NUMERIC(5) NOT NULL,
  Price NUMERIC(4) NOT NULL,
  Name VARCHAR(15) NOT NULL,
  CosherType VARCHAR(7) NOT NULL CHECK(cosherType='Dairy' or cosherType='Meat' or cosherType='None'),
  Catering_ID NUMERIC(5) NOT NULL,
  PRIMARY KEY (Dishs_ID),
  FOREIGN KEY (Catering_ID) REFERENCES Catering(Catering_ID)

);

CREATE TABLE Material
(
  Material_ID NUMERIC(5) NOT NULL,
  Name VARCHAR(15) NOT NULL,
  Type VARCHAR(15) NOT NULL,
  ExpiryDate DATE NOT NULL,
  PRIMARY KEY (Material_ID)
);

CREATE TABLE Customer
(
  Customer_ID NUMERIC(5) NOT NULL,
  Name VARCHAR(15) NOT NULL,
  Email VARCHAR(20) NOT NULL,
  Address VARCHAR(20) NOT NULL,
  PhoneNumber VARCHAR(11) NOT NULL,
  PaymentDetails VARCHAR(10) NOT NULL,
  PRIMARY KEY (Customer_ID)
```

בס"ד

```sql
  Email VARCHAR(20) NOT NULL,
  Address VARCHAR(20) NOT NULL,
  PhoneNumber VARCHAR(11) NOT NULL,
  PaymentDetails VARCHAR(10) NOT NULL,
  PRIMARY KEY (Customer_ID)
);

CREATE TABLE WorkingIn
(
  Worker_ID NUMERIC(5) NOT NULL,
  Catering_ID NUMERIC(5) NOT NULL,
  PRIMARY KEY (Worker_ID, Catering_ID),
  FOREIGN KEY (Worker_ID) REFERENCES Worker(Worker_ID),
  FOREIGN KEY (Catering_ID) REFERENCES Catering(Catering_ID)
);

CREATE TABLE Contain
(
  Quantity NUMERIC(6) NOT NULL,
  Dishs_ID NUMERIC(5) NOT NULL,
  Material_ID NUMERIC(5) NOT NULL,
  PRIMARY KEY (Dishs_ID, Material_ID),
  FOREIGN KEY (Dishs_ID) REFERENCES Dishs(Dishs_ID),
  FOREIGN KEY (Material_ID) REFERENCES Material(Material_ID)
);

CREATE TABLE The_Order
(
  Order_ID NUMERIC(5) NOT NULL,
  TotalPrice NUMERIC(7) NOT NULL,
  Order_Date DATE NOT NULL,
  Address VARCHAR(20) NOT NULL,
  PhoneNumber VARCHAR(11) NOT NULL,
  Customer_ID NUMERIC(5) NOT NULL,
  PRIMARY KEY (Order_ID),
  FOREIGN KEY (Customer_ID) REFERENCES Customer(Customer_ID)
);

CREATE TABLE Belong
(
  Amount NUMERIC(6) NOT NULL,
  Dishs_ID NUMERIC(5) NOT NULL,
  Order_ID NUMERIC(5) NOT NULL,
  PRIMARY KEY (Dishs_ID, Order_ID),
  FOREIGN KEY (Dishs_ID) REFERENCES Dishs(Dishs_ID),
  FOREIGN KEY (Order_ID) REFERENCES The_Order(Order_ID)
);
```

טבלאות הSQL:



```
Connected to Oracle Database 11g Express Edition Release 11.2.0.2.0
Connected as Saritit1@XE

SQL> desc Belong
Name     Type      Nullable Default Comments
-------- --------- -------- ------- --------
AMOUNT   NUMBER(6)
DISHS_ID NUMBER(5)
ORDER_ID NUMBER(5)

SQL> desc Catering
Name        Type         Nullable Default Comments
----------- ------------ -------- ------- --------
CATERING_ID NUMBER(5)
NAME        VARCHAR2(15)
COSHER      CHAR(1)

SQL> desc Contain
Name        Type      Nullable Default Comments
----------- --------- -------- ------- --------
QUANTITY    NUMBER(6)
DISHS_ID    NUMBER(5)
MATERIAL_ID NUMBER(5)

SQL> desc Customer
Name            Type         Nullable Default Comments
--------------- ------------ -------- ------- --------
CUSTOMER_ID     NUMBER(5)
NAME            VARCHAR2(15)
EMAIL           VARCHAR2(20)
ADDRESS         VARCHAR2(20)
PHONENUMBER     VARCHAR2(11)
PAYMENTDETAILS VARCHAR2(10)

SQL> desc Dishs
Name        Type         Nullable Default Comments
----------- ------------ -------- ------- --------
DISHS_ID    NUMBER(5)
PRICE       NUMBER(4)
NAME        VARCHAR2(15)
COSHERTYPE  VARCHAR2(7)
CATERING_ID NUMBER(5)

SQL> desc Material
Name        Type         Nullable Default Comments
----------- ------------ -------- ------- --------
MATERIAL_ID NUMBER(5)
```

```
EMAIL           VARCHAR2(20)
ADDRESS         VARCHAR2(20)
PHONENUMBER     VARCHAR2(11)
PAYMENTDETAILS VARCHAR2(10)

SQL> desc Dishs
Name        Type         Nullable Default Comments
----------- ------------ -------- ------- --------
DISHS_ID    NUMBER(5)
PRICE       NUMBER(4)
NAME        VARCHAR2(15)
COSHERTYPE  VARCHAR2(7)
CATERING_ID NUMBER(5)

SQL> desc Material
Name        Type         Nullable Default Comments
----------- ------------ -------- ------- --------
MATERIAL_ID NUMBER(5)
NAME        VARCHAR2(15)
TYPE        VARCHAR2(15)
EXPIRYDATE  DATE

SQL> desc The_Order
Name        Type         Nullable Default Comments
----------- ------------ -------- ------- --------
ORDER_ID    NUMBER(5)
TOTALPRICE  NUMBER(7)
ORDER_DATE  DATE
ADDRESS     VARCHAR2(20)
PHONENUMBER VARCHAR2(11)
CUSTOMER_ID NUMBER(5)

SQL> desc Worker
Name       Type         Nullable Default Comments
---------- ------------ -------- ------- --------
WORKER_ID  NUMBER(5)
WORKHOURES NUMBER(2)
NAME       VARCHAR2(15)
SALARY     NUMBER(6)

SQL> desc WorkingIn
Name        Type      Nullable Default Comments
----------- --------- -------- ------- --------
WORKER_ID   NUMBER(5)
CATERING_ID NUMBER(5)
```

צילומי מסך של 3 השיטות

שימוש בData Generator: (שמתי טבלה אחת לדוגמא כך בוצעו כל שאר הטבלאות)

בס"ד

WORKER

Owner: SARITITI | Table: WORKER | Number of records: 400

Name	Type	Size	Data	Master
WORKER_ID	NUMBER	5	Sequence(10, 10)	
WORKHOURES	NUMBER	2	Random(4, 12)	
NAME	VARCHAR2	15	FirstName + LastName	
SALARY	NUMBER	6	Random(4000, 20000)	

```
insert into SARITITI.WORKER (WORKER_ID, WORKHOURES, NAME, SALARY)
values (10, 8, 'JonnyLaw', 12405);

insert into SARITITI.WORKER (WORKER_ID, WORKHOURES, NAME, SALARY)
values (20, 11, 'JillShawn', 7380);

insert into SARITITI.WORKER (WORKER_ID, WORKHOURES, NAME, SALARY)
values (30, 5, 'MarleyRobinson', 17894);

insert into SARITITI.WORKER (WORKER_ID, WORKHOURES, NAME, SALARY)
values (40, 8, 'ChuckDysart', 18417);

insert into SARITITI.WORKER (WORKER_ID, WORKHOURES, NAME, SALARY)
values (50, 7, 'CharlesApplegat', 17406);

insert into SARITITI.WORKER (WORKER_ID, WORKHOURES, NAME, SALARY)
values (60, 7, 'ValCarmen', 11853);

insert into SARITITI.WORKER (WORKER_ID, WORKHOURES, NAME, SALARY)
values (70, 10, 'DavidOrbit', 13803);

insert into SARITITI.WORKER (WORKER_ID, WORKHOURES, NAME, SALARY)
values (80, 11, 'BillyGarr', 7561);

insert into SARITITI.WORKER (WORKER_ID, WORKHOURES, NAME, SALARY)
values (90, 7, 'DomRhames', 4366);

insert into SARITITI.WORKER (WORKER_ID, WORKHOURES, NAME, SALARY)
values (100, 6, 'DiamondChanning', 13256);

insert into SARITITI.WORKER (WORKER_ID, WORKHOURES, NAME, SALARY)
values (110, 10, 'RhysDuvall', 6892);

insert into SARITITI.WORKER (WORKER_ID, WORKHOURES, NAME, SALARY)
values (120, 4, 'EttaApplegate', 13085);

insert into SARITITI.WORKER (WORKER_ID, WORKHOURES, NAME, SALARY)
values (130, 7, 'NellyApple', 18779);

insert into SARITITI.WORKER (WORKER_ID, WORKHOURES, NAME, SALARY)
values (140, 7, 'GordArmstrading', 17225);

insert into SARITITI.WORKER (WORKER_ID, WORKHOURES, NAME, SALARY)
values (150, 12, 'MarianneBrown', 5126);

insert into SARITITI.WORKER (WORKER_ID, WORKHOURES, NAME, SALARY)
```

שימוש בקוד python: (קוד הpython פלט לי קובץ טקסט)

CustomerFile.py

```python
import random
import string


# Function to generate a random full name with a maximum length of 15 characters
def random_name():
    first_names = ['John', 'Jane', 'Alex', 'Emily', 'Chris', 'Katie', 'Michael', 'Sarah', 'David', 'Laura']
    last_names = ['Smith', 'Johnson', 'Williams', 'Jones', 'Brown', 'Davis', 'Miller', 'Wilson', 'Moore', 'Taylor']
    while True:
        name = f"{random.choice(first_names)} {random.choice(last_names)}"
        if len(name) <= 15:
            return name
        if len(first_names) > 0:
            first_names.pop()  # Reduce the pool to ensure we don't get into an infinite loop


# Function to generate a random email with a maximum length of 20 characters
def random_email(name):
    domains = ['gmail.com', 'yahoo.com', 'hotmail.com', 'outlook.com', 'example.com']
    email_prefix = name.replace(' ', '.').lower()
    email_suffix = random.choice(domains)
    email = f"{email_prefix}@{email_suffix}"
    if len(email) > 20:
        email = email[:20 - len(email_suffix) - 1] + '@' + email_suffix
    return email[:20]


# Function to generate a random address with a maximum length of 20 characters
def random_address():
```

בס"ד

CustomerFile.py

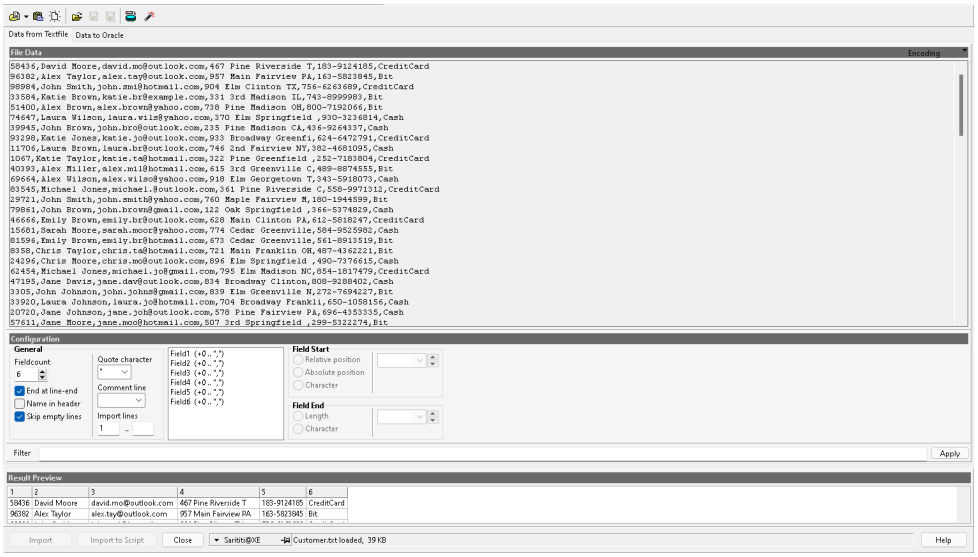
```python
    return email[:20]


# Function to generate a random address with a maximum length of 20 characters
def random_address():
    streets = ['Main', 'High', 'Broadway', 'Elm', 'Maple', 'Oak', 'Pine', 'Cedar', '2nd', '3rd']
    cities = ['Springfield', 'Riverside', 'Greenfield', 'Franklin', 'Clinton', 'Fairview', 'Greenville', 'Bristol', 'Madison', 'Ge
    states = ['CA', 'TX', 'NY', 'FL', 'IL', 'PA', 'OH', 'MI', 'GA', 'NC']
    address = f"{random.randint(100, 999)} {random.choice(streets)} {random.choice(cities)} {random.choice(states)}"
    return address[:20]


# Function to generate a random phone number
def random_phone():
    return f"{random.randint(100, 999)}-{random.randint(100, 999)}{random.randint(1000, 9999)}"


# Function to generate random payment details
def random_payment():
    methods = ['CreditCard', 'Cash', 'Bit']
    return random.choice(methods)


# Function to generate a unique ID
def unique_id(existing_ids):
    while True:
        new_id = random.randint(1, 99999)
        if new_id not in existing_ids:
            existing_ids.add(new_id)
```

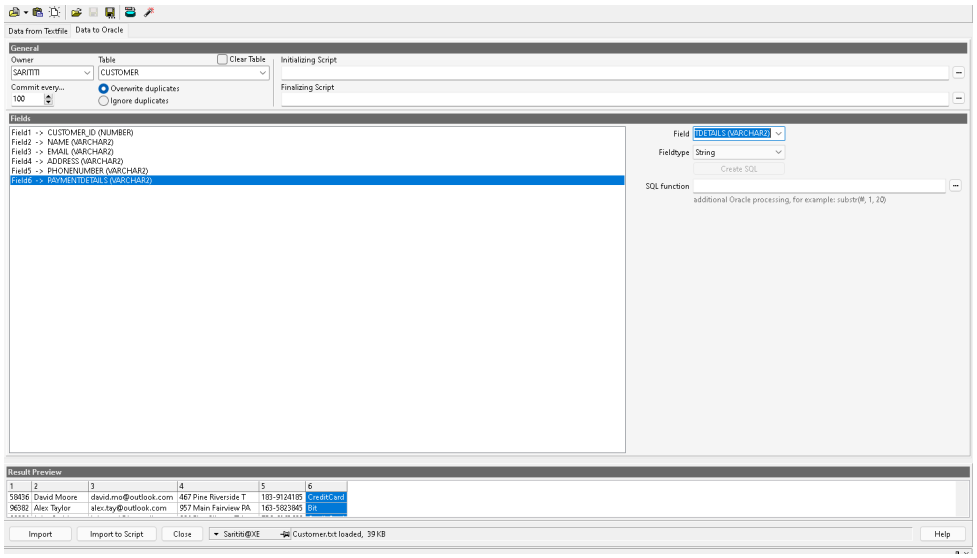
CustomerFile.py

```python
def unique_id(existing_ids):
    while True:
        new_id = random.randint(1, 99999)
        if new_id not in existing_ids:
            existing_ids.add(new_id)
            return new_id


def generate_data_file(filename, num_rows):
    existing_ids = set()
    with open(filename, 'w') as file:
        for _ in range(num_rows):
            id = unique_id(existing_ids)
            name = random_name()
            email = random_email(name)
            address = random_address()
            phone = random_phone()
            payment = random_payment()
            row = f"{id},{name},{email},{address},{phone},{payment}\n"
            file.write(row)


# Generate the data file with 500 rows
generate_data_file('Customer.txt', 500)

print("Data file generated successfully.")
```

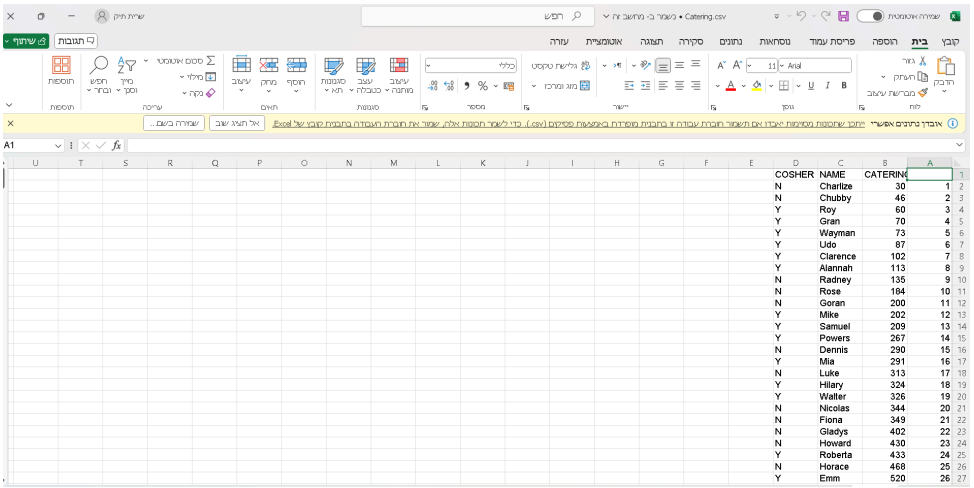
```
David Moore,david.mo@outlook.com,467 Pine Riverside T,183-9124185,CreditCard,58436
Alex Taylor,alex.tay@outlook.com,957 Main Fairview PA,163-5823845,Bit,96382
John Smith,john.smi@hotmail.com,904 Elm Clinton TX,756-6263689,CreditCard,98984
Katie Brown,katie.br@example.com,331 3rd Madison IL,743-8999983,Bit,33584
Alex Brown,alex.brown@yahoo.com,738 Pine Madison OH,800-7192066,Bit,51400
Laura Wilson,laura.wils@yahoo.com,370 Elm Springfield ,930-3236814,Cash,74647
John Brown,john.bro@outlook.com,235 Pine Madison CA,436-9264337,Cash,39945
Katie Jones,katie.jo@outlook.com,933 Broadway Greenfi,624-6472791,CreditCard,93298
Laura Brown,laura.br@outlook.com,746 2nd Fairview NY,382-4681095,Cash,11706
Katie Taylor,katie.ta@hotmail.com,322 Pine Greenfield ,252-7183804,CreditCard,1067
Alex Miller,alex.mil@hotmail.com,615 3rd Greenville C,489-8874555,Bit,40393
Alex Wilson,alex.wilso@yahoo.com,918 Elm Georgetown T,343-5918073,Cash,69664
Michael Jones,michael.@outlook.com,361 Pine Riverside C,558-9971312,CreditCard,83545
John Smith,john.smith@yahoo.com,760 Maple Fairview M,180-1944599,Bit,29721
John Brown,john.brown@gmail.com,122 Oak Springfield ,366-5374829,Cash,79861
Emily Brown,emily.br@outlook.com,628 Main Clinton PA,612-5818247,CreditCard,46666
Sarah Moore,sarah.moor@yahoo.com,774 Cedar Greenville,584-9525982,Cash,15681
Emily Brown,emily.br@hotmail.com,673 Cedar Greenville,561-8913519,Bit,81596
Chris Taylor,chris.ta@hotmail.com,721 Main Franklin OH,487-4362221,Bit,8358
Chris Moore,chris.mo@outlook.com,896 Elm Springfield ,490-7376615,Cash,24296
Michael Jones,michael.jo@gmail.com,795 Elm Madison NC,854-1817479,CreditCard,62454
Jane Davis,jane.dav@outlook.com,834 Broadway Clinton,808-9288402,Cash,47195
John Johnson,john.johns@gmail.com,839 Elm Greenville N,272-7694227,Bit,3305
Laura Johnson,laura.jo@hotmail.com,704 Broadway Frankli,650-1058156,Cash,33920
Jane Johnson,janej.oh@outlook.com,578 Pine Fairview PA,696-4353335,Cash,20720
Jane Moore,jane.moo@hotmail.com,507 3rd Springfield ,299-5322274,Bit,57611
Katie Jones,katie.jo@outlook.com,155 3rd Franklin NY,598-9105113,CreditCard,27791
Emily Jones,emily.jo@example.com,244 Elm Bristol IL,747-1244794,Cash,26189
John Smith,john.smi@example.com,431 Elm Springfield ,429-1484917,Bit,48155
Katie Miller,katie.mill@yahoo.com,979 2nd Greenfield C,299-2474396,Cash,33040
Sarah Smith,sarah.sm@example.com,837 Pine Fairview CA,859-2905573,Cash,49645
Alex Moore,alex.moore@gmail.com,500 Elm Greenville T,632-2388548,CreditCard,55703
Katie Davis,katie.da@hotmail.com,486 Cedar Clinton PA,139-3643067,Bit,30697
Michael Wilson,michael.wi@yahoo.com,282 Pine Bristol TX,799-7762339,Bit,54590
Sarah Miller,sarah.mill@yahoo.com,928 Pine Madison OH,210-3775555,Cash,51101
```

בס"ד



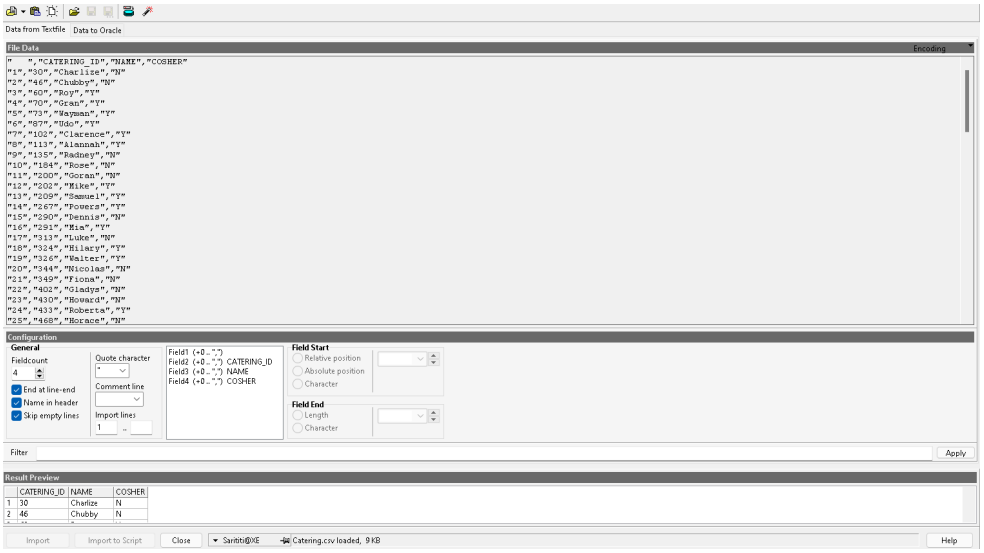


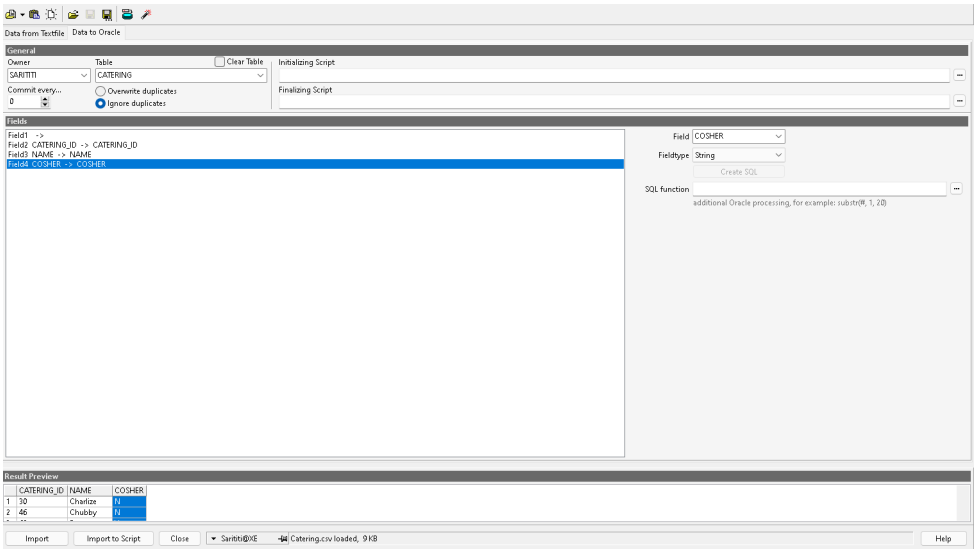
שימוש בקבצי Excel: (שוב שמתי טבלה אחת לדוגמא כך בוצעו כל שאר הטבלאות)

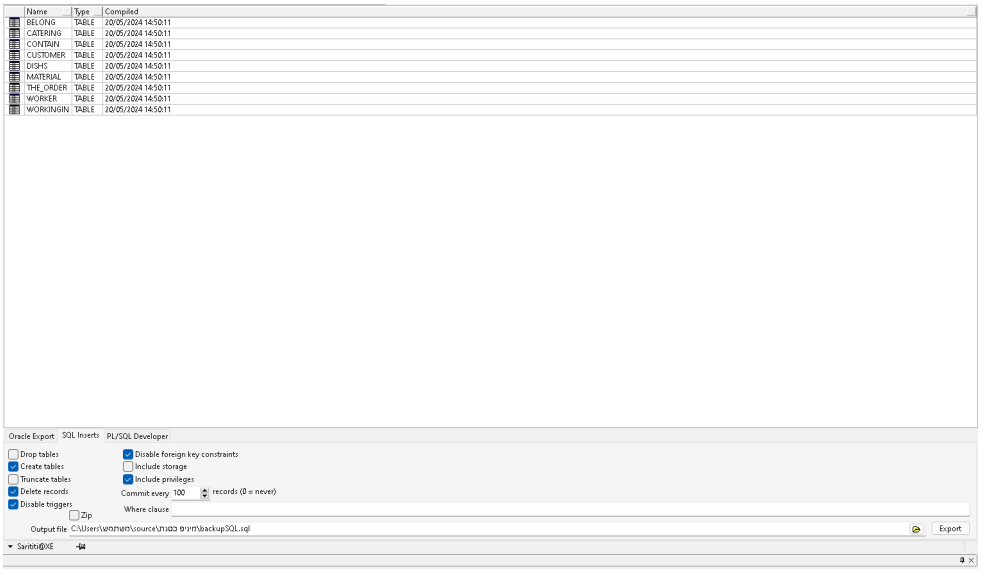


COSHER	NAME	CATERING	
N	Charlize	30	1
N	Chubby	46	2
Y	Roy	60	3
Y	Gran	70	4
Y	Wayman	73	5
Y	Udo	87	6
Y	Clarence	102	7
Y	Alannah	113	8
N	Radney	135	9
N	Rose	184	10
N	Goran	200	11
Y	Mike	202	12
Y	Samuel	209	13
Y	Powers	267	14
N	Dennis	290	15
Y	Mia	291	16
N	Luke	313	17
Y	Hilary	324	18
Y	Walter	326	19
N	Nicolas	344	20
N	Fiona	349	21
N	Gladys	402	22
N	Howard	430	23
Y	Roberta	433	24
N	Horace	468	25
Y	Emm	520	26

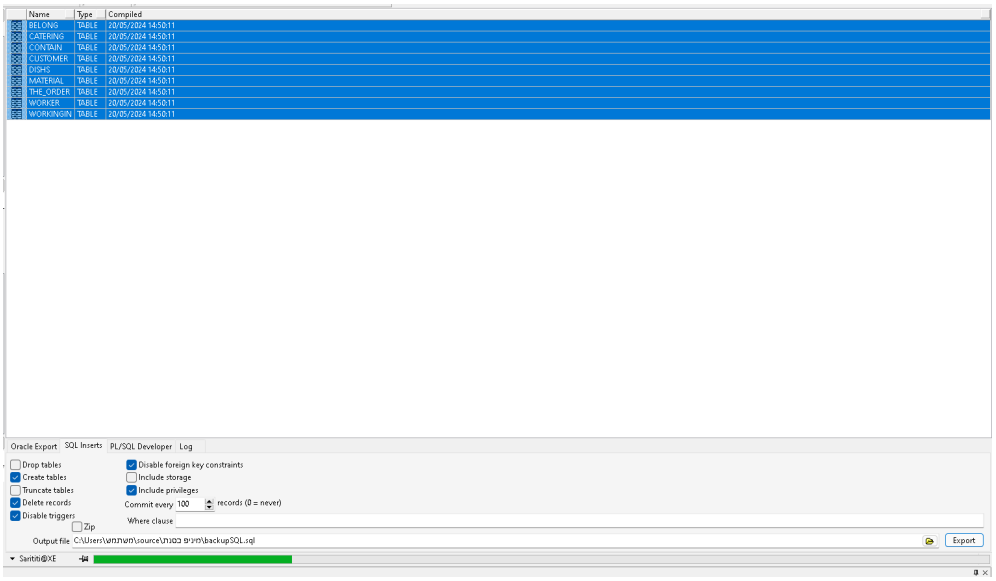
בס"ד


Data from Textfile
Data to Oracle
File Data
Encoding
" ","CATERING_ID","NAME","COSHER"
"1","30","Charlize","N"
"2","46","Chubby","N"
"3","60","Roy","Y"
"4","70","Gran","Y"
"5","73","Wayman","Y"
"6","87","Udo","Y"
"7","102","Clarence","Y"
"8","113","Alannah","Y"
"9","135","Rodney","N"
"10","184","Rose","N"
"11","200","Goran","N"
"12","202","Mike","Y"
"13","209","Samuel","Y"
"14","267","Powers","Y"
"15","290","Dennis","N"
"16","293","Mia","Y"
"17","313","Luke","N"
"18","324","Hilary","Y"
"19","326","Walter","Y"
"20","344","Nicolas","N"
"21","349","Fiona","N"
"22","402","Gladys","N"
"23","430","Howard","N"
"24","433","Roberta","Y"
"25","468","Horace","N"
Configuration
General
Fieldcount
4
Quote character
"
Comment line
Import lines
1
End at line-end
Name in header
Skip empty lines
Field1 (+0-",")
Field2 (+0-",") CATERING_ID
Field3 (+0-",") NAME
Field4 (+0-",") COSHER
Field Start
Relative position
Absolute position
Character
Field End
Length
Character
Filter
Apply
Result Preview
CATERING_ID NAME COSHER
1 30 Charlize N
2 46 Chubby N
Import
Import to Script
Close
Sarititi@XE
Catering.csv loaded, 9 KB
Help


Data from Textfile
Data to Oracle
General
Owner
SARITITI
Table
CATERING
Clear Table
Commit every...
0
Overwrite duplicates
Ignore duplicates
Initializing Script
Finalizing Script
Fields
Field1 ->
Field2 CATERING_ID -> CATERING_ID
Field3 NAME -> NAME
Field4 COSHER -> COSHER
Field COSHER
Fieldtype String
Create SQL
SQL function
additional Oracle processing, for example: substr(#, 1, 20)
Result Preview
CATERING_ID NAME COSHER
1 30 Charlize N
2 46 Chubby N
Import
Import to Script
Close
Sarititi@XE
Catering.csv loaded, 9 KB
Help

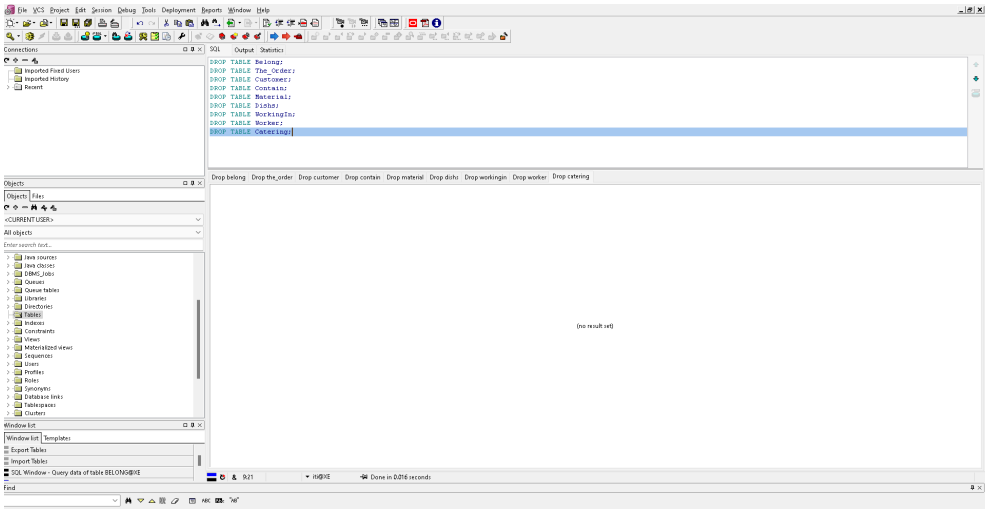

Name Type Compiled
BELONG TABLE 20/05/2024 14:50:11
CATERING TABLE 20/05/2024 14:50:11
CONTAIN TABLE 20/05/2024 14:50:11
CUSTOMER TABLE 20/05/2024 14:50:11
DISHS TABLE 20/05/2024 14:50:11
MATERIAL TABLE 20/05/2024 14:50:11
THE_ORDER TABLE 20/05/2024 14:50:11
WORKER TABLE 20/05/2024 14:50:11
WORKINGIN TABLE 20/05/2024 14:50:11
Oracle Export
SQL Inserts
PL/SQL Developer
Drop tables
Create tables
Truncate tables
Delete records
Disable triggers
Zip
Disable foreign key constraints
Include storage
Include privileges
Commit every 100 records (0 = never)
Where clause
Output file C:\Users\משתמש\source\תיקיה בסוגתו\backupSQL.sql
Export
Sarititi@XE

בס"ד

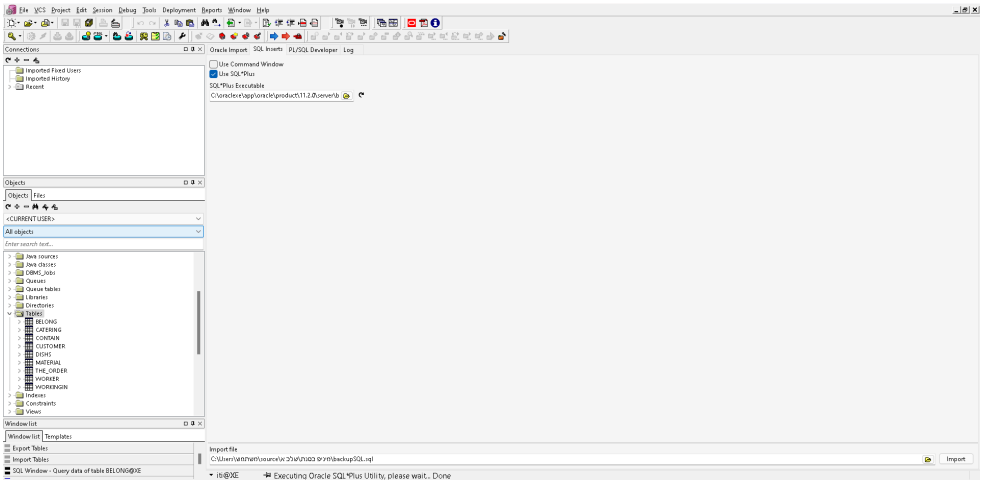


כעת נבצע שיחזור של הנתונים ונבדוק שהגיבוי אכן תקין:

נבצע מחיקה של כל הטבלאות:



ועכשיו נעלה את קובץ הbackup ונראה שכל הטבלאות שוחזרו כראוי:



סוף:)